VAASAN YLIOPISTO

JANNE KOLJONEN

# Computer Vision and Optimization Methods Applied to the Measurements of In-Plane Deformations

ACTA WASAENSIA NO 219

AUTOMATION TECHNOLOGY 1

UNIVERSITAS WASAENSIS 2010

Reviewers

Professor Juha Röning
University of Oulu
Department of Electrical and Information Engineering
P.O. Box 4500
FI–90014 University of Oulu
Finland

Professor Arto Visala
Aalto University School of Science and Technology
Department of Automation and Systems Technology
P.O. Box 15500
FI–00076 Aalto
Finland

| **Julkaisija** | **Julkaisuajankohta** | |
|---|---|---|
| Vaasan yliopisto | Maaliskuu 2010 | |
| **Tekijä(t)**<br>Janne Koljonen | **Julkaisun tyyppi**<br>Artikkelikokoelma | |
| | **Julkaisusarjan nimi, osan numero**<br>Acta Wasaensia, 219 | |
| **Yhteystiedot**<br>Vaasan yliopisto<br>Teknillinen tiedekunta<br>PL 700<br>65101 VAASA | **ISBN**<br>978–952–476–295–3 | |
| | **ISSN**<br>0355–2667, 1798–789X | |
| | **Sivumäärä**<br>209 | **Kieli**<br>Englanti |
| **Julkaisun nimike**<br>Konenäkö- ja optimointimenetelmiä tasomaisten kappaleiden muodonmuutosten mittaamiseen | | |



**Tiivistelmä**

Venymien ja muodonmuutosten mittaaminen on tärkeää niin materiaalitekniikan laboratoriotutkimuksissa kuin teollisuustuotteiden lujuusmittauksissa ja laadunvarmennuksessa. Yleisesti käytetyt mekaaniset venymämittausmenetelmät ovat puutteellisia erityisesti spatiaalisen resoluution ja joskus luotettavuudenkin suhteen. Venymämittauksiin on kehitetty monia optisia menetelmiä, joilla saadaan mekaanisia menetelmiä tarkempaa ja monipuolisempaa tietoa kappaleen muodonmuutoksista. Kullakin menetelmällä on omat vahvuutensa ja heikkoutensa liittyen mittaustarkkuuteen, spatiaaliseen ja temporaaliseen resoluution sekä laitteiston, mittausjärjestelyiden ja näytteen esikäsittelyyn.

Tutkimuksessa tarkastellaan menetelmiä, joilla pyritään parantamaan tasomaisten näytteiden satunnaiseen kuviointiin, konenäköön ja kuvien kohdistamiseen perustuvaa venymämittausmenetelmää, joka soveltuisi sekä tutkimuskäyttöön että automaattiseen laadunvalvontaan. Lisäksi tutkitaan kolmea menetelmää venymämittausten tarkkuuden estimointiin. Kuvien kohdistamisen laskennallista tehokkuutta on parannettu kehittämällä heuristiikka kuvien kohdistamisessa käytettävän korrelaatioikkunan koon säätöön. Venymämittausohjelman parametreja optimoitiin käyttäen evoluutioalgoritmeja ja kaksitavoiteoptimointia, jossa pyrittiin samanaikaisesti parantamaan sekä mittaustarkkuutta että laskentanopeutta. Evoluutioalgoritmit mahdollistivat myös käytettävän kuvien kohdistukseen moniulotteista venymämallia, jonka hyvyysmaisema on multimodaalinen.

Erityisesti ikkunansäätöheuristiikka ja ohjelmaparametrien optimointi antoivat huomattavan hyviä tuloksia. Kummallakin menetelmällä pystyttiin parantamaan kuvien kohdistamisen tarkkuutta ja samalla pienentämään kuvien kohdistamiseen tarvittavaa laskenta-aikaa. Tarkkuuden estimointimenetelmistä hyödyllisiksi osoittautuivat implisiittinen, venymien hajontaan perustuva estimaatti ja keinotekoisten venytyskuvien luontiin perustuva menetelmä.



**Asiasanat**

Evoluutioalgoritmit, konenäkö, materiaalitekniikka, mittaustekniikka, optimointi.

| **Publisher** | **Date of publication** | |
|---|---|---|
| Vaasan yliopisto | March 2010 | |
| **Author(s)**<br>Janne Koljonen | **Type of publication**<br>Selection of articles | |
| | **Name and number of series**<br>Acta Wasaensia, 219 | |
| **Contact information**<br>University of Vaasa<br>Faculty of Technology<br>P.O. Box 700<br>FI–65101 VAASA<br>FINLAND | **ISBN**<br>978–952–476–295–3 | |
| | **ISSN**<br>0355–2667, 1798–789X | |
| | **Number of pages**<br>209 | **Language**<br>English |


**Title of publication**

Computer Vision and Optimization Methods Applied to the Measurements of In-Plane Deformations


**Abstract**

Measurements of strains and deformations are important in experimental materials engineering as well as in tensile testing and quality control of industrial goods. The common mechanical methods that are used to measure elongations are sometimes insufficient as for spatial resolution and reliability. There are many optical methods that give more accurate and detailed information on the strain fields. Each method has different characteristics as for accuracy, spatial and temporal resolution, and the complexity of the experimental setup and sample preparation.

In this doctoral thesis, methods to improve a computer vision approach to measure in-plane deformations are studied. Deformations are measured from a randomly speckled sample using image registration, and the developed software could be applied both to research and industry. Furthermore, three methods to estimate the accuracy of the strain measurements are studied. The computational complexity of the image registration is reduced by a heuristic that controls the size of the template image dynamically. The user parameters of the measurement software are optimized using evolutionary algorithms and two-objective optimization, whence both accuracy and computational efficiency can be improved concurrently. Evolutionary algorithms are also used to search for the optimal parameters of a strain field model with a multidimensional and multimodal fitness landscape.

In particular, the dynamic template size control and the optimization of the software parameters gave promising results. Both methods improved the accuracy-complexity ratio significantly. An implicit error estimate that is based on the statistical analysis of the strains and a method to artificially generate test images proved to be useful when estimating the accuracy of the strain measurements.



**Keywords**

Computer vision, evolutionary algorithms, materials engineering, measurement technology, optimization.

# PREFACE

In 2002, while I was just a second-year student of electrical engineering, Erkki Lähderanta, today a Professor of physics at Lappeenranta University of Technology, asked me to work as a course assistant on his courses on basic physics for engineering students. There were also plans that I would make my master's thesis in his research group. At some point he mouthed that I would probably write my doctoral thesis before the age of 30. I have born that goal in mind ever since.

Later the same year, however, a post in automation technology became vacant during the semester, and someone was needed to attend it as soon as possible. In an unselfish way, Lähderanta introduced me to Professor Jarmo Alander who was open-minded to hire me, even though I had only little experience on signal processing, digital electronics, evolutionary computation, and the other subjects that automation technology covers at the University of Vaasa. The post naturally made me change my major from electrical engineering to automation technology, and I graduated in 2004.

I am thus very grateful to Erkki Lähderanta for initializing my interest in an academic career and the help in the beginning of it. I should also thank Timo Vekara, Professor of electrical engineering, for understanding the choice to change my major and for the support thereafter in my academic career.

Professor Jarmo Alander, the supervisor of my thesis and a mentor of my academic career, naturally deserves the greatest acknowledgment. During these seven and a half years, we have devised numerous technical inventions, new methodological concepts (unfortunately, only few of them have been implemented so far), and written over twenty scientific articles.

In addition to Prof. Alander, the other co-authors of the articles of this thesis are: Timo Mantere, Olli Kanniainen, Tuomas Katajarinne, and Annette Lönnqvist. I am grateful for their help in making measurements and improving the manuscripts as well as their flexibility in schedules and preparing experiments.

The main part of the study was done in a three-year research project called *Process Development for Incremental Sheet Forming*. The project was funded by the Finnish Funding Agency for Technology and Innovation (TEKES) and industrial partners, and it was run in years 2006–2008. The financiers are acknowledged for the support. Moreover, I would like to thank the research partners from Helsinki University of Technology (TKK): Jukka Tuomi (research manager of the project), Professor Seppo Kivivuori, Lotta Vihtonen, and Tuomas Katajarinne, for

the fruitful collaboration and pleasant atmosphere in the meetings, excursions, seminars, and conferences, which we participated together.

The objectives of the project were, in short, to study the deformation model occurring in incremental sheet metal forming (ISF), to model and simulate the deformation process, to study and simulate the properties of the object obtained by ISF, and to study the feasibility of adaptive control of ISF by computer vision and FEM simulation. In order to tackle these goals it was needed to improve the material test methods used in laboratory and to outline a measurement setup applicable to online measurement in ISF.

The role of the University of Vaasa was to develop the methods of computer vision according to the requirements of TKK, which in turn built a robot workstation where ISF and its deformation process were studied. At the final year of the project, a computer program called DeforMEERI, which was developed in the project, was applied to study the properties of objects formed by ISF in order to contribute to the objectives related to mathematical modeling of ISF.

During the project, industrial partners showed interest to apply DeforMEERI to industrial materials tests in quality control and research. Hence, more focus was given to improve the computational efficiency and usability of the software.

I am grateful to Juha Tulonen and his colleagues from Rautaruukki that the strain analyzer could be tested at the testing laboratory of Rautaruukki in Hämeenlinna. Additionally, Juha Tulonen is acknowledged for the references and the expert information on the conversions of elongation values, Raimo Ruoppa from Outokumpu for the references regarding the commercial strain analyzers and for the invitation to test our analyzer at Outokumpu, and, Pertti Lehto from T-Drill for introducing the research problems in the pipe cutting and forming machines.

The following foundations are acknowledged for the personal grants: the Institute of Technology Research of the University of Vaasa (TTI), the Finnish Cultural Foundation, and Vaasan Yliopistoseura [the society of the University of Vaasa].

I am most grateful to my lovely wife Outi for her support and encouragement.

Vaasa, Finland, January 22, 2010

Janne Koljonen

# Contents

# Figures

# Tables

# Abbreviations

| | |
|---|---|
| 2D | Two-dimensional |
| 3D | Three-dimensional |
| AI | Artificial Intelligence |
| ANN | Artificial Neural Network |
| BFGS | the Broyden–Fletcher–Goldfarb–Shanno method |
| CAD/CAM | Computer Aided Design/Manufacture |
| CCD | Charge-Coupled Device |
| CGA | Cultural Genetic Algorithm |
| cGA | cellular Genetic Algorithm |
| CPU | Central Processing Unit |
| CT | Computed Tomography |
| DG | Discrimination Gap |
| DIC | Digital Image Correlation |
| dpi | dots per inch |
| DPIV | Digital Particle Image Velocimetry |
| DTSC | Dynamic Template Size Control |
| EA | Evolutionary Algorithm |
| EP | Evolutionary Programming |
| ES | Evolutionary Strategy |
| ESPI | Electronic Speckle Pattern Interferometry |
| FEM | Finite Element Model |
| FPGA | Field Programmable Gate Array |
| GA | Genetic Algorithm |
| GP | Genetic Programming |
| HI | Holographic Interferometry |
| HSB | Hue-Saturation-Brightness |
| ISF | Incremental Sheet Forming |
| MA | Memetic Algorithm |
| MC | Maximum Correlation |
| ML | Maximum Likelihood |
| MRI | Magnetic Resonance Imaging |
| PC | Personal Computer |
| pixel | picture element |
| PSD | Position-Sensitive Detector |
| RGB | Red-Green-Blue (color model) |
| RMS | Root Mean Square |
| RMSE | Root Mean Square Error |
| RMSEC | Root Mean Square Error of Calibration |
| RMSEP | Root Mean Square Error of Prediction |
| ROI | Region Of Interest |
| SEM | Scanning Electron Microscope |
| SLR | Single Lens Reflex (camera) |
| SNR | Signal to Noise Ratio |
| SSD | Sum of Squared Difference |
| STM | Scanning Tunneling Microscope |

| | |
|---|---|
| TEKES | Finnish Funding Agency for Technology and Innovation |
| TEM | Transmission Electron Microscope |
| TKK | Helsinki University of Technology |
| USB | Universal Serial Bus |
| voxel | volume element |
| XOR | eXclusive OR |

## List of publications

This thesis consists of an introductory part and seven published articles. The bibliographic data of the articles reprinted in this thesis are as follows:

I Koljonen, J., Kanniainen, O. & Alander, J. T. (2007a). An implicit validation approach for digital image correlation based strain measurements. In *Proceedings of the IEEE International Conference on Computer as a Tool*. Warsaw, Poland: IEEE. 250–257.

II Koljonen, J., Kanniainen, O. & Alander, J. T. (2007b). Dynamic template size control in digital image correlation based strain measurements. In D. P. Casasent, E. L. Hall & J. Röning (eds.). *Intelligent Robots and Computer Vision XXV: Algorithms, Techniques, and Active Vision*. Boston, USA: SPIE. 67640L-1–12.

III Koljonen, J., Mantere, T., Kanniainen, O. & Alander, J. T. (2007). Searching strain field parameters by genetic algorithms. In D. P. Casasent, E. L. Hall & J. Röning (eds.). *Intelligent Robots and Computer Vision XXV: Algorithms, Techniques, and Active Vision*. Boston, USA: SPIE. 67640O-1–9.

IV Koljonen, J., Mantere, T. & Alander, J. T. (2007). Parameter optimization of numerical methods using accelerated estimation of cost function: a case study. In M. Niskanen & J. Heikkilä (eds.). *Proceedings of the Finnish Signal Processing Symposium* [CD-ROM]. Oulu, Finland: University of Oulu.

V Koljonen, J., Katajarinne, T., Lönnqvist, A. & Alander, J.T. (2008). Validation of digital speckle correlation strain measurements with extensometer. In N. Asnafi (ed.). *Best in Class Stamping. Proceedings of the International Conference of International Deep Drawing Research Group*. Olofström, Sweden: IDDRG. 57–68.

VI Koljonen, J. (2008). Partially separable fitness function and smart genetic operators for area-based image registration. In T. Raiko, P. Haikonen & J. Väyrynen (eds.). *AI and Machine Consciousness. Proceedings of the 13th Finnish Artificial Intelligence Conference*. Espoo, Finland: Finnish Artificial Intelligence Society. 4–14.

VII Koljonen, J. & Alander, J. T. (2008). Deformation image generation for testing a strain measurement algorithm. *Optical Engineering* 47:10. 107202-1–13.

All these articles have passed a referee procedure prior to their acceptance in publication. Publication VII appeared in an international scientific journal, publications I, II, III, and V in proceedings of international conferences, and publications IV and VI in proceedings of national conferences.

The articles are reprinted unchanged at the end of this thesis, with the courtesy permission of their original publishers. The known significant errors of the articles are reported before the reprints (p. 118). The reprinted articles form an integral part of this thesis and they constitute an entity with the introductory part as for the contributions of this thesis.

The included articles start at the following pages of this thesis:

# 1 INTRODUCTION

Economical growth and the increase in productivity are largely due to the increasing degree of automation in production and services. Automation technology can be used to raise the productivity of human labor or, in some cases, even to replace it. Automation technology also enables realization of novel applications that were otherwise impractical or, at least, infeasible.

An important branch of artificial intelligence (AI) is computer vision. It is a discipline where the information from images is used in intelligent systems. Related fields are, e.g., image processing, machine vision, and measurements based on image.

In image processing, the objective is to transform an image using, e.g., pixel operations, filtering, and geometrical transformations. Machine vision in turn refers to industrial applications where vision and real-time processing are used, e.g., to control robots or in inspection. However, the use of the nomenclature and the distinctions between these categories are not established. Different terms are also used mixed in this thesis without implying any specific distinction.

Measurements based on image could be regarded as a subcategory of computer vision. Term *measurements based on image* is preferred in this thesis to emphasize the objective and the output of the computer vision task.

The potential of applying computer vision to deformation measurements in materials engineering is studied in this thesis. A software called DeforMEERI and test methods to evaluate it are introduced. Heuristics, e.g., evolutionary and genetic algorithms, are used to improve the accuracy, computational performance, and usability of the software.

## 1.1 Background and motivation

Deformation measurements are needed to find out the macroscopic properties of materials. They are needed for both laboratory and production tasks.

Laboratory experiments are used, e.g., to obtain input values to a finite element simulation model (FEM), to validate that a given metal part meets its specifications as for tensile strength, and to study how the properties of a metal part have changed during a forming process. An example of the latter is the research concerning the forming process of *incremental sheet forming* (ISF), which is a rapid prototyping method where a metal sheet is formed by applying a force onto a sin-

gle point or a small area at a time using, e.g., a robot tool (see, e.g., Vihtonen, Tulonen & Tuomi 2008 and Vihtonen, Puzik & Katajarinne 2008 for details about the concept of ISF). In mass-production, deformation measurements are used, e.g., to validate that the tensile strength of the rolled metal sheets meet their specifications.

Mesh grids and mechanical extensometers are commonly used in the materials tests to obtain strain paths and distributions. For example, both mesh grids and a mechanical extensometer were used to study large deformations occurring in metal-forming processes in (Leung et al. 2004).

Strains based on the deforming mesh grid can be determined optically using imaging. However, the mesh grid method requires accurate equipment for sample preparation, and the spatial grid frequency has to be selected *a priori*. Leung et al. (Ibid.) also noticed that grids were difficult to identify next to the rupture. By using an optical extensometer and a random speckle pattern this problem can partly be avoided, because measurements are not limited to some predefined grid points but any point of the specimen can *a posteriori* be selected as a grid point.

Optical extensometers contain only few or no moving parts. Hence, they are presumed to be superior to the mechanical ones, which may be inaccurate and unreliable due to mechanical wear and inaccurate target tracking, at least according to the marketing material of Instron®, a manufacturer of materials testing machines and accessories (Instron; Instron 2005). In scientific literature, too, the attachment of the extensometer knife-edges has been claimed unreliable. Cotton et al. (2005) reported that slippage of the extensometer legs might have produced unrealistic results in seven out of 36 cases in a fatigue test.

Strain measurements by mechanical extensometers are limited to a predefined gauge length. Thus the information of the spatial deformation field is inadequate. Moreover, the position of the rupture is not known *a priori*, and sometimes the rupture occurs outside the prescribed gauge of the knife-edges causing laborious re-measurements.

Because the deformation measurements are common in research and production, and because the common methods are, in some aspects, laborious, unreliable, and inadequate, a method based on random speckle and computer vision was developed. Although commercial equipment for optical strain analysis also exist (GOM GmbH; LaVision GmbH; ViALUX GmbH), they are not widely used, probably due to their rather high price of tens of thousands of euros (Personal communication with R. Ruoppa from Outokumpu Oyj).

## 1.2 Authors' contributions to the publications

The ideas and scientific contributions discussed in this thesis and in the reprinted articles were invented and developed primarily by the author of this thesis (J. Koljonen). J. Koljonen designed and implemented all novel algorithms and analyzed all results. In addition, J. Koljonen is the principal author of all the articles included. The roles of the co-authors are described in what follows.

Professor Jarmo T. Alander (University of Vaasa) acted as the supervisor of the research and this thesis, and he was a co-author in publications I, II, III, IV, V, and VII. In the interactive supervising process, Prof. Alander had a major role in the early stage of planning the research topic and objectives as well as in establishing the research network of the research project. As a co-author, he proof-read the manuscripts and suggested corrections and improvements. He also looked through the literature for additional references to relevant related work.

Olli Kanniainen (University of Vaasa) was a co-author in publications I, II, and III. His main contribution was to assist in the experiments in the early part of the research project. Timo Mantere (University of Vaasa) was a co-author in publications III and IV. He contributed to the scientific work concerning the use of genetic algorithms. In particular, he discussed the ideas of J. Koljonen related to multi-objective optimization and named relevant publications.

Tuomas Katajarinne and Annette Lönnqvist from TKK were co-authors in publication V. Lönnqvist carried out the experimental tests using the facilities of TKK. Katajarinne in turn contributed to the text of the publication by providing information of the equipment used in the experiments. Moreover, he discussed the potential sources of errors, which emerged in the results of the experiments, with J. Koljonen.

According to the contract of the research consortium, the articles were inspected by the partners of the consortium prior to publication. However, each paper was accepted by the partners without requirements of modification.

## 1.3 Objectives and contributions

According to the framework of the research project, the objective of this study was *to develop methods and computational algorithms, with which deformation fields of planar objects could be measured fast and accurately using measurements based on image*. The measurement setup should be implemented using inexpensive off-the-shelf components. Furthermore, the software should utilize au-

tomation as far as possible for easy usability and applicability to automatic online measurements.

Several research questions emerged:

1. Can an accurate, fast, and easy-to-use optical extensometer superior to the mechanical extensometer be developed?
2. Which approach of nonrigid body image registration meets the requirements set for the method best?
3. Can compromises between computational complexity, accuracy, and resolution of the strain measurements be avoided?
4. How the accuracy of the deformation measurements can be evaluated?

In short, these research questions were tackled, at least, with partial success. The first question was studied by developing and comparing two approaches. Novel methods to reduce complexity without sacrificing accuracy were developed with surprisingly positive results. Three methods to evaluate accuracy were devised and tested.

The main contributions of this thesis are the following:

- an implicit method to estimate the accuracy of strain measurements,
- a method to generate realistic artificial test images for testing purposes of strain measurement algorithms based on image registration,
- a dynamic window size control method for template matching,
- accelerated optimization of the parameters of algorithms and programs, and
- a variety of evolutionary algorithms, genetic operators, and fitness functions to search for the parameters of a deformation field.

## 1.4 Structure of the thesis

This thesis consists of two parts: an introductory part with references and seven publications reprinted in their original form at the end of this thesis.

Chapter 1 introduces the topic in short, motivates the need of optical deformation measurements, and names the objectives and contributions of this thesis. Chapter 2 deals with the basics of computer vision that are needed to master when carrying out accurate measurements based on image. These include geometric image transformations, template matching, sub-pixel image registration, parameterized camera models, and camera calibration. Because camera calibration was not included in the publications, it is dealt in detail in the Section 2. In addition to the reviews based on literature and the attached publications, Section 2 introduces

some experiments and results. For instance, some customized features of the camera calibration procedure are introduced. They are included here to clarify and complement the discussion on the concepts used in the publications.

In Chapter 3, the literature related to optical strain measurements is reviewed. Moreover, the computer vision methods, i.e., experimental setups and computational algorithms, that are used to measure strains during uni-axial tensile tests are introduced and discussed. The concept of 'Conversion of elongation values' that was excluded in the publications is included in Chapter 3.

Chapter 4 gives a short general introduction to evolutionary algorithms (EAs) and related methods. Concepts of multi-objective and *Pareto* optimization and their application to the optimization of the software parameters of DeforMEERI are introduced. Moreover, the challenges related to the multimodality and the real-valued variables of a third-order displacement model are studied and discussed.

Chapter 5 presents each of the reprinted articles briefly. Conclusions of the study and this thesis are finally drawn in Chapter 6, after which the reprints of the articles start.

# 2 MEASUREMENTS BASED ON IMAGE

In measurements, attributes of objects are estimated and expressed usually numerically. The attributes can be physical, such as length, but also, e.g., economical. Measurements are done using an instrument, which may be physical, such as a camera or an economical survey (Bureay International des Poids et Mesures).

The capabilities of human beings are often used as benchmarks and source of inspiration in science and technology. Computer vision is an undisputable example of such a discipline. In many cases, human vision is still superior to the computer vision due to the flexibility and learning capabilities of human beings. Nevertheless, in metric image based measurements, computer vision usually outperforms the human visual system.

In measurements based on image, one or several, nowadays usually digital, cameras, or other sensors, are utilized. The information of the physical properties of the objects is transmitted to the sensors, e.g., by electromagnetic, electron, or sonar waves. The sensors transform the quanta usually into a 2D grid of digits that form an image.

The images are used to make interpretations of the physical world. Typical attributes that can be measured using images are metric dimensions, location, and velocity, or, e.g., the color of the object. Images are in many respects imperfect representations of the physical world. They are usually distorted 2D projections of the 3D world, and they include noise. Therefore, the range of applications and the accuracy of the measurements based on image are restricted.

Many shortcomings of the measurements based on image can be overcome, e.g., using several cameras, camera calibration, and more complex computation. The advances in computer vision algorithms and the rapid increase in the computational power of computers have indeed made many applications feasible only just recently.

Measurements based on image have many advantages. With camera calibration, accurate metric measurements with good repeatability can be done. With pattern recognition, stochastic or other landmarks can be detected quite reliably, whence, e.g., the deformation of an object can be measured without an expensive or labor intensive marking process. On the other hand, measurements can be sensitive to disturbances, such as changes in illumination, vibration, dirt, etc.

This chapter deals with the basics of computer vision that are needed to master when carrying out accurate measurements based on image. In particular, theory

and practice that are needed when developing methods for nonrigid body registration and measurements of deformation are gone through. These issues are rigid body image registration, subpixel registration, and camera calibration.

## 2.1 Image registration

Image registration is the process of aligning the coordinate systems of different images. The image with the reference coordinate frame is called *base image*, or reference image, and the image whose coordinate transformation to the reference frame is solved is called *input image*, or target image (Hajnal, Hill & Hawkes 2001). Image registration using digitalized images and computers has been used, at least, since the late 1960's (Anuta 1969; Anuta 1970).

Although the main objective is to solve the relation between the coordinate frames, the image of one coordinate system can also be transformed and re-sampled to the other coordinate system, but this is not necessary in all image registration applications. Re-sampling may require interpolation of intensities, which causes interpolation noise. Thus sequential transformations should be avoided and the original image should be used instead (Hajnal, Hill & Hawkes 2001).

As a result of image registration, the information of the aligned images can be integrated, the corresponding features of objects are easily related, and thus the images can be compared and the changes measured more easily (Ibid.).

The images to be registered may be acquired from different perspectives, using different imaging modalities, or using the same sensor at different times (Ibid.). These three categories of image registration are referred as *multiview*, *multimodal*, and *multitemporal analysis*, respectively (Zitova & Flusser 2003). Different imaging modalities are, e.g., near infrared imaging, X-ray computed tomography (CT), and magnetic resonance imaging (MRI).

Typical disciplines and applications that use image registration are, e.g., three-dimensional computer vision (Faugeras 1993), remote sensing (Schowengerdt 2007), medical imaging (Hajnal, Hill & Hawkes 2001; Suetens 2002), and optical surveillance systems (Hu et al. 2004), and also strain measurements.

In stereo vision, the target is imaged from different perspectives using two or more cameras. Alternatively, multiple viewing angles with a single camera can be used. After relating the corresponding points of interest of the target, three-dimensional measurements and reconstruction can be obtained using triangulation (Faugeras 1993; Sonka, Hlavac, & Boyle 2008).

In remote sensing, images from different areas are merged using image registration. Furthermore, changes, e.g., in natural resources and land usage are measured (Schowengerdt 2007). The process of merging images into larger images is known as *mosaicing*. Subpixel image registration and mosaicing can also be used to obtain *super-resolution* images, i.e., to up-sample images spatially (Capel & Zisserman 1998; Capel & Zisserman 2003).

In medical image registration, the combined information of images of different modalities is used to assist diagnostics. For example, a brain tumor can be imaged using both CT and MRI that produce 3D image cubes consisting of volume elements, *voxels*. Different modalities are sensitive to different chemical, anatomical, and pathological features, and thus the registration process is challenging. The images may also be compared to an *atlas*, i.e., an image of an average person (Hajnal, Hill & Hawkes 2001: 31). In general, this type of image registration is referred as *scene to model registration* (Zitova & Flusser 2003). Online image registration is nowadays used in medical operations (Shen et al. 2003).

In automatic optical surveillance systems, foreground objects are recognized and tracked. Moreover, the behavior of, e.g., people and vehicle can be analyzed. In general, background and foreground need to be detected and registered separately (Freer et al. 1997; Fuentes & Velastin 2006).

Image registration algorithms include, at least, the following basic parts: a transformation model to map points from one coordinate frame to another, an objective function to evaluate the coordinate transformation, and a search algorithm to optimize the objective function with respect to the transformation model. The image transformations are subdivided into rigid and nonrigid body transformations. The most common categories of objective functions are based on: the correspondence of homologous landmarks, correspondence of arcs, i.e., 2D features, or surfaces, i.e., 3D features, and similarity of intensities. The first category is known as feature based registration (Hajnal, Hill & Hawkes 2001; Zitova & Flusser 2003).

As for computation, image registration methods are modular. This means that distributed computation can be used, as suggested already, e.g., by Yang et al. (2007). There is also a great opportunity to use FPGAs in massive parallel computation in order to achieve real-time computation in complex image registration tasks (Jiang, Luk & Rueckert 2003; Sen et al. 2006; Sen et al. 2008).

The method of maximizing the similarity of intensities can be applied practically only when the images to be registered are of the same modality. The approach can be used both for traditional 2D images and for 3D voxel images. Ideally, the

aligned images should be identical for noiseless images or the difference image should follow the noise characteristics of the images. However, this method requires image transformation with re-sampling; thus at least interpolation noise occurs.

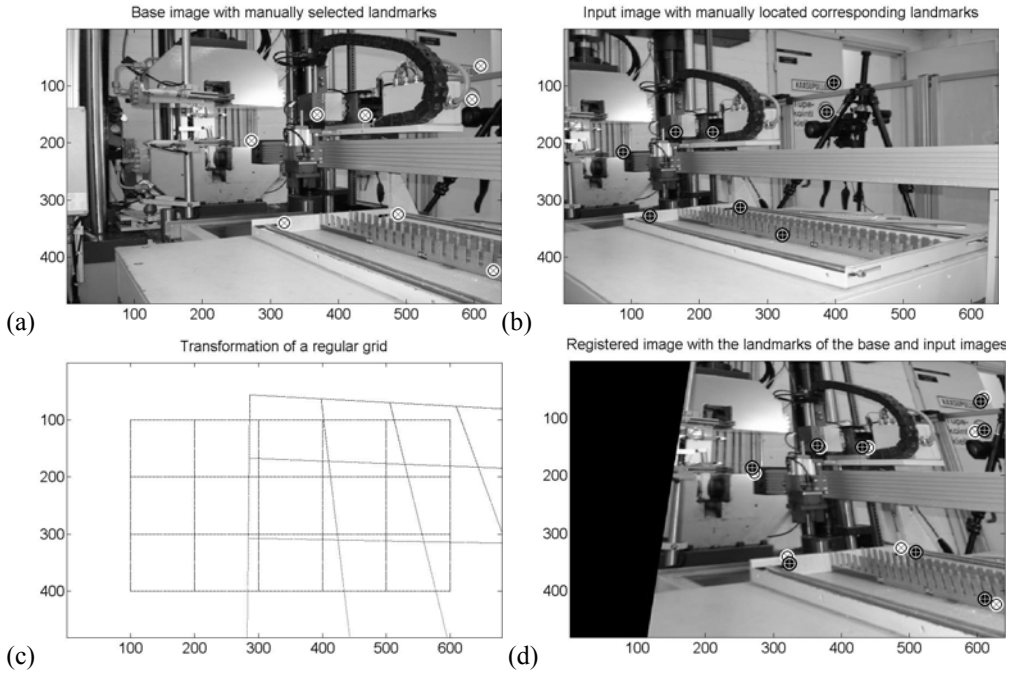



**Figure 1.** An example of applying feature-based image registration. (a) Base image. Eight landmarks are manually selected. (b) Input image. The same eight landmarks are detected and located manually. (c) The perspective transformation that minimizes the squared distance between the landmarks in (a) and the transformed landmarks in (b). The regular grid corresponds to the raster of (b). (d) Input image transformed using the transformation in (c). The registration error is visualized by the corresponding landmarks overlaid. It can be seen that the registration result is good but not perfect. The selection of the transformation space, i.e., perspective transformation, was not sufficient to capture the differences in the coordinate frames of the base and input images.

In order to establish a 2D rigid body transformation, i.e., translation, scaling, and rotation, between two 2D images it is sufficient to obtain two pairs of corresponding points at the base and input images. However, usually more points are used to average out positioning errors of the corresponding points or to eliminate outliers. The corresponding points are called, e.g., homologous or fiducial landmarks. The objective functions of this approach are based on the distances between the registered corresponding points. The square root of the mean of the squared distances,

i.e., the RMS error, is often used as the objective function (Hajnal, Hill & Hawkes 2001: 20–21).

Accurate and robust localization of the landmarks is crucial for successful image registration. Landmarks can be located, e.g., by selecting clearly recognizable structures or artificial markers on the object manually, by detecting salient interest points, such as closed-boundary regions or corners, or artificial markers automatically by specific algorithms, or by locating arbitrary points of the images using digital image correlation (DIC). An example of feature based registration is given in Figure 1. Another example can be found, e.g., in (Zitova & Flusser 2003).

Using DIC for landmark localization can be regarded as applying the method of 'maximizing the similarity of intensities' locally. This *single-point registration*, usually referred as template matching or object localization, is next dealt in more detail (Section 2.2).

## 2.2 Single-point registration using digital image correlation

The basic idea of template matching is to sample a, usually rectangular, template image from the base image, and to locate the same intensity pattern in the input image. When searching for the pattern, the template is subject to spatial transformations, such as translation, rotation, scaling, and shear (Zitova & Flusser 2003). In principle, the intensities of the template could also be modified, but this approach seems to be rare. However, at least Georgescu and Meer (2004) have used perspective transformations, color distribution matching, and illumination compensation in single point registration.

The similarity between the transformed template and the target sub-image is maximized using some search method. Alternatively, a distance measure is minimized. Template matching can also be performed in the Fourier space (Zitova & Flusser 2003).

Single-point registration can be presented as the following algorithm:

1. Initialize transformation, i.e., select initial values for the free parameters of the transformation.
2. Transform the template.
3. Evaluate the match criteria of the transformed template and the input image.
4. Check the stopping condition.
5. Update transformation parameters. Iterate from step 2.

Next methods to transform templates (Section 2.2.1), typical transformations (Section 2.2.2), match criteria (Section 2.2.3), and search methods (Section 2.2.5) are presented. Moreover, methods to apply subpixel accuracy in image registration are reviewed and discussed (Section 2.2.4).

### *2.2.1 General spatial image transformation*

In spatial image transformations, the intensities of pixels are preserved but their positions are changed. However, the intensities may also be changed due to interpolation. The transformations are in principle performed in the following steps:

1. Compute the new coordinates for the pixels. Use real numbers for the output coordinates. The result is an image, whose pixels form a nonuniform grid.
2. Interpolate the output image intensities to a uniform image raster using an appropriate interpolation kernel. The result is an image, whose intensities are real numbers.
3. Quantize the output image to obtain a digital image.

**Step 1:** Let the position of a pixel be $(x, y)$ in the coordinate frame of the base image. The translated position $(x', y')$ of the pixel is obtained, in general, by a vector function **T**:

$$\textbf{(1)} \qquad \begin{cases} x' = T_x(x, y) \\ y' = T_y(x, y). \end{cases}$$

Usually the output raster uses the same discrete coordinate system that the input image has. The translated pixels do not, in general, coincide with the image raster.

**Step 2:** In order to obtain the intensity values at the discrete raster positions, the intensities of the translated pixels are interpolated and re-sampled. Typical interpolation methods are: nearest neighbor, linear, and bi-cubic. For instance, in nearest neighbor interpolation, for each pixel of the output raster, the nearest translated pixel is searched for and assigned to that pixel. There are efficient algorithms to look for the nearest neighbors, but nevertheless, it may be computationally too slow a procedure.

A more efficient way to perform the intensity interpolation step is as follows:

First, compute the inverse transformation $(x, y) = \mathbf{T}^{-1}(x', y')$. The inverse transformation tells that the value of $(x', y')$ in the output image $f'$ is found on pixel $(x, y)$ in the input image $f$. In photography, the input image is originally a continuous function, from which the digital image is sampled. Thus the value of $f(x, y)$ is

known only for the discrete pixel positions. However, the intensities between the raster positions can be estimated by interpolation (Sonka 2008: 121). The two approaches for the intensity interpolation are illustrated in Figure 2.

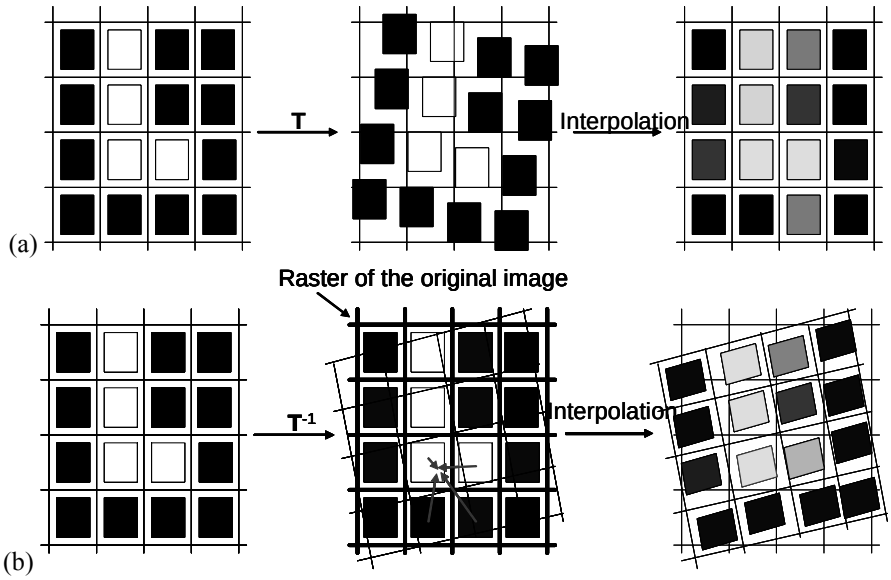



**Figure 2.** Two ways to interpolate intensities in geometrical image transformations. (a) Forward transformation **T** translates pixels that are interpolated to a regular image raster. (b) Inverse transformation $\mathbf{T}^{-1}$ transforms the raster of the output image. The original image is interpolated to obtain the values of the output image.

When using the inverse method for intensity interpolation the search of the nearest neighbors is almost trivial. For example, the nearest neighbor is found by rounding the real-valued coordinates ($x$, $y$) to the nearest integers (Ibid.: 122). Hence,

**(2)** $$f'(x', y') = f(\text{round}(x), \text{round}(y)).$$

In linear interpolation, four nearest points are used. Moreover, it is assumed that the intensity function changes linearly between the pixels. The four nearest pixels for ($x$, $y$) are: ($\lfloor x \rfloor$, $\lfloor y \rfloor$), ($\lfloor x \rfloor$, $\lceil y \rceil$), ($\lceil x \rceil$, $\lfloor y \rfloor$), and ($\lceil x \rceil$, $\lceil y \rceil$), where $\lfloor \cdot \rfloor$ and $\lceil \cdot \rceil$ refer to the floor and ceiling operators, respectively.

Nearest neighbor and linear interpolation usually give quite poor results with deteriorated resolution and image quality. A common interpolation kernel, which usually captures the intensities of the original continuous image more precisely, is the bi-cubic kernel. It uses 16 nearest neighbors and a 3rd order polynomial to model the intensities (Ibid: 123).

**Step 3:** In quantization, the real-valued intensities are truncated, for instance, to 8 bits. Quantization as well as interpolation induces noise to the transformed image. In particular, the quantization step can and should be avoided in some image processing techniques, where the transformed image is used only for computation, not for visualization.

### *2.2.2 Common spatial image transformations*

Next some common image transformations are reviewed: translation, scaling, rotation, affine transformation, and perspective transformation. All these transformations can be formulated by means of linear algebra, because the coordinate transformation is shared by all the pixels.

In homogeneous coordinates, the common transformations and their combinations are easily handled (Sonka 2008: 553–558). In homogeneous coordinates, points $(x, y, W)$ and $(x', y', W')$ coincide, if $x = ax'$, $y = ay'$, and $W = aW'$ for some $a \neq 0$. Usually the scale $a$ is selected so that $W = 1$. Hence, point $(x, y)$ in Cartesian coordinates maps to point $(x, y, 1)$ in homogeneous coordinates. The inverse mapping is $(x', y', W') \rightarrow (x'/W', y'/W')$.

Let $\mathbf{T}$ be a $3 \times 3$ matrix presenting a geometric, planar transformation in homogeneous coordinates. The transformation is:

$$\begin{pmatrix} x' \\ y' \\ W' \end{pmatrix} = \mathbf{T} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}. \tag{3}$$

#### *2.2.2.1 Translation, $\mathbf{T}_t$*

The most important and the most common, as for image registration, spatial image transformation is translation. In translation, each pixel is moved by a common, constant vector $\mathbf{T}_t = (t_x, t_y)$. In homogeneous coordinates:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \mathbf{T}_t \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}. \tag{4}$$

Translation is used, e.g., in pattern matching to ‘slide’ the template image over the target image. In addition, it is used in conjunction with other transformations, for example, to move the origin to a desired point, around which the image is to be rotated.

If the components of the translation vector are not integers, subpixel translations occur. Subpixel techniques are important, e.g., in image registration and camera calibration. Therefore, they are dealt separately in Section 2.2.4.

An example of translation is given in Figure 3.

(a) (b)

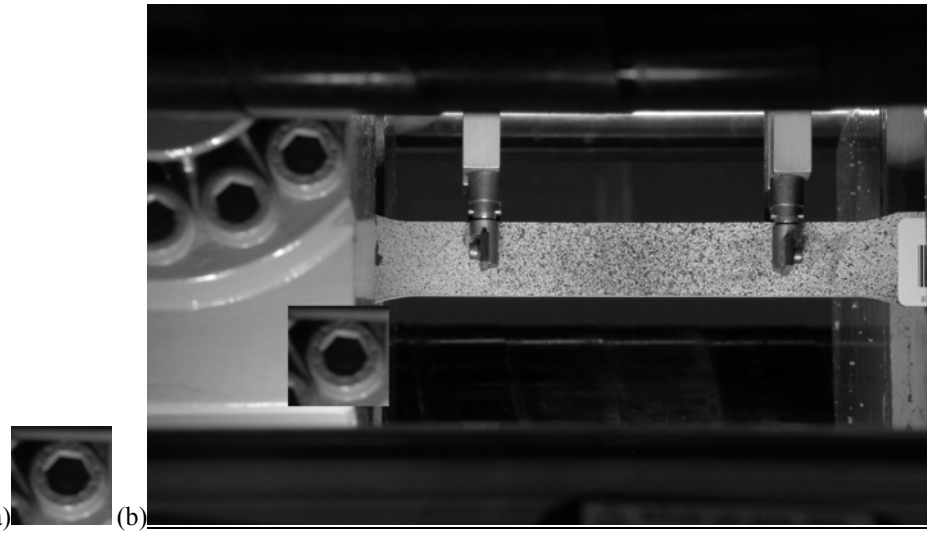


**Figure 3.** Translation. (a) Input sub-image cropped from a larger image. (b) Input sub-image translated to pixel (700, 1500) and overlaid in another image.

*2.2.2.2 Scaling,* $\mathbf{T}_c$

Scaling is used to change the height and width of an image. In image registration, it usually compensates for the change of distance, whence the aspect ratio is maintained. The aspect ratio may be changed in nonrigid registration, e.g., in optical strain measurements during uni-axial tensile tests. In general, scaling an image by factors $c_x$ and $c_y$, in width and height, respectively, can be done in homogeneous coordinates:

**(5)**
$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} c_x & 0 & 0 \\ 0 & c_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}.$$

Figure 4 shows two examples of image scaling. Now scaling is performed so that the dimensions of the input image are preserved. Normally, if the scaling factor was smaller than 1, the inverse transformation would suggest to look for the intensities outside the input image for some near-border pixels. If, instead of the

input image, a larger image and the coordinates of the corners of the input sub-image are supplied to the scaling algorithm, a size preserving scaling is possible.

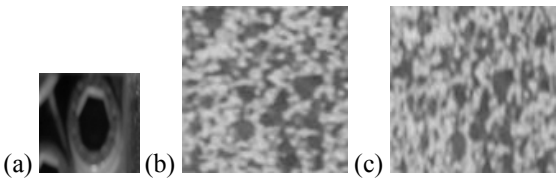



**Figure 4.** Scaling using bi-cubic interpolation. (a) Sub-image in Figure 3 (a) scaled with $c_x = 0.7$ and $c_y = 1.4$. (b) Input sub-image. (c) Image (b) scaled with $c_x = 0.7$ and $c_y = 1.4$.

*2.2.2.3 Rotation,* $\mathbf{T}_r$

In-plane rotation has only one free parameter, rotation angle θ. In homogeneous coordinates:

**(6)**
$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}.$$

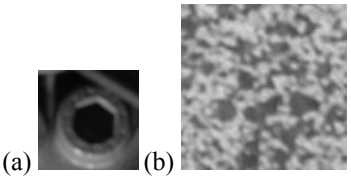



**Figure 5.** Rotation using bi-cubic interpolation. (a) Sub-image in Figure 3 (a) rotated 30° clock-wise (θ = –30°). (b) Sub-image in Figure 4 (b) rotated 30° clock-wise.

Figure 5 shows two examples of rotated images. The same size preserving procedure as for scaling is used.

*2.2.2.4 Affine transformation,* $\mathbf{T}_a$

Affine transformation is a more general geometrical transformation that combines linear transformations, i.e., scaling, rotation, and shear, with translation. In general, affine transformation preserves straight lines straight. An affine transformation is in general as follows:

$$\textbf{(7)} \qquad \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}.$$

Before determining the coefficients using translation, scaling, rotation, and shear, the missing basic transformation, shear, is studied. Shear $\mathbf{T}_s$ along x-axis is defined as follows:

$$\textbf{(8)} \qquad \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & s & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}.$$

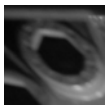
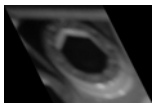
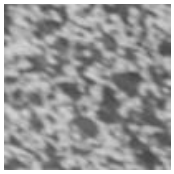
(a)  (b)  (c) 

**Figure 6.** Image shearing using bi-cubic interpolation. (a) Sub-image in Figure 3 (a) sheared using $s = 0.5$. (b) Sub-image in Figure 3 (a) sheared using $s = 0.5$ and zero-padding outside the sub-image. (c) Sub-image in Figure 4 (b) sheared using $s = 0.5$.

Examples of sheared image are shown in Figure 6. Panel (b), particularly, visualizes well what shear does to the images. Now, when $s = 0.5$, pixels are moved to the right by 0.5 pixels per image row. Shear can be used to compensate for small changes in the perspective angle.

Now a full affine transformation can be composed of the four basic transformations. In homogenous coordinates, the transformations are combined by multiplication of the transformation matrices. Because the matrix product is not commutative, the mutual order of the transformations affects the result. One possibility results in the following affine transformation:

$$\textbf{(9)} \qquad \mathbf{T}_a = \mathbf{T}_t \mathbf{T}_c \mathbf{T}_s \mathbf{T}_r = \begin{pmatrix} c_x \cos\theta + s c_x \sin\theta & -c_x \sin\theta + s c_x \cos\theta & t_x \\ c_y \sin\theta & c_y \cos\theta & t_y \\ 0 & 0 & 1 \end{pmatrix}.$$

Let us combine rotation, shear, scaling, and translation used in the previous examples. The affine transformation matrix (eq. 9) becomes:

$$\mathbf{T}_a = \begin{pmatrix} 0{,}7\cos(-30°) + 0.5 \cdot 0.7\sin(-30°) & -0.7\sin(-30°) + 0.5 \cdot 0.7\cos(-30°) & 700 \\ 1.4\sin(-30°) & 1.4\cos(-30°) & 1500 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0.43 & 0.65 & 700 \\ -0.7 & 1.21 & 1500 \\ 0 & 0 & 1 \end{pmatrix}. \quad \textbf{(10)}$$

The resulting transformation is visualized in Figure 7.

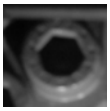
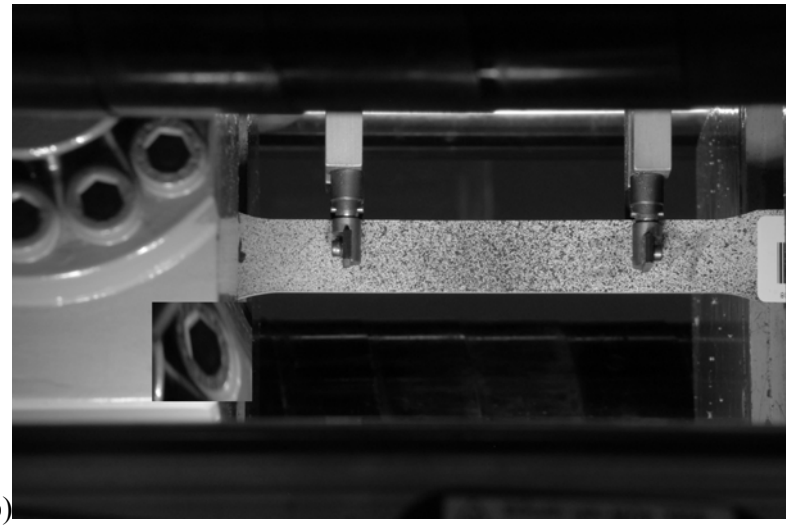
(a)  (b) 

**Figure 7.** Sub-image in Figure 3 (a) transformed using the affine transformation in eq. (10) and overlaid in another image.

*2.2.2.5 Perspective transformation*

Perspective transformations can be used to change the viewing angle of the observer with respect to an image or, more generally, a planar view. Equivalently, the place and orientation of the image can be altered. Perspective transformations are common, e.g., in computer graphics.

In general, any non-singular 3 × 3 matrix defines a planar perspective transformation in homogeneous coordinates:

$$\begin{pmatrix} x' \\ y' \\ W' \end{pmatrix} = \begin{pmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}. \quad \textbf{(11)}$$

Perspective transformation is used to model the image projection of pin-hole cameras. Usually it is needed in conjunction of camera calibration, which is dealt

in detail later in this thesis (see Section 2.3). A plane-to-plane perspective transformation can be obtained as follows:

1. Set the input image of the transformation as the scene of the camera.
2. Select the world coordinate system so that $z = 0$ for the input image.
3. Select a new viewing angle and distance: define the position of the principal point and the new image plane.
4. Project the input image to the image plane.

A perspective transformation can also be obtained by pairs of corresponding points in the reference and target images. At least four pairs of points that form a quadrilateral are needed (Kim, Jang & Hwang 2002). In Figure 1, this approach was applied to transform an image into another perspective angle. The result was obviously imperfect, because the scene was not planar as it was implicitly assumed in the transformation process.

### *2.2.3 Match criteria*

The third step in single point registration, i.e., the evaluation of the image similarity, is discussed next. In the previous section, typical geometrical transformations that can be applied to the image templates were reviewed. The most important transformation is translation when desired patterns are being localized from an image. Similarity evaluation usually requires that the transformed template and the target image share a raster of a common coordinate system.

Let $f$ denote the target image and $h$ the transformed template image. For simplicity of notation, it is supposed that both the width and height of $h$ are $M$ pixels and, furthermore, that the origin of $h$ is set to its top-left corner, i.e., translation is omitted in the transformation. Now a translation of $(t_x, t_y)$ pixels is achieved by shifting the coordinates of $f$ and, e.g., the maximum of pixel-wise differences of intensities between the target image and the translated template is computed as follows:

**(12)** $$D_1(t_x, t_y) = \max_{(x,y)\in h} \left| f(x + t_x, y + t_y) - h(x, y) \right|.$$

Actually, $D_1$ defines an optimality criterion for image matching. Ideally, $D_1 = 0$ for some translation vector, and for some combination of the other transformation parameters, and $D_1 > 0$ for others. Obviously, $D_1 = 0$ implies that the sub-images that are compared were identical, which is rarely the case. Hence, the objective is to minimize the *distance function* $D_1$.

$D_1$ is probably not very suitable for many applications, because changes even in one pixel are directly reflected to the distance measure, i.e., it is sensitive to outliers. Smoother and more common distance measures compute the absolute mean difference:

$$D_2(t_x, t_y) = \frac{\sum_{(x,y)\in h} \left| f(x+t_x, y+t_y) - h(x,y) \right|}{M^2} \tag{13}$$

or the RMS difference of the intensities:

$$D_3(t_x, t_y) = \sqrt{\frac{\sum_{(x,y)\in h} (f(x+t_x, y+t_y) - h(x,y))^2}{M^2}}. \tag{14}$$

One of the most popular similarity measures, together with $D_3$, is cross-correlation. Normalized correlation coefficient maps to [-1, 1]:

$$r(t_x, t_y) = \frac{\sum_{(x,y)\in h} (f(x+t_x, y+t_y) - \bar{f})(h(x,y) - \bar{h})}{\sqrt{\sum_{(x,y)\in h} (f(x+t_x, y+t_y) - \bar{f})^2 \sum_{(x,y)\in h} (h(x+t_x, y+t_y) - \bar{h})^2}}, \tag{15}$$

where $\bar{f}$ and $\bar{h}$ are the average intensities of the image region of $f$ that coincides with $h$ and the template, respectively, i.e.:

$$\bar{f}(t_x, t_y) = \sum_{(x,y)\in h} f(x+t_x, y+t_y) / M^2, \tag{16}$$

and

$$\bar{h}(t_x, t_y) = \sum_{(x,y)\in h} h(x,y) / M^2. \tag{17}$$

Contrary to the distance measures $D_1$, $D_2$, and $D_3$, cross-correlation is subject to maximization when determining point correspondence. The distance measures can be transformed to map to $C \in [0, 1]$, e.g., by the following simple transformation (Sonka 2008: 238):

$$C_i(t_x, t_y) = 1/(1 + D_i(t_x, t_y)). \tag{18}$$

There are also numerous other similarity measures. A common one is, e.g., mutual information (Viola & Wells III 1997). It is well known that correlation can be computed using Fourier transformation. A quite common registration method

called *phase correlation* uses Fourier transformation; it neglects the amplitude but uses only the phase information of the images (Foroosh & Zerubia 2002).

Some distance measures use segmented images, i.e., images whose pixels are grouped to foreground and background according to their tonal properties and relationships to other pixels. Registration methods based on segmentation have good properties, e.g., in situations, where local spatial distortions or changes in illumination occur. On the other hand, the robustness of the segmentation is crucial as for the success of the image registration. Matching methods that use segmented images are, e.g., the *chamfer* algorithm (Barrow et al. 1977; Burger & Burge 2008: 443–447) and the *Hausdorff* distance (Huttenlocher, Klanderman & Rucklidge 1993; Dubuisson & Jain 1994). The latter one is dealt next in more detail.

Let $f$ and $h$ be segmented images. Then the coinciding pixel regions can be considered as two sets of pixels, A = $\{a_1, \ldots, a_p\}$ and B = $\{b_1, \ldots, b_q\}$, respectively, where $p$ is the number of foreground pixels in the region of $f$ that coincides with $h$ and $q$ is the number of foreground pixels in template $h$. The Hausdorff distance computes the maximum distances between the nearest neighbors in A and B:

**(19)** $$D_4 = \max_{a \in A} \min_{b \in B} \|a - b\| .$$

If the template and the target image differ only by additive white Gaussian noise, the RMS distance (eq. 14) is the optimal distance measure as for minimizing the probability of false detection (Martin & Crowley 1995). However, if changes in the local mean intensity in image $f$ occur, the RMS distance may be inferior to the correlation coefficient (eq. 15) that is actually immune to the changes in the overall intensity level. Therefore, correlation coefficient is often preferred in computer vision applications.

Burger and Burge (2008: 432–435) show how these two similarity measures are related: nonnormalized cross-correlation and the sum of squared difference (SSD) would be, in principle, the same operation as long as the energy, or variance, of image $f$ is constant across the image. In practice, variance is rarely spatially constant.

Correlation coefficient is quite sensitive to outliers or occlusion of objects. Kim and Fessler (2004) have developed a robust correlation coefficient that tolerates for outliers. They modeled a maximum likelihood estimator for a distribution with heavy tails, i.e., with a great probability of outliers. Optimal weight coefficients are searched for in two cascade iterative processes.

#### *2.2.3.1 Correlation landscapes*

Next correlation landscapes with respect to different transformation parameters are computed and analyzed. Template size, one of the most important parameters, is also changed to demonstrate its effect on registration reliability and accuracy. Two sets of tests are done: first without and then with deformation. In both tests, the reference and target images are from a uni-axial tensile test specimen with a random speckle pattern.

In the first set of tests, the images differ only by added white Gaussian noise with $\sigma = 10$. The search area, i.e., the region of interest (ROI), of translations is $100 \times 100$ pixels. The template image and ROI of the noisy image are shown in Figure 8. Now the correct image transformation is known exactly *a priori*; there is no transformation and even the translation is a null vector.

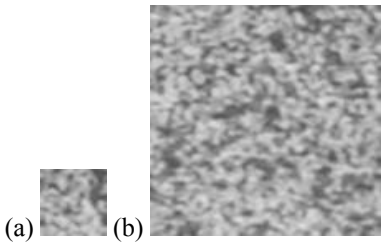



**Figure 8.** (a) The template image, i.e., the pattern to be searched from the region of interest. (b) ROI with added noise.

First the effect of the template size is studied. Only translation is applied to the template (Figure 8 a), and correlation coefficients are calculated. These correlation landscapes are shown in Figure 9. It can be seen that the larger the template is the more clearly the correct correlation peak is distinguished from the others. Too small a template evidently results in, more or less, unstable registration, depending on the type of the image and noise content. On the other hand, small template size implies a small computational cost. Traditionally, a trade-off between robustness and computational efficiency has been made. In Publication III, the compromise was avoided by introducing a dynamic control strategy to the template size parameter.

Figure 10 shows correlation landscapes when the template is scaled using different scaling factors. The translation is a constant null vector. The aim is to test how sensitive the correlation coefficient is if the scaling factors deviate from the correct ones, which are $c_x = 1$ and $c_y = 1$. At the same time, the results indicate how accurately the correct scaling can be estimated. In fact, the scaling factor, if accurately estimated, gives directly the amount of the local deformation, provided that

the local deformation field is constant (Vendroux & Knauss 1998). The effect of the template size is also studied.

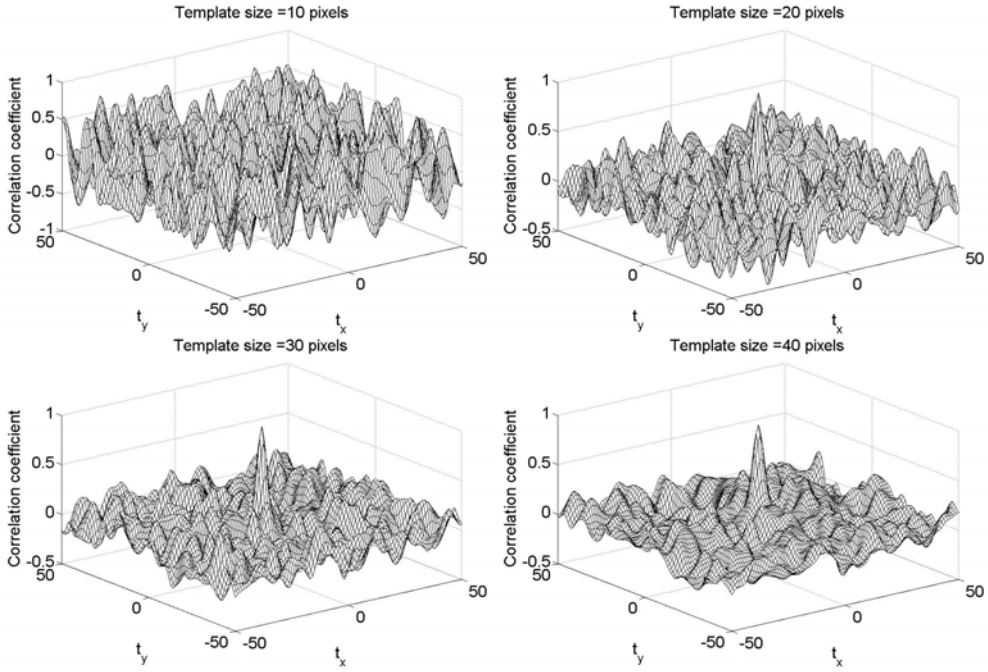



**Figure 9.** Correlation landscapes with respect to translation using different template size.

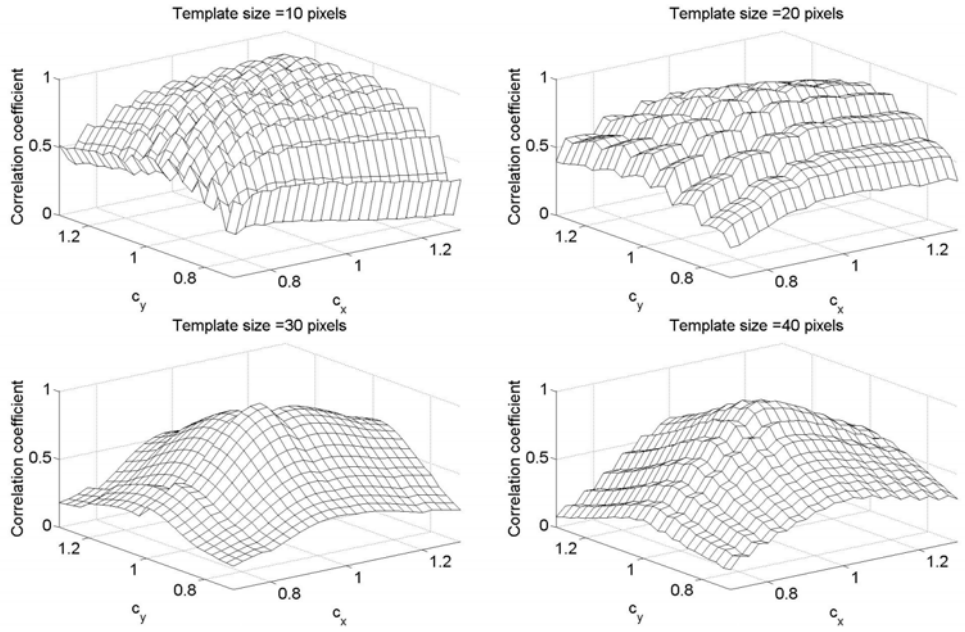



**Figure 10.** Correlation landscapes with respect to the scaling factors $c_x$ and $c_y$ using different template size.

The results indicate that the correct scaling can be, to some extent, estimated when the template size is large enough. With small templates the scaling landscape is rather flat and even rugged. Hence, there may be local optima that complicate the estimation of the scaling factors. If solely translation is to be estimated, the selection of the scaling factors is not very strict; small changes in scaling usually have a small effect on the correlation coefficient.

As a recommendation, if scaling is to be estimated, a larger template has to be used. On the other hand, if the deformation field is not locally constant, pure scaling may be insufficient. In that case, a more complex deformation model is needed. For instance, Lu and Cary (2000) have used second order polynomials to model the local deformation, while Cheng et al. (2002) have used B-splines, which guarantee the continuity of the derivatives.

In many registration applications, the target is subject to rotations, or the perspective view may change. Consequently, rotation and shear are applied to the template when determining point correspondence. A perspective transformation may be used, too. Tests indicate (Figure 11) that the shear parameter has only little influence on the correlation coefficient with any of the template size tested, whereas the selection of a correct rotation angle is important, regardless the template size.

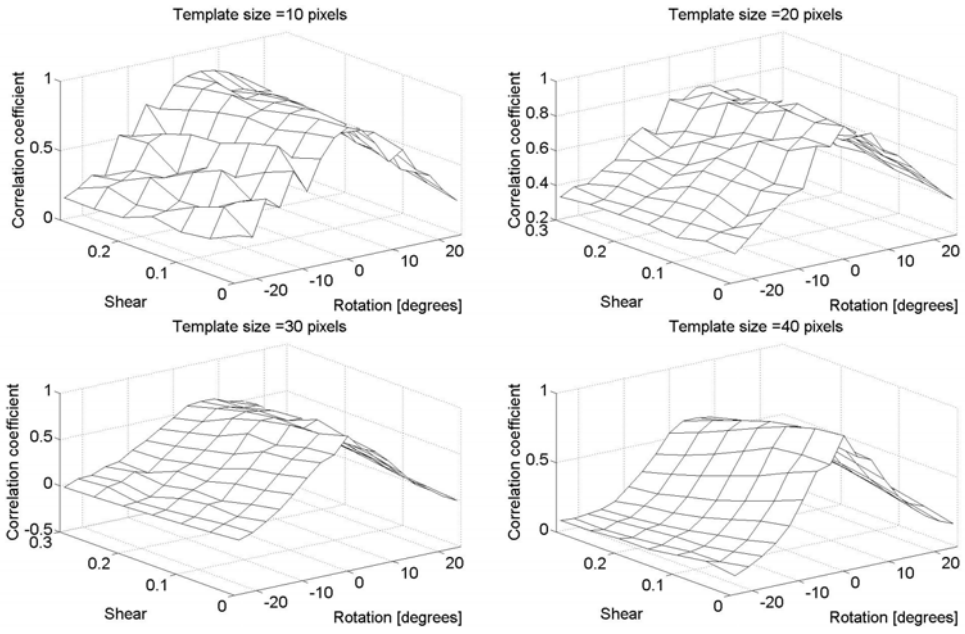



**Figure 11.** Correlation landscapes with respect to rotation and shear using different template size.

In the second set of tests, two images taken during a uni-axial tensile test at different strain states are used. Thus there is a deformation between the images. In addition, there may be noise and some changes in illumination. The template and ROI are shown in Figure 12. ROI was now 150 × 100 pixels.

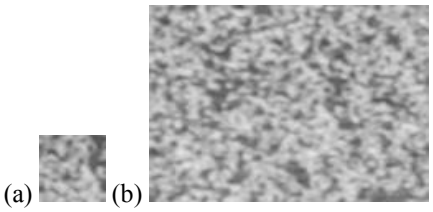



**Figure 12.** (a) Template image sampled from a reference image. (b) ROI cropped from the target with an unknown deformation field relative to the reference image.

Now it is studied how template size and scaling factors affect the correlation coefficient and the registration accuracy. The translation vector that gives the maximum correlation is searched for some combinations of the scaling factors. The corresponding maximum correlations are plotted in Figures 13 (a), (c), and (e). The most frequent translation vector, i.e., the mode, is supposed to be the correct registration position of the template. The Euclidian distances to the mode vector are calculated and plotted for every combination of the scaling factors (Figures 13 b, d, and f). This difference of translation can be regarded as the registration error.

It is interesting to see that the registration error is rather small regardless of the scaling factors and template size. It can be considered that only with a very bad scaling the template pattern was not found at all, i.e., the registration error was of several pixels. This observation suggests that it is not always necessary to scale the template in deformation measurements, which would bring great benefits as for the computational complexity.

The correlation landscapes support the observations based on Figure 10 that a larger template is needed if the scaling factors are to be estimated. Moreover, the correlation coefficient is more sensitive to changes in scale when a larger template is used, which is also intuitively evident: Suppose that the scale is changed from 1 to 1.1. If the template size is 20 pixels, the border pixels will be shifted by 1 pixel, whereas they are shifted by 2 pixels if the template size was 40. Hence, the spatial mismatch of pixels is greater with larger templates.

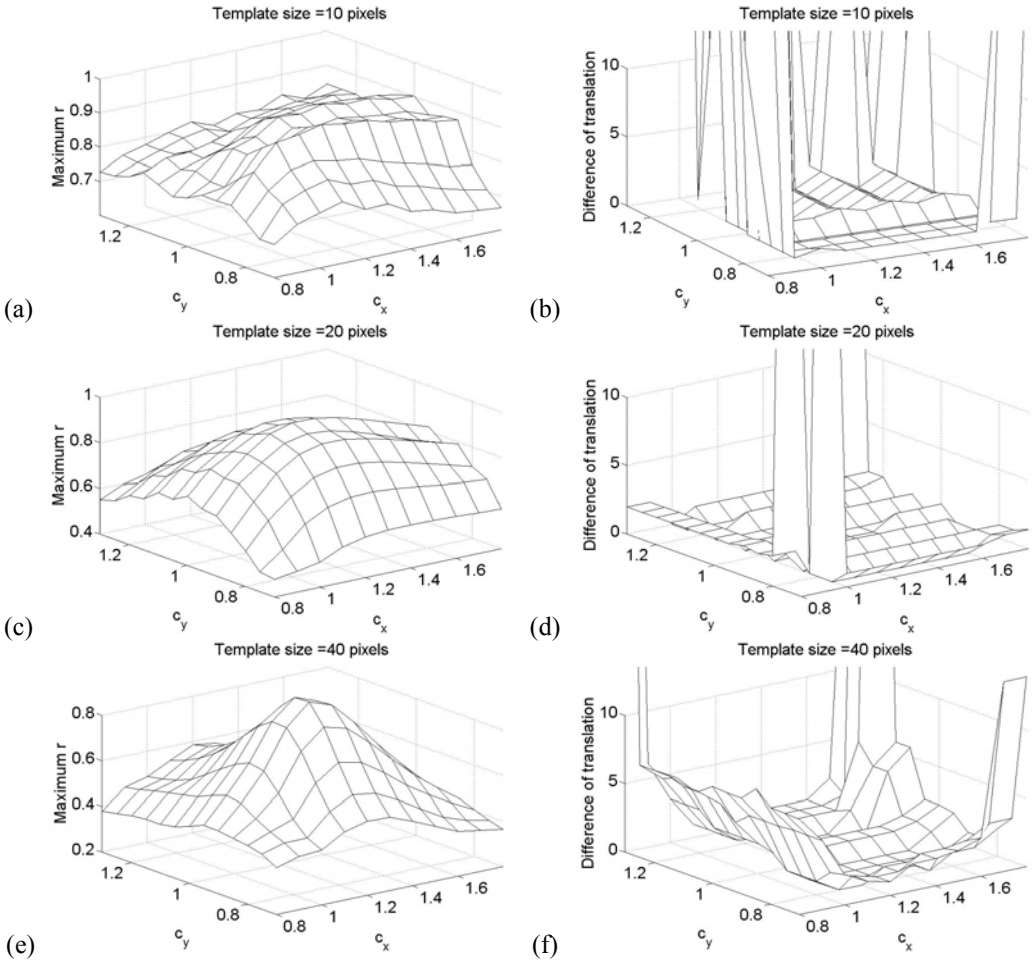



**Figure 13.** (a), (c), (e) Correlation landscapes with respect to the scaling factors when the target image is subject to deformation using different template size. (b), (d), (f) Registration errors with respect to the scaling factors using different template size.

### *2.2.4 Subpixel registration*

In many image registration applications, particularly, if measurements are done, the translation vector needs to be estimated at subpixel accuracy. In subpixel localization, the original continuous image function needs to be, effectively, reconstructed from the noisy samples, i.e., the image function is interpolated from the pixel values of a digital image. In some methods, the interpolation is done for the samples of correlation coefficients or in another feature space. A common approach is to use the Fourier space.

In an early review paper, Tian and Huhns (1986) describe four common subpixel registrations methods: correlation interpolation, intensity interpolation, differen-

tial method, and phase correlation. They also formulate a fast iterative algorithm for image registration using intensity interpolation and compare the methods as for accuracy, complexity, and applicability to implementation.

Correlation interpolation is a simple, non-iterative method for subpixel registration: Samples of correlation coefficients, or of another similarity measure, are computed at discrete translations, and the position of the maximum correlation is located. A parametric model is selected to describe the underlying continuous correlation function. The correlation peak and the samples around it are used to estimate the continuous correlation function. The coordinates of the peak of the function are solved analytically.

Obviously, the selection of the function is somewhat arbitrary. Hence, the interpolation of correlations is biased (Robinson & Milanfar 2004). However, if the sampling frequency is high enough, good subpixel estimates of translation are obtained with a rather small computational cost. Typical models for the correlation function are quadratic polynomials (Dvornychenko 1983; Gleason, Hunt & Jatko 1990) and Gaussian functions (Nobach & Honkanen 2005). Nobach and Honkanen (Ibid.) also formulate a method how to assign weights to the correlation samples, depending on their estimated reliability.

The following example demonstrates the method of correlation interpolation illustratively (see Figure 14). First correlation coefficients at discrete translations are computed at the selected 3 × 3 neighborhood around the peak correlation. Scaling factors were fixed to the rather optimal values of $c_x = 1.2$ and $c_y = 0.9$. Nine correlation samples are marked with circles at the top of vertical lines in Figure 14. A second order polynomial is used to model the continuous correlation surface:

**(20)** $$r(x, y) = (x^2 \quad y^2 \quad xy \quad x \quad y \quad 1)\mathbf{a}\,.$$

Parameter vector **a** is solved using the pseudo-inverse method:

**(21)** $$\mathbf{a} = (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{r}\,,$$

where

**(22)** $$\mathbf{X} = \begin{pmatrix} x_1^2 & y_1^2 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_N^2 & y_N^2 & x_N & y_N & 1 \end{pmatrix} \text{ and } \mathbf{r} = \begin{pmatrix} r_1 \\ \vdots \\ r_N \end{pmatrix},$$

where $N$ is the number of correlation samples.

The resulting paraboloid is plotted in Figure 14. The peak can be located analytically by solving the root of the derivative. The peak is thus located at (marked by a triangle in Figure 14):

$$\textbf{(23)} \qquad \begin{cases} x_{\text{peak}} = (2a_2a_4 - a_3a_5)/(a_3^2 - 4a_1a_2) \\ y_{\text{peak}} = (2a_1a_3 - a_3a_4)/(a_3^2 - 4a_1a_2). \end{cases}$$

In intensity interpolation, image intensities between the sampled pixels are estimated by means of interpolation. When correlations with non-discrete translations are estimated, either the reference or the target image is interpolated. Now the latter alternative is used. To obtain a continuous image function $g(x, y)$ the intensities of the discrete target image $f(x, y)$ are interpolated using some interpolation, e.g., a bi-cubic, kernel. If non-integer translations are used, $g$ is used instead of $f$ when computing correlations. In practice, samples of $g$ are computed only at positions that are needed in the registration.

Let us study the same subpixel correlations that are shown in Figure 14, now using intensity interpolation instead of correlation interpolation. The resolution of the translation is set to 0.2 pixels. The results are given in Figure 15. By upgrading the resolution to 0.02 pixels with an iterative step the subpixel translations of the peak correlation became: $x_{\text{peak}} = 0.22$, $y_{\text{peak}} = -0.20$, which deviate from the results of the correlation interpolation by about –0.01 and –0.05 pixels, respectively.

Let us compare the subpixel estimates of the correlation coefficients obtained by the two methods. Figure 16 shows the difference. The subpixel estimates of the correlation coefficients deviate from each other by less than 0.007 in this case. Nevertheless, the subpixel translation estimates deviate by 0.05 pixels. This can be regarded as an estimate of the subpixel localization error. Note that in both methods the model is parameterized. In correlation interpolation, the function model is arbitrarily selected, while in intensity interpolation, the interpolation kernel is selected.

The accuracies of different subpixel registration methods have been studied both theoretically and empirically. Schreier, Braasch, and Sutton (2000) remarked that interpolation of intensities alter the signal spectrum, both magnitude and phase. They performed numerical studies to estimate the amount of bias with respect to the subpixel shift. They generated artificial test images with prescribed translations and scaling factors from different speckle pattern. Translation was applied in the Fourier domain so that the phase and amplitude were not corrupted.

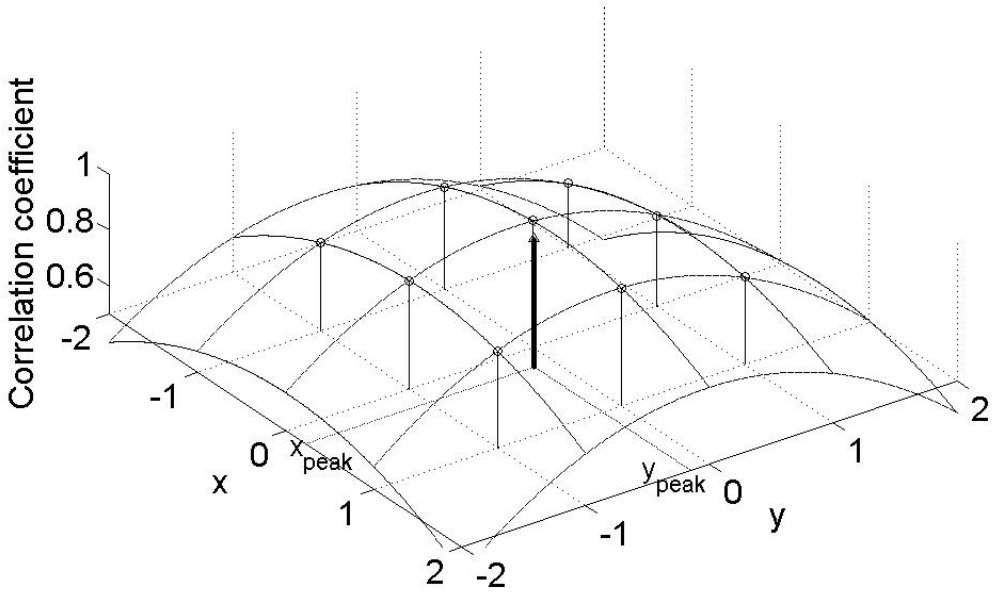



**Figure 14.** Subpixel registration by correlation sampling. Nine correlation coefficient samples at discrete intervals (o) are used to fit a paraboloid, whose peak (Δ) is located analytically. The subpixel registration results are: $x_{peak} = 0.23$, $y_{peak} = -0.15$.

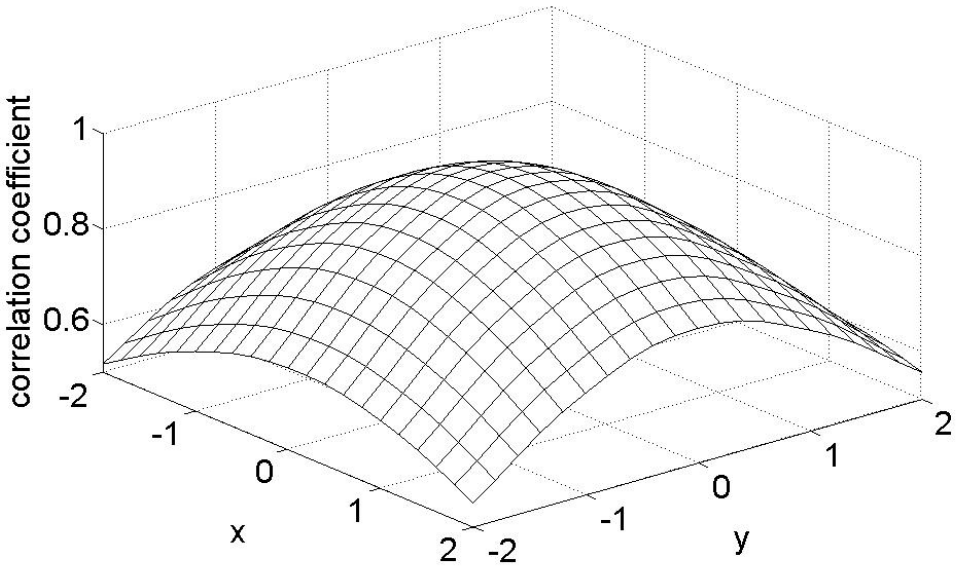



**Figure 15.** Subpixel correlation estimates obtained by interpolation intensities with a bi-cubic kernel. The subpixel registration results are: $x_{peak} = 0.22$, $y_{peak} = -0.20$.

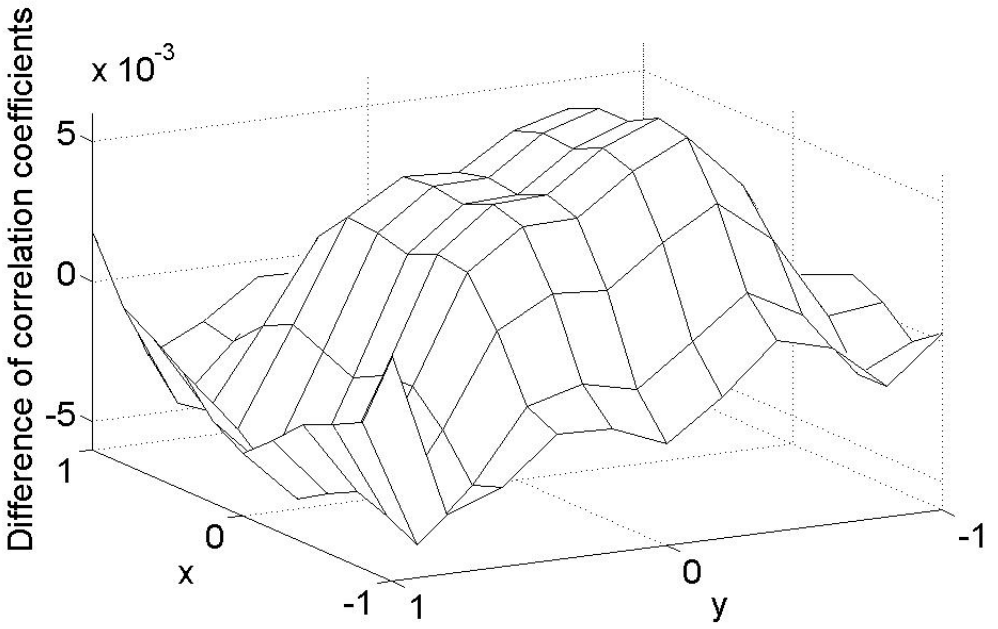



**Figure 16.** Difference between the subpixel correlation coefficients estimated by intensity interpolation and correlation interpolation.

The results showed, among others, that the typical bias was about 0.01 pixels when bi-cubic interpolation was used. The mutual superiority of the three interpolators tested, bi-cubic, cubic spline, and quintic spline, varied from test to test depending on the gray-value distribution. They showed empirical evidence that low-pass filtering reduces the bias caused by interpolation (Ibid.).

Forliti, Strykowski, and Debatin (2000) studied the bias and variance of various subpixel registration algorithms applied to digital particle image velocimetry (DPIV). A rotating mirror method was used to measure the registration error. Clear curves of bias errors, with respect to the subpixel component of the displacement, were detected for each of the methods tested. For parabolic fit, e.g., the bias varied from –0.17 to +0.17 pixels, the maximum bias being app. at subpixel displacements of 0.82 and 0.18, respectively. The other methods, Gaussian curve-fit, Centroid algorithm, and Whittaker's reconstruction, showed lower bias. For all these methods, the bias tended to shift the measurements towards integer pixels.

As for deviation, parabolic curve-fit was again the poorest of the tested methods when the subpixel displacement was close to 0.5. The standard deviation of the registration error was 0.1 pixels. On the other hand, the deviation decreased almost linearly to zero when approaching integer displacements. Gaussian curve-fit

was found to have the lowest bias and deviation. Moreover, the two components of error were found to be interdependent (Ibid.).

A calibration procedure to compensate for the bias by modeling it with, say, polynomial curves was suggested. The results showed that bias could be reduced, but overcompensation occurred in some cases. For the Whittaker's method, e.g., the bias could be reduced from about 0.1 pixels to as low as 0.03 pixels, as to the calibration set. The authors remarked, however, that the calibration model is valid only for a specific experimental setup (Ibid.).

Robinson and Milanfar (2004) studied the accuracy and bias of seven common subpixel registration algorithms: mean square distance with quadratic fit, correlation with quadratic fit, linear gradient-based method, multiscale gradient-based method, projection gradient-based method, projection multiscale gradient-based method, and phase of the Fourier transform. They also derived bounds to the achievable accuracy using the Cramer-Rao inequality. Kakarala and Hero (1992) have in turn studied the accuracy limit of localizing ideal step edges.

Robinson and Milanfar (2004) deduced that the variance would be infinite for any unbiased translation estimator. It was also inferred that the common correlation function would provide the desirable maximum likelihood (ML) estimator if only the continuous image function was known. Some image interpolation methods result in an approximate ML solution, particularly, if the signal to noise ratio (SNR) is high. Moreover, it was stated that images with a high contrast texture are easier to register.

The performance of the seven registration methods were evaluated with respect to the SNR. Test images were generated using discrete Fourier transform. The SNR was varied by adding white Gaussian noise. The measured root mean square errors (RMSE) were compared with each other and with the theoretical lower bound. In low-noise conditions, the phase method was superior with RMSE being about 0.002 pixels, but, conversely, it was one of the worst ones in high noise conditions, whence RMSE was about 0.8 pixels. The multiscale gradient-based method was found to be the best one in low and medium noise conditions. The mean square distance and correlation were almost insensitive to the SNR level. Their RMSE were about 0.05 and 0.11 pixels, respectively (Ibid.).

It is now evident that the mutual superiority of the different subpixel registration methods varies depending on, among others, the imaging setup and the image content. Moreover, there are many selection criteria that can be applied. These include bias and variance, tolerance to noise and other variations, as well as computational efficiency. Hence, the final selection of the method is up to several

practical issues, which are perhaps the most comfortably evaluated experimentally or on the basis of experience. In a typical experimental test, prescribed translations are produced using accurate mechanical stages (Chu et al. 1985; Hung & Voloshin 2003). There are also methods to predict the accuracy beforehand: Cox, Kruskal, and Wallach (1990) give some formulas to predict the accuracy as well as some rules of thumb to avoid situations of degraded accuracy. They used an edge detection method and a chamfer-matching like method as their test cases.

### *2.2.5 Search methods*

The previous subsections dealt with image transformations, similarity measurements, and subpixel translation estimation methods. Recall that the objective of image registration is to estimate the correct parameters of the image transformation or, at least, optimal parameters with respect to the selected similarity measure. Sometimes only translation, or another individual parameter, is of the primary interest, while others are adjusted merely to support the efficient estimation of the objective parameter.

Optimization using computational methods involves a few common concepts that are discussed next. Sometimes the result of optimization can be computed directly, but often iterative search methods are needed. The following basic steps are usually taken in iterative, numerical optimization:

1. Select a set of objective free parameters, and define their bounds and numerical encoding.
2. Define and implement an objective function.
3. Select an initial solution candidate or candidates.
4. Evaluate the candidate(s) by the objective function.
5. Update the candidate(s) by deterministic or stochastic rules.

Steps 3–5 are repeated until a stopping condition is met.

In general, image registration is a multidimensional and, usually, a multimodal optimization problem, in which an objective, or fitness, function is optimized. In addition to a similarity measure, the fitness function may have so-called regularization terms, too (Rueckert et al. 1999). The dimensionality of the objective function depends on the selected transformation model and its free parameters.

An objective function is a function that maps the values of the free parameters into a scalar value that describes the optimality of the given input values. In case of multi-objective optimization, several objective functions are used. The objec-

tive function can be a formula or, quite commonly, a computational algorithm, which may be very complex in some cases.

The output is not necessarily an absolute figure of optimization, because the range of possible values and, particularly, the value of the extreme optimum, is not necessarily known *a priori*. Nevertheless, a proper objective function $f$ should obey the following inequalities: Let vector $\mathbf{x}_0$ denote the optimal set of free parameters. Then

**(24)** $$f(\mathbf{x}_0) \leq f(\mathbf{x}) \text{ for all } \mathbf{x} \neq \mathbf{x}_0$$

should hold true for a minimization problem.

The equality is allowed if multiple minima with equal fitness exist. Moreover, in a minimization problem, for any solution candidate $\mathbf{x}_1$, that can be argued in some sense to be superior to $\mathbf{x}_2$, it should hold that $f(\mathbf{x}_1) < f(\mathbf{x}_2)$. Now equality is not allowed, because then there would be no guidance to update the solution candidates. In practical optimization problems, however, it is often difficult to evaluate whether these conditions hold true.

The shape of the fitness function, known as the fitness landscape, depends, at least, on the selection of the similarity measure, regularization term, image and noise content, transformation model, and interpolation methods. Moreover, the free parameters should have some lower and upper bounds as well as a resolution defined. The curse of multimodality can obviously be avoided as long as tight enough bounds can be set, which requires good prediction of the optimal values. As for computation, a selection between fixed-point and floating-point representation of the parameters has to be done. A great emphasis has to be laid, in particular, on the computational efficiency of the implementation of the objective function, because it is evaluated repeatedly (Frischholz & Spinnler 1993; Jian, Luk & Rueckert 2007; Burger & Burge 2008: 435). All these aspects, overall, have effects on the degree of difficulty of the optimization problem, as for the objective function.

The selection of an efficient method to update the solution candidates is another important issue. The optimal parameters are searched for by a suitable method that should usually be, at least, fast, accurate, and robust. Figure 9 showed that there are several local correlation maxima as for the plane spanned by translations. Thus iterative search methods may not find the global optimum from every starting point.

In general, exhaustive search is the only method to find a global optimum reliably. When the only transformation in use is translation with integer values, exhaustive search is still computationally feasible and quite commonly used. A direct subpixel method, say, parabolic fitting, can also be used without significantly adding the computational cost (Zitova & Flusser 2003).

Hierarchical coarse-to-fine approaches are used both to speed up the search and to avoid the convergence to local optima. An image pyramid of different resolutions is obtained by downsampling images using, e.g., antialiazing filters (Thévenaz, Ruttimann & Unser 1998) or wavelet transforms (You & Bhattacharya 2000). Sole low-pass filtering of the images can also enhance the robustness of registration under distortions sufficiently (Berg & Malik 2001). Hierarchical procedures can also be applied to tackle the complexity of the model (Lester & Arridge 1999).

Local optimization methods that are based on the partial derivatives of the fitness function and used in image registration are, according to the survey article of Zitova and Flusser (2003), Gauss-Newton, Levenberg-Marquardt, and gradient descent. The Powell's multidimensional direction set method (Press et al. 1992: 412–414), that does not require the computation of the gradient, has also been used (Maes et al. 1997; Jenkinson & Smith 2001; Fan & Han-ling 2007). These methods can be fast, but, as local methods, they are prone to converge to a local optimum close to the starting point. Multiple starting points can be used to find the global optimum more reliably, but then a trade-off between complexity and reliability has to be done.

According to Jenkinson and Smith (2001), local optimization methods in conjunction with multi-resolution approaches are not sufficient enough to find the global optimum positively in image registration problems. When local methods are not preferred and exhaustive search is too slow, global optimization methods can be used.

One popular method is to use evolutionary algorithms (EAs) and, particularly, genetic algorithms (GAs) to global optimization (Forrest 1993; Whitley 2001). In short, EAs apply the principles of evolution to computational optimization. An introduction to evolutionary algorithms is given later in Chapter 4.

Genetic algorithms have been applied to image registration in numerous publications. These can be retrieved from an indexed GA bibliography (Alander 1994). One of the earliest papers, in which the image registration problem is addressed by genetic algorithms, was published in 1984 (Fitzpatrick, Grefenstette & Gucht 1984). Some recent journal articles that report the application of GAs to image

registration are, among others, (Shaopeng & Guanchang 2003; Pilch, Mahajan & Chu 2004; Chow, Tsui & Lee 2004; Jin & Bruck 2005; He et al. 2006).

Hybrid algorithms that utilize several search strategies have also been tested. In one of such studies, Jenkinson and Smith (2001) used a combination of global and local search. The initial global search to find preliminary solutions was carried out by splitting the search space of rotation into a coarse grid and evaluating preliminary translation and scaling coefficients at each grid point. Subsequently, a finer grid of rotation angles was applied and the optimal translations at those rotations were predicted by interpolation. The costs of the intermediate grid points of rotation were then evaluated using the true rotations and the predicted translations. Finally, local minima were localized by optimization. Moreover, a multiresolution approach was adopted. Hybrids have also been used, among others, in the following publications (Luck, Little & Hoff 2000; He et al. 2006; Fan & Hanling 2007).

In addition to genetic algorithms, other heuristic algorithms inspired by nature have been applied to image registration, too. These include simulated annealing (Starink & Backer 1995; Luck, Little & Hoff 2000) and ant colony optimization (Fan & Hang-ling 2007).

## 2.3 Single camera calibration

Calibration is essential when accurate measurements based on image are done. Camera calibration is a process where the values of an appropriate imaging model, a camera model, are acquired. In a frequently cited paper, Roger Y. Tsai (1987) gives a comprehensive tutorial to different camera calibration techniques and describes practical algorithms in detail. Although the paper was published as early as in the late 80's, it can be recommended for a tutorial paper even today. Clarke and Fryer (1998) in turn give a good review of the key developments in the long history of camera calibration methods and models.

For a single camera, camera calibration can be used to acquire the mapping between the world and the image coordinate systems. Such a projective model describes the orientation of the camera, i.e., the extrinsic camera parameters, and the intrinsic parameters of the camera. The intrinsic parameters may include focal length, dimensions and orientation of the imaging cell and pixels, and geometrical distortions caused by the optics. An inverse mapping, from image coordinates to world coordinates, is called back-projection. For a single camera, back-projection defines a line in the 3D world coordinate system. If a stereo vision system, with

two or more cameras, is used, back-projection can be used to measure 3D positions from pairs of corresponding feature points in two images and, furthermore, to reconstruct 3D objects or motion (Sonka 2008: 561–569).

It is not necessary to solve the extrinsic and intrinsic parameters explicitly. In implicit calibration, the objective is to model the overall projection or back-projection without having parameters of physical meaning related to the camera and the imaging setup (Wei & Ma 1994).

Usually calibration involves an experimental part, where a scene with some prescribed calibration landmarks with known positions is imaged (Sonka 2008: 565–566). The landmarks are commonly defined as points, but an approach that uses edges has been reported, too (Robert 1994). In self-calibration, on the contrary, a sequence of images is taken from an unknown scene, usually from different perspectives. The camera model is then obtained by finding common landmarks and maximizing the consistency of their projections on different images. Yet, usually some constraints on the camera parameters, scene, and camera motion are required to set in self-calibration (Maybank & Faugeras 1992; Hemayed 2003).

A typical procedure for a single-camera calibration obeys the following workflow:

1. Image a scene with known landmarks.
2. Localize the landmarks, preferably in subpixel accuracy.
3. Select a parameterized camera model.
4. Select an objective function to evaluate the accuracy of the camera model.
5. Optimize the camera model with respect to the objective function.

Next these steps of camera calibration are dealt in more detail. Moreover, the calibration procedure used in our strain analyzer, called DeforMEERI, is introduced. In addition to more accurate strain measurements, calibration enables to image the test object from other than perpendicular angles. Moreover, a metric calibration enables to select the gauge length afterwards.

The next subsections are organized as follows. Section 2.3.1 introduces the camera model that uses the common pinhole model and compensation for radial and tangential distortions. Section 2.3.2 deals with different calibration patterns and objects. Now calibration was done using monoview coplanar landmarks that implemented by printing chess-board-like pattern with a laser printer. Section 2.3.3, introduces an algorithm to localize the landmarks as automatic as possible. Section 2.3.4 discusses the optimization of the free camera model parameters. Finally, formulas to the back-projection when using the single-camera model are given

in Section 2.3.5. Experiments with the proposed calibration method as well as results are handled in parallel with the discussion.

### *2.3.1 Camera model*

Perspective projection is usually used when the imaging geometry, i.e., the paths of the light rays, of typical cameras is modeled. Projective geometry corresponds to the ideal pinhole camera, or *camera obscura*, model. According to Lindberg (1968), the analysis of pinhole cameras originates from the pioneering scientific work of Ibn Al-Haytham (965–1040 CE), who is sometimes regarded as the "father of modern optics" (Tbakhi & Amr 2007).

Orthographic projection, which is obtained from perspective projection by increasing the focal and the object distance to infinity, can also be used in some cases. The advantage of an orthographic model is its linearity, which greatly simplifies the computational issues (Sullivan et al. 1997). On the other hand, explicit camera models are not needed if, e.g., artificial neural networks (ANN) are used to learn the mapping between the measured image coordinates, subject to distortions, and the world coordinates (Do 1999; Memon & Khan 2001). ANNs have also been applied to model just the image distortion (Choi et al. 1994).

An imaging method antithetic to the pinhole camera has also been suggested. In 'pinspeck', or anti-pinhole, imaging, the roles of the pinhole and the aperture are interchanged. During a solar eclipse such a negative geometrical image of the sun is visible (Cohen 1982).

Next the perspective projection model and the aberration models used in DeforMEERI are introduced. Parameterized formulas to map points in the world coordinate frame into the pixel coordinates of the image, i.e., a forward projection model, are dealt using, mostly, the homogeneous coordinates presented in Section 2.2.2.

The aberration-free pinhole camera model is illustrated in Figure 17. Let $W$ be the world coordinate frame and $\mathbf{O}_w$ its origin. $C$ is the camera coordinate frame, whose origin $\mathbf{O}_c$ is fixed to the center of projection, or focal point, i.e., the pinhole, through which all light rays are assumed to pass. The image frame is parallel to the $xy$-plane of $C$. The $z$-axis of $C$ is called the optical axis. Point $(0, 0, f)^{\mathrm{T}}$ in $C$, i.e., the intersection of the optical axis and the image frame, is called the principle point. Distance $f$ is the focal length of the camera (Tsai 1987; Heikkilä 2000; Sonka 2008: 561–562).

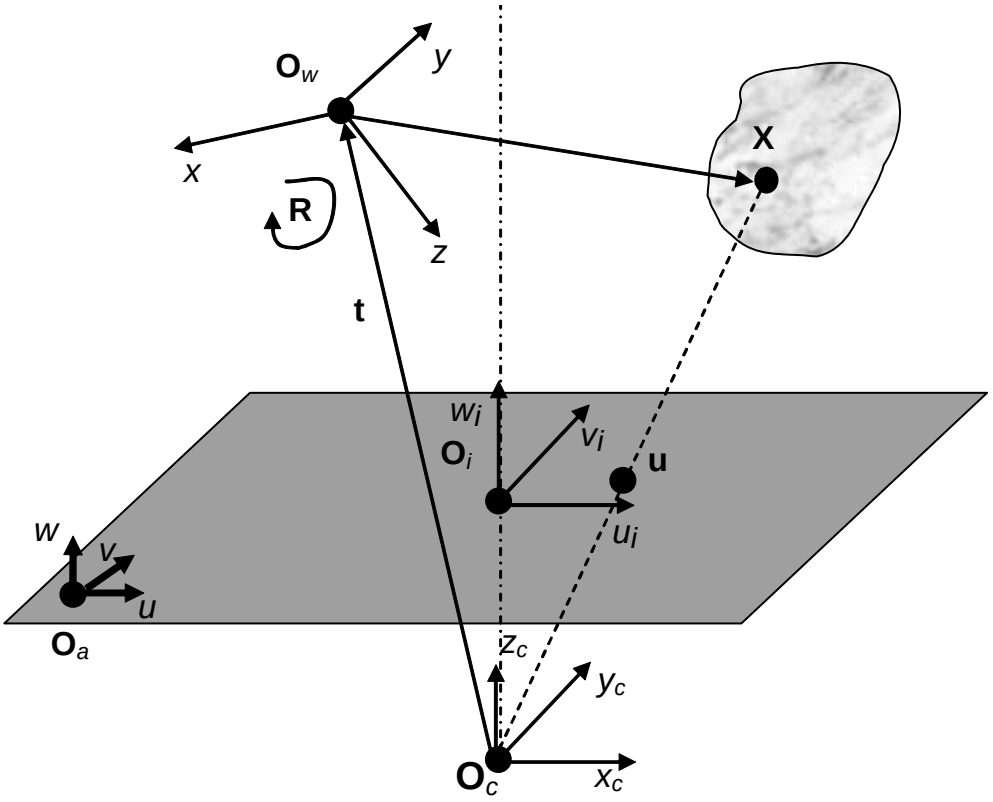



**Figure 17.** Pinhole camera model and different coordinate systems. The subscripts are as follows: $w$ = world coordinates, $c$ = camera coordinates, $i$ = image Euclidian coordinates, and $a$ = image affine coordinates. $\mathbf{O}_c$ is the focal point and $\mathbf{O}_i$ is the principal point. **u** is the image of **X**.

The image Euclidean coordinate system $I$ has its origin $\mathbf{O}_i$ at the principal point. Its coordinate axes $u_i$, $v_i$, and $w_i$ are parallel to $x_c$, $y_c$, and $z_c$, respectively. The image affine coordinate system $A$ is used when the, possibly real-valued, pixel coordinates are referred. Its origin $\mathbf{O}_a$ is fixed to the top-left pixel of the image sensor. The $u_a$ and $v_a$ axes are parallel to the rows and columns of the pixels, respectively. While $w_a$ is parallel to $w_i$, either $u_a$ or $v_a$ may not be parallel to $u_i$ or $v_i$, respectively, if the image cell has some shear (Tsai 1987; Heikkilä 2000; Sonka 2008: 563).

When aberrations are modeled, another coordinate system $D$ is used. Its origin $\mathbf{O}_d$ coincides with $\mathbf{O}_i$ and the coordinate axes are collinear with $u_i$, $v_i$, and $w_i$. However, there is a nonlinear transformation from $I$ to $D$, which implies that $D$ is not a Euclidean space.

**X** denotes an arbitrary point $(x, y, z)$ in $W$ such that $z_c > f$. Its projection, i.e., its image, which is obtained using the aberration-free pinhole model is marked by **u**. The projection model is found by performing a series of coordinate transforma-

tions, namely, $W \rightarrow C \rightarrow I \rightarrow D \rightarrow A$. Apart from the third mapping, all of these transformations can be written in linear terms in homogeneous coordinates.

The aberrations are often modeled in $A$ due to computational issues, particularly, when a linear calibration method is used to initialize a nonlinear method. Yet the mapping order used here is more natural, because the aberrations occur in the optics of the camera, not in the image cell. This mapping order has been used, e.g., in (Bouchouicha, Khelifa & Pusch 2003).

The first transformation ($W \rightarrow C$) describes to position and the orientation of the world coordinate frame relative to the camera coordinate frame.

$$\mathbf{X}_c = \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix} = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{pmatrix} \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix}. \tag{25}$$

where $\mathbf{t} = (x_0, y_0, z_0)^{\mathrm{T}}$ is the translation vector and $\mathbf{R}$ is a $3 \times 3$ orthonormal rotation matrix. The rotation matrix is formed using three *Euler angles* $\alpha$, $\phi$, and $\kappa$, i.e., rotation about the $x$-, $y$-, and $z$-axes of $W$, respectively:

$$\mathbf{R} = \mathbf{R}_\kappa \mathbf{R}_\phi \mathbf{R}_\omega = \begin{pmatrix} \cos\phi\cos\kappa & \cos\omega\sin\kappa - \sin\omega\sin\phi\cos\kappa & \sin\omega\sin\kappa - \cos\omega\sin\phi\cos\kappa \\ -\cos\phi\sin\kappa & \cos\omega\cos\kappa - \sin\omega\sin\phi\sin\kappa & \sin\omega\cos\kappa - \cos\omega\sin\phi\sin\kappa \\ \sin\phi & -\sin\omega\cos\phi & \cos\omega\cos\phi \end{pmatrix}, \tag{26}$$

where

$$\mathbf{R}_\omega = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\omega & \sin\omega \\ 0 & -\sin\omega & \cos\omega \end{pmatrix}, \tag{27}$$

$$\mathbf{R}_\phi = \begin{pmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{pmatrix}, \text{ and} \tag{28}$$

$$\mathbf{R}_\kappa = \begin{pmatrix} \cos\kappa & \sin\kappa & 0 \\ -\sin\kappa & \cos\kappa & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{29}$$

The three translation components and the three Euler angles constitute the extrinsic parameters.

The transformation from the camera coordinates to the image Euclidean coordinates ($C \rightarrow I$) is a nonlinear projection, which, however, can be written in linear terms in homogeneous coordinates:

$$\mathbf{u}_i = \begin{pmatrix} u_i \\ v_i \\ z_c \end{pmatrix} \cong \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix}. \tag{30}$$

According to eq. (30), the image coordinates of an ideal perspective projection are thus $u_i = \frac{x_c f}{z_c}$ and $v_i = \frac{y_c f}{z_c}$. In reality, the light rays do not intersect the image frame at exactly these positions due to optical distortions. Next these geometrical distortions caused by the optical system are discussed, modeled, and corrected in order to obtain the true intersection coordinates.

The pinhole camera model assumes that an ideal lens with zero thickness is used. In an ideal pinhole camera, with this paraxial approximation, rays pass through the focal center of the lens and continue as straight lines. In practice, the lens causes some distortions, i.e., aberrations, to this simple geometry. Aberrations are mainly due to the nonzero thickness of the lens and the diffraction occurring at the edges of the aperture. In addition to the lens, the shape of the image frame may deviate from flat and cause distortions to the image.

*Coddington equations*, invented by Henry Coddington as early as 1829 (as cited in Campbell 2006), and *ray tracing* (Chang & Stavroudis 1980; DeJager & Noethen 1992) provide a general tool to compute the image of an incoming bundle of rays that pass through a refracting optical system. However, ray tracing may be too complex a method for practical measurements unless extreme accuracy is needed.

In the past, different optical instruments were used to compensate for the aberrations when traditional photographing films were studied in photogrammetric applications. Back then, it was possible to correct only aberrations that were symmetrical with respect to the optical axis (Hothmer 1958).

Nowadays aberration corrections are usually done computationally, and unsymmetrical distortions can also be handled. Typical distortion models manage radial, decentering, also known as tangential, and linear distortions with polynomials.

The widely adopted formulae for the radial and tangential distortions were suggested by Brown as early as in the 60's (Brown 1971). The calibration methods of Brown were originally developed for military applications. It has been argued that polynomials are not a desirable modeling method, because the different terms have a high intercorrelation (Ziemann 1986 as cited in Clarke & Fryer 1998).

Radial aberration is modeled with respect to the radial center using even-order polynomials:

$$\begin{cases} \delta u^{(r)} = u_i \left( \kappa_1 r^2 + \kappa_2 r^4 + \kappa_3 r^6 + \ldots \right) \\ \delta v^{(r)} = v_i \left( \kappa_1 r^2 + \kappa_2 r^4 + \kappa_3 r^6 + \ldots \right) \end{cases}, \tag{31}$$

where

$$r^2 = u_i{}^2 + v_i{}^2, \tag{32}$$

because the origin of $I$ was fixed to the optical axis, which in turn should coincide with the radial center.

Tangential distortion is caused, e.g., by the misalignment of the lens. The polynomials that approximate the tangential distortions are:

$$\begin{cases} \delta u^{(t)} = 2 p_1 u_i v_i + p_2 \left( r^2 + 2 u_i{}^2 \right) \\ \delta v^{(t)} = p_1 \left( r^2 + 2 v_i{}^2 \right) + 2 p_2 u_i v_i \end{cases}. \tag{33}$$

Now the distorted image coordinates are obtained by:

$$\begin{cases} u_d = u_i + \delta u^{(r)} + \delta u^{(t)} \\ v_d = v_i + \delta v^{(r)} + \delta v^{(t)} \\ w_d = 1 \end{cases}. \tag{34}$$

Note that the inverse transformation $D \rightarrow C$ is not straightforward, as discussed in (Heikkilä 2000). This issue will be revisited when back-projection is discussed in Section 2.3.5.

The last coordinate transformation, $D \rightarrow A$, is performed using the affine transformation introduced in eq. (7):

$$\mathbf{u}_a = \begin{pmatrix} u_a \\ v_a \\ 1 \end{pmatrix} \cong \begin{pmatrix} D_u & s & u_0 \\ 0 & D_v & v_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} u_d \\ v_d \\ 1 \end{pmatrix}, \tag{35}$$

where the translation vector $(u_0, v_0)^T$ determines the position of the optical axis in terms of pixel coordinates. Scaling factors $D_u$ and $D_v$ calibrate the conversion from the units of $D$ to pixels independently along the $u$- and $v$-axes, respectively. Because $f$, $z_0$, and the scaling factors depend strongly on each other, a so called normalized image plane with $f = 1$ is sometimes used to simplify equations and calibration (Sonka 2008: 564). If $f$ and $z_0$ were calibrated in meters, the inverse of $D_u$ and $D_v$ would give the dimensions of the pixels. However, normalized image plane is used in this study.

The shear parameter $s$ is used to compensate for the linear distortion. Linear distortion occurs if the image cell is misaligned so that its normal is not parallel to the optical axis or if the image axes are not orthogonal to one another (Ibid.: 564).

### *2.3.2 Calibration objects and landmark patterns*

Accurate camera calibration needs an adequate number of reliably detected landmarks whose world coordinates are known *a priori*, except in self-calibration (see the beginning of Section 2.3), and whose image coordinates are measured in subpixel accuracy. Usually artificial landmarks that form a regular pattern are used.

Because the selection of the origin $\mathbf{O}_w$ and the orientation of $W$ are arbitrary, and provided that the dimensions of the landmark pattern are known, the world coordinates of the landmarks can be solved analytically. The regularity of the pattern definitely helps the computation of the coordinates, but, still, there is a correspondence problem between the known world coordinates and the measured image coordinates. Particularly, image noise may lead to miss landmarks or to have false detections.

Robustness may be enhanced by manual control, because calibration is done only seldom in some typical applications. Nevertheless, automation and speed of calibration is usually preferred at least in robot guidance (Heikkilä 2000) and in industrial applications.

Different calibration patterns have been reported in the literature. Villa-Uriol et al. (2004) discussed the suitability of different calibration tools, including calibration objects, to different applications. Calibration objects with artificial landmarks can be either two- or three-dimensional. The latter option is usually used in stereo vision calibration. 3D calibration objects are cubical or spherical with either a circular, square, or line pattern (Ibid.; Orteu, Garric & Devy 1997). It has been claimed that it is difficult to produce and maintain 3D calibration objects (Orteu, Garric & Devy 1997).

Planar objects are in turn used in single camera calibration, but also in stereo calibration, whence multiple viewing angles are used (Zhang 2000). Typical patterns used in 2D calibration objects are line grids (Orteu, Garric & Devy 1997), squares (Zhang 2000), chessboards (checkerboards) (Arca, Casiraghi & Lombardi 2005), diamonds (Weng, Cohen & Herniou 1992), and circles (Heikkilä 2000). Usually the patterns are in black and white, but sometimes bright colors are also used (Villa-Uriol et al. 2004).

While the manufacture of accurate 3D calibration objects may be laborious, a rather accurate planar object can be produced without any special equipment. Nowadays the off-the-shelf laser printer technology enables printing the pattern on a paper with, say, 600 dpi resolution, i.e., with an accuracy of app. 0.04 mm. Subsequently, the paper should be attached diligently to a planar surface (Shang 2000). Weng, Cohen, and Herniou (1992) have in turn used a high-precision photographic process and glass plates for high-accuracy patterns with an accuracy < 0.05 mm. Furthermore, they mounted the calibration object on a precision stand with a positioning accuracy being also 0.05 mm.

It is often beneficial to use, say, the centers of a square as the landmark position, because the center can be localized very accurately using averaging of several measurements. However, Heikkilä (2000) has remarked that aberrations cause a bias to the image coordinates when landmarks with a finite geometry are used. For instance, the projection of a circular landmark appears, approximately, as an ellipse in the image, but the center of the ellipse in the projection is not necessarily the projection of the center of the circle. Heikkilä suggest an algorithm to remove the bias.

Conversely, if 0-dimensional objects, such as object corners or intersection of lines, are used, the bias problem should be avoided. Therefore, the corners of a chessboard were selected as the landmarks.

The aberration and camera models are only practical simplifications of the reality. In fact, each object point has an aberrated three-dimensional image called a *caustic surface* (Chang & Stavroudis 1979).

### *2.3.3 Landmark localization*

The calibration object may be imaged, e.g., in different distance, orientations, and lighting conditions. Moreover, the calibration image may be partly occupied by a background of various textures, and the position as well as the scale of the cali-

bration pattern may be unknown. Hence, automatic and robust localization of the landmarks is generally a demanding task.

Arca, Casiraghi & Lombardi (2005) have tackled this automation task in three steps. First, a subimage containing only the chessboard, which was their pattern, is extracted. Second, the dimensions of the basic elements, i.e., the squares, are estimated. Finally, the corners are localized in subpixel accuracy. In steps 1 and 2, they used four oriented first order derivative of Gaussian filters as edge filters and the Radon transformation to detect lines. The final corner localization is done using a binarized image and evaluating some rule-based characteristic functions for each pixel. The final subpixel position is obtained by a morphological closing and by circle centroid localization.

Next a novel approach to localize the corners of a chessboard calibration pattern automatically in a general case is introduced. The position and orientation of the calibration object can be somewhat arbitrary, even though extensive testing has not been done so far. The corner detection method is a modification of the Harris detector (Harris & Stephens 1988; Sonka 2008: 158). In addition to the corner localization method, an algorithm to remove false corner detection and to assign correct world coordinates to the localized landmarks is given. Moreover, some tests and results are shown.

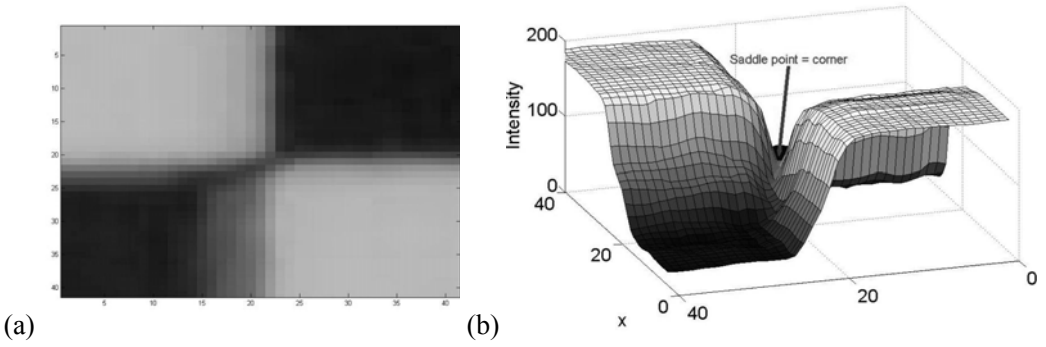



**Figure 18.** (a): A close-up of a chessboard corner. (b) The corner in (a) presented as an *x*-y-intensity plot.

Figure 18 (a) depicts a close-up of one corner of the chessboard. In Figure 18 (b), the intensities of the same corner are plotted against the image coordinate axes, a manner of representation that illustrates that the corner forms a saddle point. Saddle points can be detected with the aid of the Hessian matrix **H**:

**(36)** $$\mathbf{H}(x,y)=\begin{pmatrix}\dfrac{\partial^2 f(x,y)}{\partial x^2} & \dfrac{\partial^2 f(x,y)}{\partial x\partial y}\\ \dfrac{\partial^2 f(x,y)}{\partial x\partial y} & \dfrac{\partial^2 f(x,y)}{\partial y^2}\end{pmatrix}.$$

For a saddle point, the eigenvalues of the Hessian have opposite signs, which is equal to have a negative determinant for a 2 × 2 Hessian. As for a digital image, the Hessian can be estimated numerically using derivative filters. The modified Harris corner detection algorithm is then as follows:

1. Filter the image with a Gaussian or another isotropic low-pass filter.
2. Estimate the partial derivatives required in the Hessian (eq. 36) using, say, the following simple convolution masks for horizontal and vertical derivatives, respectively:

**(37)** $$\mathbf{D}_x=\begin{pmatrix}-1 & 0 & 1\\ -1 & 0 & 1\\ -1 & 0 & 1\end{pmatrix} \text{ and } \mathbf{D}_y=\begin{pmatrix}-1 & -1 & -1\\ 0 & 0 & 0\\ 1 & 1 & 1\end{pmatrix}.$$

   The second order derivatives are obtained by two consecutive convolutions, e.g.:

**(38)** $$\frac{\partial^2 f}{\partial x\partial y}=\mathbf{D}_x * \mathbf{D}_y * \mathbf{F}_G,$$

   where $\mathbf{F}_G$ is the Gaussian filtered image and $*$ refers to the convolution operator.
3. Construct matrix **A**, an estimate of the Hessian matrix, at each pixel ($x$, $y$).
4. For each pixel, compute the response function $R$ as follows:

**(39)** $$R(x,y)=-\det(\mathbf{A}(x,y))-\kappa\cdot\operatorname{trace}^2(\mathbf{A}),$$

   where a tunable parameter $\kappa \approx 0.04 \ldots 0.15$ and trace(**A**) is the sum of the diagonals of **A**.
5. Apply a suitable threshold to the response function to obtain corner candidates.

Figure 19 (a) shows an image of the chessboard pattern planar calibration object. The values of the response function $R$ are in turn shown in Figure 19 (b). The results show that the response is strong in the true corners. However, some weaker false responses also occur, particularly, at the borders of the calibration object where some half-cut squares exist.

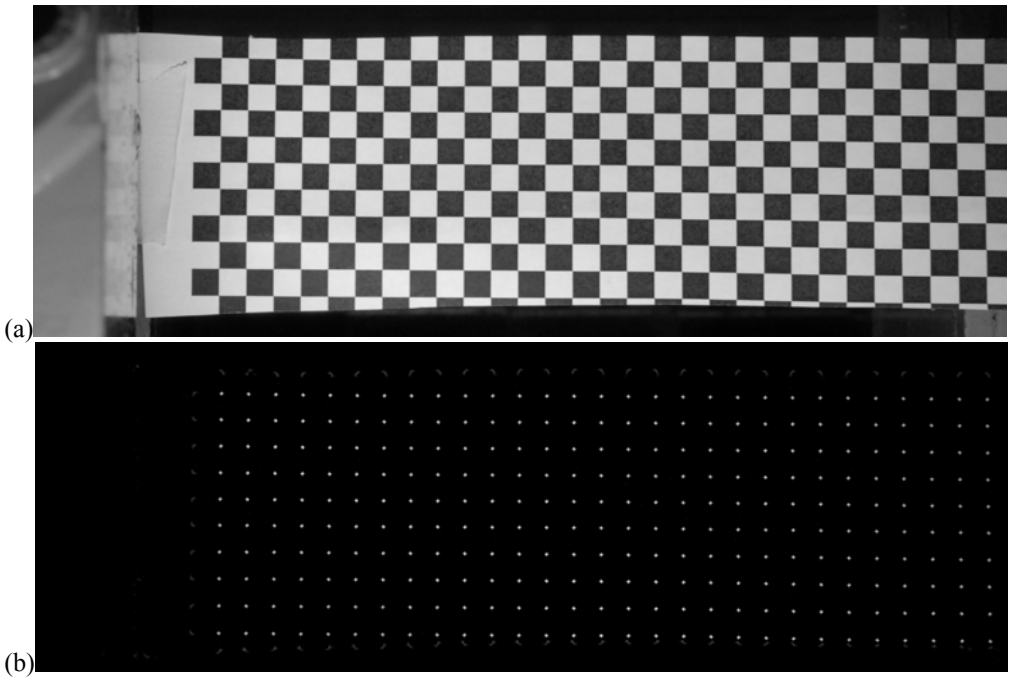



**Figure 19.** (a) Input image to single camera calibration. A chessboard pattern with known dimensions. (b) Response function *R* scaled to gray values {0, 1, …, 255}.

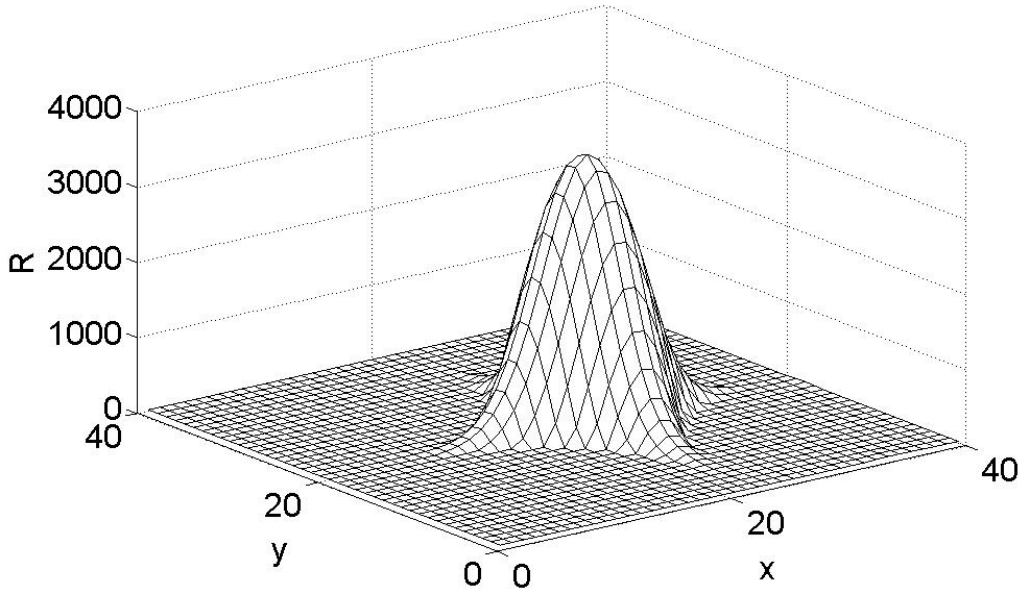



**Figure 20.** A peak of the response function around a corner location.

Figure 20 shows how the response function behaves around a corner; a smooth and continuous surface is detected. Thus a straightforward application of a threshold would give a set of pixels per corner, although the desired output is to have only one segmented pixel per corner. The desired result is achieved as follows:

Filter the response function with a morphological maximum filter so that the local maxima are spread enough. The result is denoted by $R_m$. Subsequently, apply the following logical filtering for each pixel $(x, y)$:

$$\begin{aligned} &\text{If } R_m(x,y) > R_{\text{thres}} \text{ and } R_m(x,y) = R(x,y), \text{ then } R_{\text{peak}}(x,y) \leftarrow 1 \\ &\text{otherwise } R_{\text{peak}}(x,y) \leftarrow 0 \end{aligned} \quad , \tag{40}$$

where $R_{\text{thres}}$ is a suitable threshold value. Now the coordinates of the peaks can be recorded.

The modified Harris corner detector, as described previously, gives only preliminary corner candidates. Subsequently, false corners have to be eliminated and the world coordinates of the true corners computed. The robust landmark search algorithm suggested here consists of the following steps:

1. Find corner candidates as described above.
2. Compute the median distance between the neighboring corners.
3. Remove false corners.
4. Fix the origin $\mathbf{O}_w$ to some peak $\mathbf{X}_0$.
5. Find the four nearest neighbors of $\mathbf{X}_0$. If four reliable peaks cannot be found, return to step 4 and select another peak.
6. Look for the other peaks and compute their world coordinates by traveling systematically from peak to peak, starting from the origin. Utilize the median distance between the neighboring corners for predicting the peak positions and for robust localization.
7. Obtain the subpixel coordinates of the peaks using response function interpolation, e.g., by a parabolic fit.

The elimination of false corners (step 3) is based on the median distance computed in step 2. True corners have at least two neighbors approximately at the median distance, a situation which is unlikely to occur for false corners.

Steps 4 and 5 create, iteratively if necessary, a reliable starting point for the final corner localization and fixing of world coordinates. First, find a peak $\mathbf{X}_1$, whose distance to $\mathbf{X}_0$ has the lowest deviation from the median distance computed in step 2. Next, reflect $\mathbf{X}_1$ through $\mathbf{X}_0$ and find the closest peak candidate. If the candidate is found within a prescribed tolerance distance of the predicted position, select it as $\mathbf{X}_2$.

Peaks $\mathbf{X}_1$ and $\mathbf{X}_2$ are then removed from the set of peak candidates, after which the same steps that were performed when looking for $\mathbf{X}_1$ and $\mathbf{X}_2$ are repeated to find $\mathbf{X}_3$ and $\mathbf{X}_4$. If any peak $\mathbf{X}_{1–4}$ is not found within the tolerance distance, the algorithm returns to step 4.

Step 6 is implemented in a radial column-wise manner. The search order is shown in Figure 21. Note that the algorithm can also handle missing peaks: If a peak is not found within the tolerance distance of its predicted position, it is ignored and the next peak is then looked for.

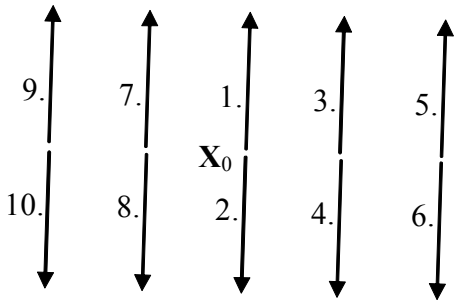



**Figure 21.** Search order of the peaks. $\mathbf{X}_0$ is the selected origin and the starting point.

### *2.3.4 Search of camera parameters*

Once an adequate number of corresponding image and world coordinate pairs have been obtained, the values of the camera parameters can be solved. However, some particular orientations of the calibration object may lead to singularities in computation (Sturm & Maybank 1999). Closed-form, linear solutions exist for distortion-free camera models. While they are fast to compute, they are also considered unreliable, because they do not produce minimum variance estimates (Heikkilä 2000). Nevertheless, the linear methods are widely used, at least, to give an initial estimate for nonlinear optimization methods (Heikkilä 2000; Zhang 2000).

In general, the search of the camera parameters, denoted by $\boldsymbol{\theta}$, requires optimization of an objective function. Typically, it is assumed that the measurement noise of the landmarks is independent and identically distributed, whence the maximum-likelihood estimate is obtained by minimizing the sum of squared distances between the measured image points $\mathbf{u}_i$ and the predicted points $\hat{\mathbf{u}}_i$ obtained by the camera model from the corresponding world coordinates $\mathbf{X}_i$:

$$\arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{N} \left\| \mathbf{u}_i - \hat{\mathbf{u}}(\boldsymbol{\theta}, \mathbf{X}_i) \right\|^2 , \tag{41}$$

where $N$ is the number of localized landmarks and the norm is the Euclidian distance (Zhang 2000).

Heikkilä (2000) noted that the measurement noise is not statistically independent and identically distributed if the landmarks appear, say, at different size. He took that into account by a covariance matrix of the measurement noise.

Minimizing the objective function in eq. (41) requires nonlinear optimization. A common iterative method is the Levenberg-Marquardt method that was used, e.g., in (Heikkilä 2000; Zhang 2000). Then a good initial guess is required so that the global optimum is reached. Global optimization methods, for instance, genetic algorithms, have also been applied (Ji & Zhang 2001; Bouchouicha, Khelifa & Puech 2003). Sometimes only the pose is estimated by GA (Hati & Sengupta 2001) or GA is used to determine the scale factors in self-calibration (Lei et al. 2002). On one hand, local search methods are fast. On the other hand, global methods do not require so good initial estimates of the solution.

In DeforMEERI, the search of the camera parameters is done in three steps: initialization and two iterative optimization steps using the objective function in eq. (41). In initialization, the values of some camera parameters are computed directly.

First, it is supposed (temporarily) that there is no distortion, i.e., $\kappa_1 = 0$, $\kappa_2 = 0$, $\kappa_3 = 0$, $p_1 = 0$, $p_1 = 0$, and $s = 0$. Second, a good guess for the principal point is at the center of the image, i.e.:

**(42)** $$u_0 = M / 2 \text{ and } v_0 = N / 2 \ ,$$

where $M$ and $N$ are the width and height of the image, respectively, given in pixels.

Third, the size of one chessboard square is known both in meters and in pixels, as the median size $\Delta u_{\text{med}}$ was measured in the corner localization algorithm. However, the deviation of the square size in pixels may be great if the calibration board stands very oblique to the optical axis. Due to the intercorrelations between $f$, $z_0$, $D_x$, and $D_y$, parameters $f$ and $z_0$ can be fixed to arbitrary values, unless their absolute values are of interest. Let $\Delta X$ be the length of a chessboard square in meters. The width of a horizontally oriented square in the affine pixel coordinates, distortions omitted, is $\Delta u = D_x \, \Delta X \, f / z_0$.

Assuming that the aspect ratio of the pixels is 1 and using the median square size, the initial guesses for the scaling factors are obtained by arranging the terms:

**(43)** $$D_x = D_y = \frac{z_0}{f} \frac{\Delta u_{\text{med}}}{\Delta X} .$$

The values of $x_0$ and $y_0$, related to the origin $\mathbf{O}_w$, are obtained similarly. $\mathbf{O}_w$ was fixed to landmark $\mathbf{X}_0 = (X_0, Y_0)$, which must not be confused with the principal point, even though they may locate close to one another. Point $x_0$ projects to pixel $X_0 = D_x\, x_0\, f / z_0 + u_0$. Substituting eq. (43) to $D_x$ gives: $X_0 = x_0\, \Delta u_{\text{med}} / \Delta X + u_0$. Finally, rearrangement of terms gives:

$$\textbf{(44)} \qquad \begin{cases} x_0 = (X_0 - u_0)\dfrac{\Delta X}{\Delta u_{\text{med}}} \\ y_0 = (Y_0 - v_0)\dfrac{\Delta X}{\Delta u_{\text{med}}} \end{cases}.$$

The rotation $\kappa$ about the $z$-axis can be estimated using the five starting points $\mathbf{X}_{0\text{–}4}$. Select, e.g., vectors $\mathbf{X}_1$–$\mathbf{X}_0$ and $\mathbf{X}_4$–$\mathbf{X}_0$ as the base of the world coordinate frame. Then the rotation between the world and the image coordinate systems can be calculated. Originally, zeros are assigned to the other Euler angles.

The initial estimates for every camera model parameter used in DeforMEERI have now been obtained, and they can be used as a seed for the iterative search steps. The two iterative search steps were performed using the unconstrained `fminunc` function, and its medium-scale version, of Matlab. It uses a quasi-Newton line method with a mixed quadratic and cubic line search. The approximation of the Hessian matrix is updated using the Broyden–Fletcher–Goldfarb–Shanno (BFGS) method (Broyden 1970; Fletcher 1970; Goldfarb 1970; Shanno 1970; The MathWorks 2006).

First, nonlinear optimization is performed without tangential distortion, i.e., $p_1 = 0$ and $p_2 = 0$. When all the parameters were optimized simultaneously, the search was found to be sensitive to noise and to the selection of the seed values. Moreover, the search tended to converge to different local optima. Although $f$ and $z_0$ were fixed in the initialization step, they are now subject to optimization, too. Finally, all camera parameters are subject to optimization in the second iterative step.

Some experiments are now done to evaluate the accuracy and robustness of the proposed calibration method. The objective is not to make any extensive validation tests of the method but to test the feasibility of the approach.

Calibration error is estimated by the root mean square error of calibration (RMSEC) and prediction (RMSEP). RMSEC corresponds to the fitness function in eq. (41). The calibration points are randomly subdivided into calibration and test sets of the same size. RMSEC is the calibration error of the calibration set and RMSEP is obtained by the independent test set. Because the degree of freedom of

the camera calibration model is rather low, RMSEP should be close to RMSEC, i.e., overfitting should not be a significant problem.

Nikon D200 with a Nikon AF-S DX Zoom-Nikkor 18–135mm f/3.5–5.6G IF-ED lens was used for imaging. The setups of the exposure were 1/30s f/14.0 ISO500, i.e., a rather small aperture was used in order to easily attain good focus. The camera has an RGB CCD cell with a resolution of 3872 × 2592 pixels, but now a smaller subregion of the image containing only the calibration pattern is used. The dimensions of the image sensor are 23.6 × 15.8 mm.

The calibration pattern was imaged several times using fixed optical setups of the camera. Thus the images should differ only by their specific pixel noise and by some vibration of the camera and the calibration object. Lighting, at least, is considered a significant source of pixel noise.

Table 1 gives the camera parameter values and the calibration errors after the initial direct estimate and the two iterative optimization steps. Note that $z_0$ is fixed to 20 and seven other parameters to 0 in the initial guess. Yet $z_0$ is allowed to change in the optimization phases, even though it correlates with $D_u$ and $D_v$ as shown in eq. (43). Conversely, $f$ is always fixed to 1. Moreover, recall that $p_1$ and $p_2$ are fixed to 0 until the final optimization phase.

In Figure 22, the localized landmarks and the image coordinates given by the camera model are shown on top of an image of the calibration pattern. The image is 3016 pixels in width and 1032 pixels in height. The results (Table 1 and Figure 22) show that the initial estimate captures the translation vector ($x_0$, $y_0$, $z_0$), rotation angle $\kappa$, and the principal point ($u_0$, $v_0$) rather accurately. Moreover, the scale factors $D_u$ and $D_v$ are estimated well on average. However, the initial estimate assumes that $D_u$ and $D_v$ were equal, i.e., that the aspect ratio was 1. In reality, $D_v/D_u$ was estimated to be about 1.028, i.e., the pixels seem to be 2.8 % wider in the direction of the $v$-axis than of the $u$-axis. However, according to the datasheet, the pixels should be almost square.

Although the camera parameters were initially quite accurately estimated, calibration error was as large as 24 pixels, which corresponds to about 1 % of the size of the image. The error increases as the radial distance from the principal point increases, as can be seen from Figure 22 (a). When the three parameters of the radial distortion model were optimized, calibration error was reduced to about 1.6 pixels. The inclusion of the tangential distortion model reduced RMSEC further by 0.50 pixels and RMSEP by 0.16 pixels. Figure 22 (b–c) shows that the calibration error is not significantly position dependent anymore.

**Table 1.** Values of the camera parameters and calibration errors after the initial guess, the first optimization phase, and the final optimization phase. Angles are given in radians.

| Parameter | Initial | Intermediate | Final |
|---|---|---|---|
| $x_0$ | 0.806 | 0.826 | 0.809 |
| $y_0$ | –3.020 | –2.904 | –2.907 |
| $z_0$ | 20 | 19.565 | 19.587 |
| $\omega$ | 0 | 0.0091 | 0.0073 |
| $\phi$ | 0 | 0.0251 | 0.0093 |
| $\kappa$ | 3.140 | 3.132 | 3.135 |
| $\kappa_1$ | 0 | –0.0097 | –0.0100 |
| $\kappa_2$ | 0 | –0.00033 | 0.0039 |
| $\kappa_3$ | 0 | 0.00076 | –0.0027 |
| $p_1$ | 0 | 0 | 0.00075 |
| $p_2$ | 0 | 0 | –0.0057 |
| $D_u$ | 1620.1 | 1597.8 | 1597.8 |
| $D_v$ | 1620.1 | 1642.5 | 1642.5 |
| $u_0$ | 516 | 516.6 | 516.5 |
| $v_0$ | 1508 | 1508.4 | 1508.4 |
| RMSEC [pixels] | 24.0 | 1.62 | 1.12 |
| RMSEP [pixels] | 23.9 | 1.69 | 1.53 |

It is interesting to compare the camera parameters obtained with and without the tangential distortion model. It seems that $z_0$, $\kappa$, the scale factors, and the principal point obtained in the first iterative step do not change significantly in the final step. The other camera parameters, on the contrary, probably have some correlations with the tangential distortion parameters.

Table 2 lists the camera parameters and calibration errors for three images. Due to specific imaging noise, the measured calibration points should have some measurement noise but the underlying camera parameters should be equal for each image. The results indicate that the ranges ($\Delta$) of the obtained angles are: $\Delta\omega$ = 0.9°, $\Delta\phi$ = 1.2°, and $\Delta\kappa$ = 0.8°. The distortion parameters mainly change from image to image. Conversely, the translational parameters, scale factors, and the principal point are obtained robustly. The variations of the rotation angles and the distortion parameters may be explained by their intercorrelations as was revealed already from Table 1.

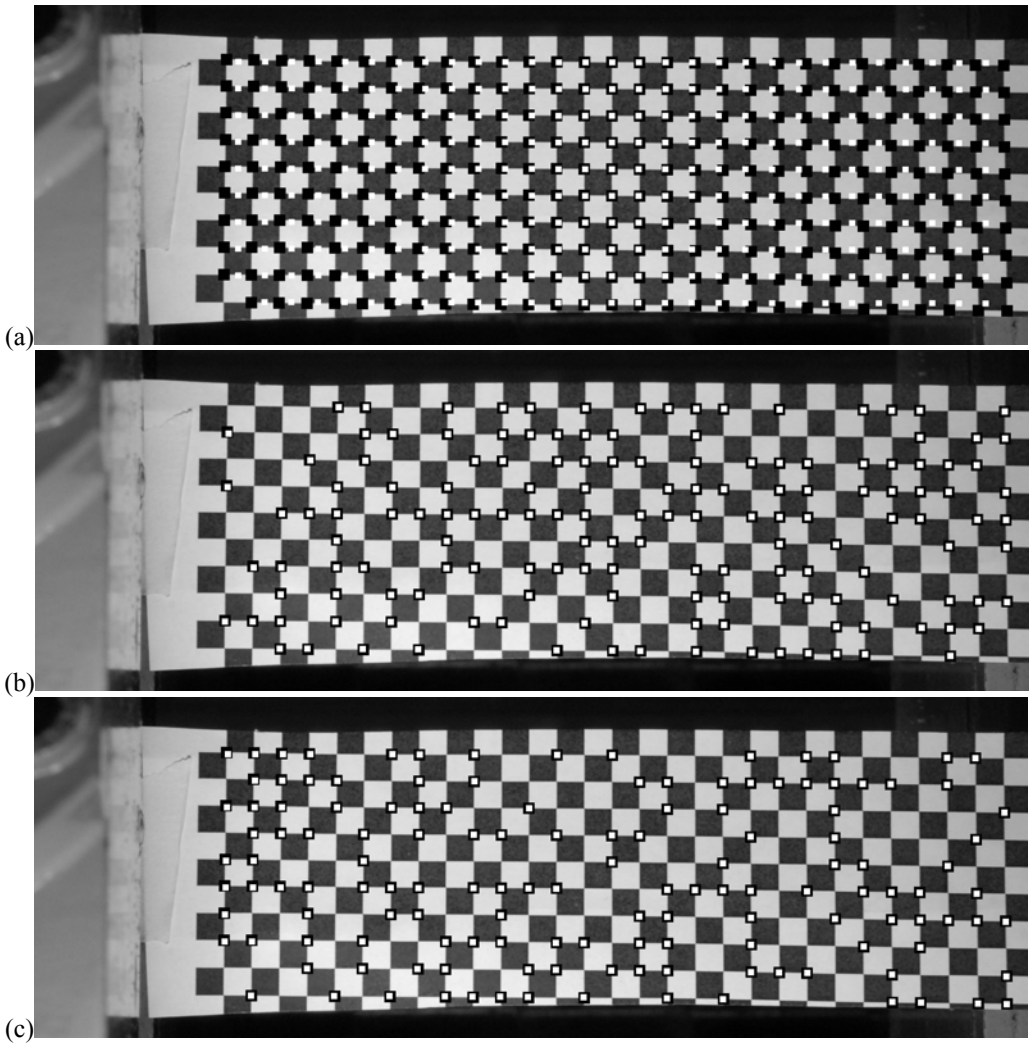



**Figure 22.** Localized (black) and projected landmarks (white) after initial guess (a) for both calibration and test sets, and after full optimization for the calibration set (b) and for the test set (c).

The most important factor of camera calibration is usually the amount of prediction error, RMSEP, or the error of back-projection. Now RMSEP ranged from 1.33 to 1.53 pixels. Hence, the calibration approach can be considered quite robust as for accuracy of measurements. Furthermore, one can notice that RMSEP > RMSEC, i.e., some overfitting also occurs consistently. Nonetheless, overfitting should not be regarded as a major drawback. On the other hand, if neural networks are used, overfitting is usually a more severe problem that must be somehow addressed.

As a conclusion, the calibration procedure localizes the landmarks and computes the corresponding world coordinates automatically. Moreover, it can manage with missing landmarks. The camera model is able to compensate for the radial and

tangential distortions. However, the camera parameters are not calibrated in absolute units but a normalized image plane with $f = 1$ is used.

**Table 2.** Camera parameters and calibration errors after final optimization for three different images obtained using the same camera setups. Angles are given in radians.

| Parameter | Image 1 | Image 2 | Image 3 |
|---|---|---|---|
| $x_0$ | 0.809 | 0.809 | 0.808 |
| $y_0$ | –2.907 | –2.911 | –2.910 |
| $z_0$ | 19.587 | 19.596 | 19.593 |
| $\omega$ | 0.0073 | –0.0090 | –0.0049 |
| $\phi$ | 0.0093 | –0.0122 | –0.0112 |
| $\kappa$ | 0.0066 | –0.0077 | –0.0069 |
| $\kappa_1$ | –0.0100 | –0.0171 | –0.0131 |
| $\kappa_2$ | 0.0039 | 0.0069 | 0.0079 |
| $\kappa_3$ | –0.0027 | –0.00094 | –0.0052 |
| $p_1$ | 0.00075 | –0.00094 | 0.0013 |
| $p_2$ | –0.0057 | –0.0050 | –0.0053 |
| $D_u$ | 1597.8 | 1597.5 | 1598.3 |
| $D_v$ | 1642.5 | 1642.8 | 1642.0 |
| $u_0$ | 516.5 | 516.5 | 516.7 |
| $v_0$ | 1508.4 | 1508.2 | 1508.3 |
| RMSEC [pixels] | 1.12 | 1.24 | 1.32 |
| RMSEP [pixels] | 1.53 | 1.36 | 1.33 |

### *2.3.5 Back-projection*

The forward projection using the camera model maps 3D world coordinates to 2D image coordinates. It does not have a unique inverse mapping but each image point spans a line-of-sight, each point of which can be the pre-image scene point. However, once the distance $z_w$ of the scene point $\mathbf{X}$ (or other constraints) is known, the coordinates of the scene point can be computed using a pseudo-inverse projection mapping: $A \rightarrow D \rightarrow I \rightarrow C \rightarrow W$.

Mapping $A \rightarrow D$ is straightforward, because affine transformation is invertible:

$$\mathbf{u}_d = \begin{pmatrix} u_d \\ v_d \\ 1 \end{pmatrix} \cong \begin{pmatrix} D_u & s & u_0 \\ 0 & D_v & v_0 \\ 0 & 0 & 1 \end{pmatrix}^{-1} \begin{pmatrix} u_a \\ v_a \\ 1 \end{pmatrix}. \quad \textbf{(45)}$$

Mapping $D \rightarrow I$ is problematic. Inversing eq. (34) gives that:

$$\textbf{(46)} \qquad \begin{cases} u_i = u_d - \delta u^{(r)} - \delta u^{(t)} \\ v_i = v_d - \delta v^{(r)} - \delta v^{(t)} \\ w_i = 1. \end{cases}$$

Unfortunately, the distortion components ($\delta u^{(r)}$, etc.) were originally computed using undistorted image coordinates ($u_i$, $v_i$), which are now unknown. A common approach is to approximate the distortion components using the distorted image coordinates ($u_d$, $v_d$). For example, the radial distortion components are approximated by:

$$\textbf{(47)} \qquad \begin{cases} \delta u^{(r)} \approx u_d \left( \kappa_1 r^2 + \kappa_2 r^4 + \kappa_3 r^6 + \ldots \right) \\ \delta v^{(r)} \approx v_d \left( \kappa_1 r^2 + \kappa_2 r^4 + \kappa_3 r^6 + \ldots \right), \end{cases}$$

where

$$\textbf{(48)} \qquad r^2 \approx {u_d}^2 + {v_d}^2 .$$

If the distortion coefficients are relatively small, satisfactory results are obtained by this approximation. Heikkilä (2000) gives a few more accurate approximation approaches. One is to use recursion, i.e., to obtain ($u_i$, $v_i$) from eq. (46) using ($u_d$, $v_d$) for the distortion components in the first iteration and then recursively substitute ($u_d$, $v_d$) by ($u_i$, $v_i$) in the computation of the distortion components until the parameters converge. Another alternative is to approximate the distortion components by a Taylor series about ($u_i$, $v_i$) and to correct the distortion components (see Heikkilä 2000: eqs. 9–12 for details).

Mapping $I \rightarrow C$ is from $\mathbb{R}^2$ to $\mathbb{R}^3$, and it requires some constraints to be unique. Eq. (30) gave that $u_i = \frac{x_c f}{z_c}$ and $v_i = \frac{y_c f}{z_c}$. Rearrangement gives: $u_c = \frac{x_i z_c}{f}$ and $u_c = \frac{y_i z_c}{f}$, respectively. Substituting these to eq. 25 gives:

$$\textbf{(49)} \qquad \mathbf{X}_c = \begin{pmatrix} \frac{x_i z_c}{f} \\ \frac{y_i z_c}{f} \\ z_c \\ 1 \end{pmatrix} = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{pmatrix} \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix} = \mathbf{P}\mathbf{X} ,$$

which can be inverted:

$$\mathbf{X} = \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix} = \begin{pmatrix} \mathbf{R}^{\mathrm{T}} & -\mathbf{R}^{\mathrm{T}}\mathbf{t} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{pmatrix} \begin{pmatrix} \frac{x_i z_c}{f} \\ \frac{y_i z_c}{f} \\ z_c \\ 1 \end{pmatrix} = \mathbf{P}^{-1}\mathbf{X}_c \,. \tag{50}$$

Still, $z_c$ is unknown. If $z_w = z$ is constant, i.e., all points lie in a plane parallel to the calibration grid, $z_c$ can be solved from equation:

$$\left(\mathbf{P}^{-1}\right)_{3j}\mathbf{X}_c = \left(\mathbf{P}^{-1}\right)_{31}\frac{u_i z_c}{f} + \left(\mathbf{P}^{-1}\right)_{32}\frac{v_i z_c}{f} + \left(\mathbf{P}^{-1}\right)_{33} z_c + \left(\mathbf{P}^{-1}\right)_{34} = z_w, \tag{51}$$

where $(\mathbf{P}^{-1})_{3j}$ refers to the third row (vector) of matrix $\mathbf{P}^{-1}$. Finally, $z_c$ can be solved:

$$z_c = \frac{z - \left(\mathbf{P}^{-1}\right)_{34}}{\left(\mathbf{P}^{-1}\right)_{31}\frac{u_i}{f} + \left(\mathbf{P}^{-1}\right)_{32}\frac{v_i}{f} + \left(\mathbf{P}^{-1}\right)_{33}} \,. \tag{52}$$

Finally, after substituting $z_c$ to eq. (50), $x_w$ and $y_w$ can be computed. In terms of uni-axial tensile tests, the calibration grid is attached to a test specimen when the camera is calibrated. Hence, all points of the specimen should approximately have $z_w = 0$.

# 3 OPTICAL STRAIN MEASUREMENTS BY NONRIGID IMAGE REGISTRATION

Strain measurements are perhaps one of the most common applications of image processing in the field of material engineering. Strain is a measure of deformation. It represents the relative displacements between points of an object.

The displacements can be measured with a contact, e.g., by a resistive or piezoelectric (Sirohi & Chopra 1999) strain gauges, or by mechanical extensometers. The measurement principles are simple, but contact methods have, intrinsically, some shortcomings. First, they need a reliable attachment, which may be difficult to guarantee. Second, they always affect the object itself, at least to some extent, thus skewing the results. Moreover, it is difficult to measure spatial strain distributions or strains in several directions concurrently when using contact methods.

Non-contact strain measurement methods solve the shortcomings of the contact methods. Most of the non-contact methods are based on electro-magnetic radiation. In addition to the optical methods, at least an acousto-optic method has also been tested (Kiesewetter & Schiller 1976). Scanning electron microscopes (SEM) and transmission electron microscopes (TEM) (Haque & Saif 2002) as well as scanning tunneling microscopy (Vendroux & Knauss 1998) have also been used for nano-scale tensile testing.

This chapter is organized as follows. First, applications of optical strain measurements, optical strain measurement methods, and, particularly, publications using nonrigid body image registration are briefly reviewed. Section 3.2 gives a short introduction to the experimental setups that are used when DeforMEERI is used to measure strains in uni-axial tensile tests. Section 3.3 discusses the theory and practice of nonrigid image registration. Finally, formulas to convert displacements to strains as well the concept *conversion of elongation values* and their applications are discussed in Section 3.4.

## 3.1 A review of optical strain measurements

Measurements of deformations are needed in many applications, each of which has different requirements. Therefore, several optical methods to strain analysis have emerged. Each one has its advantages and disadvantages. A large variety of spatial and strain resolutions, accuracy, and dynamic ranges is covered by the methods. Light sources also vary from method to method. For example, when x-rays are used, even strains inside material can be measured (Hemley et al. 1997;

Synnergren, Goldrein & Proud 1999). In general, this is possible if the material is optically transparent.

### *3.1.1 Applications*

Applications of optical strain measurements have been reported in many fields of engineering. Optical strain measurements have been applied, according to the literature, at least to: uni-axial (Keeler 1968) and bi-axial (Geiger & Merklein 2003) tensile tests of sheet metals, combustion engine moving belts (Berger 2002), semi-sweet biscuits in order to predict spontaneous cracks (Saleem et al. 2003), composites (Périé 2002), wood and paper (Choi, Thorpe & Hanna 1991), crimped glass wool (Bergonnier, Hild & Roux 2004), human skin (Marcellier et al. 2001), biological membranes (Malcolm et al. 2002), cortical bone (Nicolella et al. 2001), small components of electronics (Zhou & Goodson 2000), and bridges (Maaskant et al. 1997).

### *3.1.2 Methods with coherent lights*

Perhaps the most accurate method is to use interferometers to measure displacements. For example, Yi, Kim, and Kwak (2000) reported standard deviations of displacement measurement errors below 3 nm. This technique is applicable, at least, to measure small changes of thickness, i.e., the out-of-plane deformations. Kadono, Bitoh, and Toyooka (2001) have proposed a method called statistical interferometry, where both the specimen and the moving mirror of the Michelson type interferometer are made rough to obtain a speckle field, from which surface roughness, velocity, and deformation can be estimated with a high accuracy.

Coherent light is used in a few other methods, including holographic interferometry (HI), electronic speckle pattern interferometry (ESPI), and speckle correlation (Feiel & Wilksch 2000).

In holographic and speckle pattern interferometry, an object with a diffusely reflecting surface is illuminated by two coherent beams that produce a superposition of two uncorrelated speckles. In-plane displacements of the object produce a phase difference between the speckle patterns. The strain states of the object, from one point, can subsequently be estimated from the speckle pattern using image analysis. Moreover, both HI and ESPI enable to determine 3D deformations using three interferograms of different sensitivity vectors. The two methods use different strategies to estimate the phase shift (Spajer, Rastogi & Monneret 1981; Robinson 1983; Moore & Tyrer 1990; Vrooman & Maas 1991; Saleem et

al. 2003). Moore and Tyrer (1990) have devised a system to measure two in-plane interferograms simultaneously. Furthermore, the method enables to record strain fields with a good time resolution.

In laser speckle correlation, coherent light is used to produce a high-contrast speckle pattern, whose movement is analyzed by means of image correlation. By high optical magnification and using two CCD sensors imaging separate areas– i.e., two-spot geometry–the strain resolution can be as good as 10 με with a base length of 10 mm (Feiel & Wilksch 2000).

Laser speckle correlation disposes of some problems occurring with HI and ESPI, namely, sensitivity against vibration and rigid body movements of the test specimen. Moreover, the experimental setup of laser speckle correlation is much simpler. Resolution and accuracy depends on the optical magnification and the accuracy of image registration (Zagar & Kargel 1999; Feiel & Wilksch 2000). Anwander et al. (2000) have applied laser speckle correlation at high temperatures up to 1200°C.

Another approach to use coherent light in strain measurements is to attach a reflecting diffraction grating foil onto the surface of the specimen. It is presumed that the grating foil deforms as much as the specimen does. Consequently, the grating constant should change linearly with respect to the deformation component perpendicular to the grating rulings. The change of the grating constant can be measured with an optical setup, where the shift of the diffracted laser beam is measured. Subsequently, the change of the diffraction angle can be computed using knowledge of the geometry of the measurement setup (Asundi & Zhao 2000; Václavík, Minster & Houha 2002; Iqbal & Asundi 2006). The shift of the diffracted laser beam can be measured, e.g., by position-sensitive detectors (PSD) (Asundi & Zhao 2000). Iqbal and Asundi (2006) devised a multipoint diffraction strain sensor capable to measure more than a thousand points at the same time. It uses the principles of moiré interferometer, a CCD camera, and a microlens array. Wang et al. (2006) have used a similar measurement setup to measure force.

### *3.1.3 Methods with white light*

A very common method to measure strains in material tests is to apply a regular mesh, a pattern of circles, or regular stripes of prescribed dimensions onto the surface of the specimen. These landmarks are generated, e.g., by etching (Keeler 1968) or by laser ablation (Mäntyjärvi et al. 2008). Because the original dimensions of the pattern are known, the current strain state can be computed by measuring the grid pattern, e.g., with a manual profilometer. When using image anal-

ysis, it is also possible to trace the strain states (Tan, Melin & Magnusson 1992). Commercial software to measure grid patterns three-dimensionally exist (ViALUX GmbH).

In another marking method, random patterns are applied onto the specimen, e.g., by spray-painting. The analysis of random patterns always requires computerized methods, such as digital image correlation (DIC). The randomness of the pattern is expected to guarantee that every small patch of the specimen is unique.

According to Su and Anand (2003), "the idea of using digital images as a strain measurement technique, was first introduced by Peters et al. (Peters & Ranson 1982)". Another early paper, where DIC has been applied to strain measurements was written by Chu, Ranson, Sutton, and Peters (1985). It follows a common procedure in DIC based strain measurements: First, the undeformed image is subdivided into small subsets. Second, the intensities of the subimages are interpolated to obtain a continuous image function. Third, translations and their partial derivatives are estimated in subpixel accuracy using an iterative procedure. The objective is to maximize the similarity between the transformed subimage and the target image subject to an unknown deformation. The derivatives of the translations are inferred as the local deformations around the center of the subset.

Chu et al. (Ibid.) presumed implicitly that the local deformation was constant inside a subset. Moreover, the search process of the deformation parameters did not apply full-scale optimization but a step-by-step method, in which certain parameters were searched at a time while the others were fixed.

Vendroux and Knauss (1998) applied a second-order Taylor series to model the local deformation field. Rigid body rotation was also taken into account. Deformation parameters were obtained by the Newton-Raphson method, which is fast but requires a good initial guess to reach the global optimum. Because topographic data from a scanning tunneling microscope (STM) was used, it was possible to estimate the out-of-plane displacement component, too, thus leading to three-dimensional image correlation.

Later also Lu and Cary (2000) applied second-order Taylor series to model the displacement field and Newton-Raphson for optimization. They reported improvements in the accuracy. Nonetheless, the deformation field model has discontinuities between the subsets. This can be regarded as an imperfection as for the physical reality. Fortunately, continuity can be guaranteed, e.g., by B-splines (Cheng et al. 2002). Su and Anand (2003) in turn used the displacements of the eight neighboring subsets to interpolate the displacement field by a bi-cubic kernel. Again Newton-Raphson was used for optimization.

Another transformation model could be adopted from medical image registration. For example, Castellanos, Angel, and Medina (2004) suggested certain transformation functions that are based on affine transformation and circular domains to constitute globally differentiable warping model composed of local transformations.

Subpixel accuracy is quite commonly achieved by intensity interpolation, whence, e.g., Newton-Raphson is used for the search of parameters. Hung and Voloshin (2003) in turn used a pixel-by-pixel search in a restricted range followed by a quadratic surface fit for sub-pixel estimation. The proposed algorithm was considered much faster than Newton-Raphson. Interpolation can also be done in the Fourier domain (Amodio et al. 2003).

Usually monochromatic images are used, or, at least, the computations are done using images transformed to gray-scale. Yoneyama and Morimoto (2003) suggested, instead, colored random patterns for strain analysis. Sutton and McFadden (2000) developed a method and formulas to measure deformations from specimens in liquid.

Most of the papers report single camera imaging setups capable to measure in-plane strains of planar objects accurately. Recently, stereo vision techniques have also been adopted to obtain more accurate strain estimates, 3D strain measurements, and strain measurements from non-planar objects (Garcia, Orteu 2001; Garcia, Orteu & Penazzi 2002). Commercial equipment also exists, both for 2D and 3D strain measurements (GOM GmbH; LaVision GmbH).

### *3.1.4 Comparison of the methods*

The properties of the optical strain measurement methods are summarized in Table 3.

For 1D in-plane strains the most accurate method is to use a diffraction grating attached to the specimen. The obvious disadvantages of the method are: the finite number of prescribed spatial samples, the difficulties of properly attaching the grating foil, and the fact that the grating interferes with the measurements, particularly, if the tensile strength of the specimen is low.

**Table 3.** Comparison of the optical strain measurement methods. $\delta\varepsilon$ describes the attainable strain accuracy in με, i.e., in engineering strain divided by a factor of $10^6$. Spatial resolution describes how many strain samples can be measured. The last column tells which strains can be measured simultaneously. Out-of-plane strain corresponds to the relative change of thickness.

| **Method** | **Strain accuracy $\delta\varepsilon$** | **Spatial resolution** | **1D/2D/3D** |
|---|---|---|---|
| White light/random speckle correlation | $\delta\varepsilon = 2\delta_x/\Delta x^{(0)}$, where the registration error $\delta_x \approx 0.01\ldots1$ pixels depending on the SNR, and $\Delta x^{(0)}$ is the sampling interval [pixels]. E.g., if $\delta_x = 0.1$ and $\Delta x^{(0)} = 100$, $\delta\varepsilon = 1000$ με for in-plane strains. | Depends on the image resolution and the optics. Full-field measurements. | In-plane 2D strains by a single camera, 3D by stereo vision. |
| Laser speckle correlation with two-spot geometry | $\delta\varepsilon = 2\delta_x/\Delta x^{(0)}$, where the registration error $\delta_x \approx 50$ nm, and $\Delta x^{(0)}$ is the sampling interval. E.g., if $\Delta x^{(0)} = 10$ mm, $\delta\varepsilon = 10$ με. | One or a few prescribed gauge lengths and positions. | In-plane 1D strain. |
| Mesh grid | $\delta\varepsilon = 2\delta_x/\Delta x^{(0)}$, where $\Delta x^{(0)}$ is the resolution of the grid. $\delta_x$ depends on the accuracy of the mesh and image registration. | Depends on the resolution of the mesh grid. Full-field measurements. | 3D strains by stereo vision or by a stereo microscope. |
| Diffraction grating | Depends on the distance to the sensor. Increasing distance increases accuracy. In practice $\delta\varepsilon < 1$ με. | One or a few prescribed points. | In-plane 1D strain. |
| Interferometer | $\delta\varepsilon = 2\delta_x/\Delta z^{(0)}$, where $\Delta z^{(0)}$ is the original thickness of the specimen. $\delta_x \approx 3$ nm. If $\Delta z^{(0)} = 2$ mm, $\delta\varepsilon \approx 1.5$ με. | One or a few prescribed points. | Out-of-plane strain. |
| Statistical interferometry | $\delta\varepsilon = 2\delta_x/\Delta z^{(0)}$, where $\Delta z^{(0)}$ is the original thickness of the specimen. $\delta_x \approx \lambda/800 \approx 1$ nm. If $\Delta z^{(0)} = 2$ mm, $\delta\varepsilon \approx 0.5$ με. | One or a few prescribed points. | Out-of-plane strain. |
| HI and ESPI | $\delta\varepsilon = 2\delta_x/\Delta z^{(0)}$, where $\Delta z^{(0)}$ is the original thickness of the specimen. $\delta_x \approx \lambda/10 \approx 80$ nm. If $\Delta z^{(0)} = 2$ mm, $\delta\varepsilon \approx 80$ με. | A full-field method. | In-plane and out-of-plane 2D strains. 3D possible with three interferograms. |

Laser speckle correlation is another method for accurate 1D strain measurements. However, it is also limited to prescribed gauge positions and out-of-plane strains reduce its accuracy.

The change of thickness can be measured very accurately by traditional interferometers. In practice, the measurement is limited to a single sample point.

By HI and ESPI in-plane and out-of-plane strains can be measured simultaneously. The accuracy of the in-plane measurements is comparable to that of laser speckle correlation. However, the out-of-plane accuracy is app. 100 times worse in comparison to the interferometers. On the other hand HI and ESPI give full-field measurements, i.e., strains from the whole specimen can be measured simultaneously. HI and ESPI require a surface that does not depolarize the back-scattered light. Hence, careful sample preparation is needed.

The white light methods–i.e., the random speckle correlation and the mesh grid method–also have a full-field nature. Their accuracies depend primarily on image resolution and the accuracy of image registration. In theory, accuracies near laser speckle correlation could be achieved by optical magnification and several image sensors. With one camera and full-field strain measurements the accuracy is usually at least ten times worse than with the coherent light methods. On the other, the spatial resolution of, particularly, the random speckle method is superior to almost every other. The sample preparation of the random speckle method is simple, in contrast to, e.g., the mesh grid method, whose accuracy is largely affected by the accuracy of the grid. Furthermore, the experimental setup of the random speckle method is rather simple and inexpensive.

As a conclusion, different methods should be used for different purposes, depending on the requirements for accuracy, resolution, and the directions of strains to be measured. For the most accurate deformation analysis, different method could be used in parallel. In particular, the out-of-plane strain should be measured by statistical interferometry if high accuracy is required, while the full-field in-plane strains could be measured, e.g., by random speckle correlation.

## 3.2 Experimental setup

Figure 23 shows the experimental setup that was used to measure strains optically during uni-axial tensile tests. The hardware consists of a tensile test machine with a control PC, a single lens reflex (SLR) camera, a tripod, a PC for camera control and strain analysis, lights, and a specimen with a random speckle pattern.

An SLR camera was selected due to its good price-quality ratio in comparison to machine vision cameras of the same resolution. Moreover, Nikon provides easy-to-use remote-control software that was adequate for the purposes of method and software development as well as the laboratory usage during the research project. On the other hand, the temporal resolution achievable through an USB connection was as low as a few seconds, which may be inadequate for tensile tests with fast pulls or fragile material. Hence, USB2 connection or even faster machine vision data transfer interfaces may be needed in the future.

In industrial applications, the strain measurement system should be able to communicate with the other applications of the material testing cell. Therefore, e.g., the camera control software should be implemented in a way that allows automatic software or hardware control without manual interference. Camera manufacturers may supply some camera models with remote-control drivers with software interfaces, but when the camera model is changed, the control software has to be rewritten, too. Fortunately, there are also some open source software interfaces that support several camera models. For instance, gPhoto2 that runs in a large range of UNIX operating systems could be used (gPhoto).

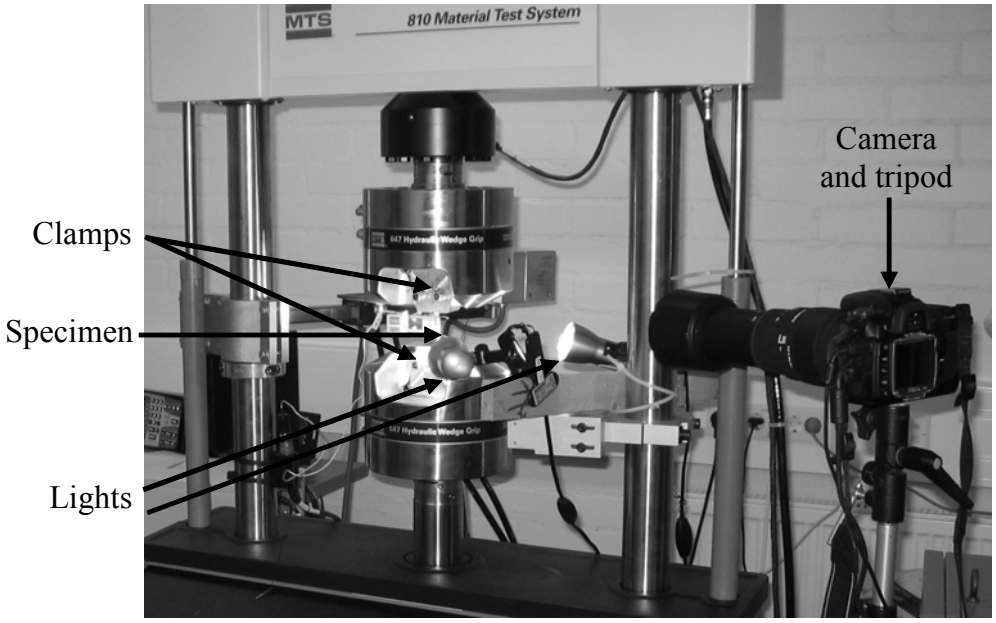



**Figure 23.** Experimental setup for strain measurements based on image during uni-axial tensile tests.

There are, at least in theory, several alternatives to apply the random patterns onto the surface of the specimen: etching, laser marking, spray-paint, splashing of paint, scratching, and ink spray printers. Usually spray-paint was used, but scratching and etching were also tested. Figure 24 shows random pattern obtained by

different marking strategies. In (a–e), black and white spray-paints are used as a primer and for speckles in various combinations. In (f), an etched pattern is combined with white paint speckles to increase randomness. In (g), only random scratches are used.

In experiments, it was noticed that the color of the specimen changes as a result of local deformations. This phenomenon inhibits the registration of the homologous points and, furthermore, the evaluation of the strain tensors, particularly, at regions where the material is necking. The change of color may originate from the changing density of the pigment particles, as demonstrated in (Koljonen & Alander 2008), or from some true changes of color of the metal or the paint material under great deformation.

It was discovered that the problem is partially avoided by first applying a uniform background coating with one color and subsequently speckling the foreground with another color. Moreover, if either black or white is used in the background and the other one in the foreground, the contrast of the speckle pattern is maximized.

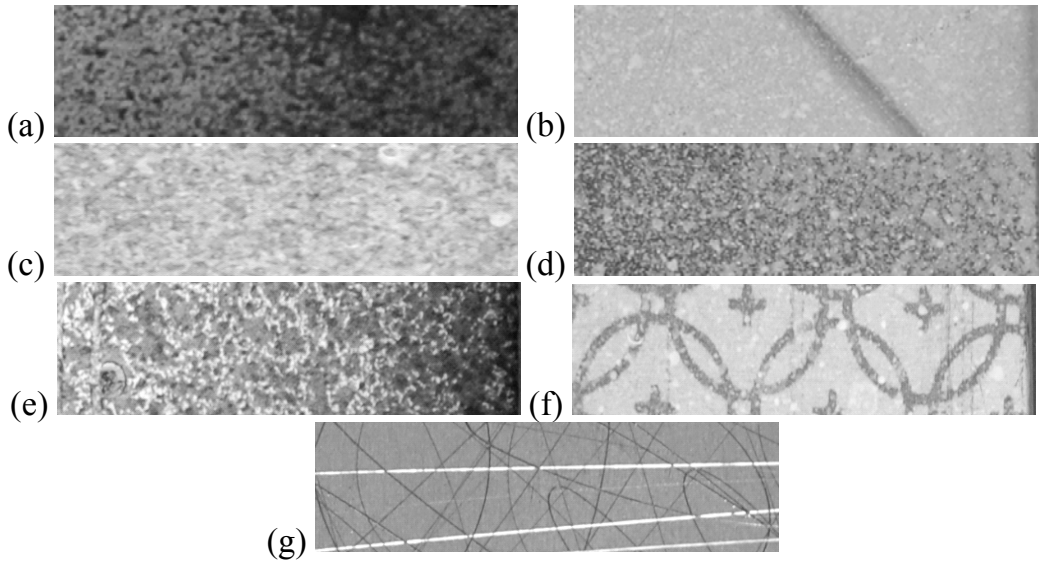



**Figure 24.** Speckle patterns obtained by different marking methods. (a) No primer, black speckles. (b) No primer, white speckles. (c) No primer, white and black speckles. (d) Black primer, white speckles. (e) White primer, black speckles. (f) Black etching, white speckles. (g) Scratches, no paint.

The change of color and the effect of the background coating are demonstrated in Figure 25. The undeformed image and a deformed image with a necking region are aligned using the displacement field measured by DeforMEERI. The images

are transformed to the hue-saturation-brightness (HSB) color space. The difference image of the brightness channels is computed and filtered with a 19 × 19 averaging Box filter.

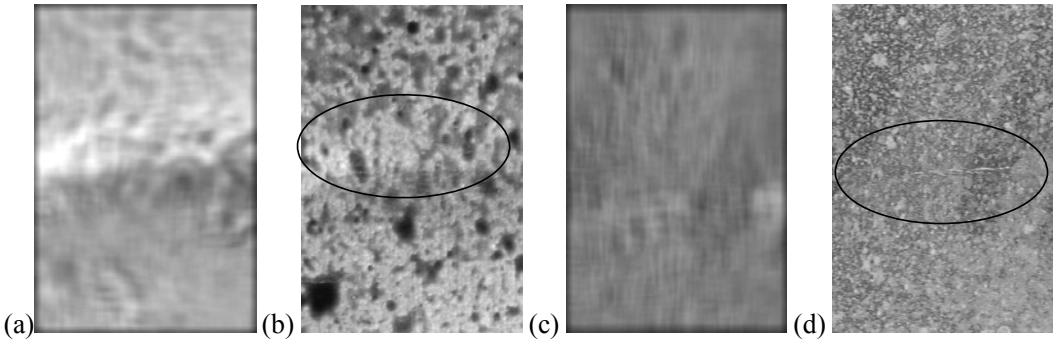



**Figure 25.** Comparison of different speckling methods by comparing the image brightness before and after deformation. (a) Difference of brightness values when only foreground speckle coating was used. (b) Deformed image, only foreground coating. (c) Difference of brightness values when using black background and white foreground. (d) Deformed image with black background and white foreground. Scale in (a) and (c): mid-gray = no change in brightness, dark = negative change, bright = positive change. Brightness differences have been magnified by a factor of 4. The necking region has been circled in (b) and (d).

Figure 25 (a) shows the difference image of a specimen with black speckle and no background coating. Panel (b) shows the corresponding deformed image aligned to match with coordinate axes of panel (a). The necking region is visible in panel (b). The corresponding area is bright in panel (a), which means that the brightness has increased at the necking region during deformation. The brightness has increased also on average.

Figure 25 (c) shows the difference image of a specimen with black background paint coating and white speckle foreground coating. Panel (d) shows the corresponding deformed image, again aligned to match with coordinate axes of (c). Now there is virtually no net change of brightness. Particularly, the necking region cannot be distinguished from the brightness difference image.

As a conclusion, background painting helps measure deformations at the necking region. On the other hand, more sample preparation is needed.

Lighting is usually one of the most important factors of successful machine vision systems. As for strain measurements, some general requirements for proper illu-

mination can be stated. Specular reflections from glossy specimen should be avoided, while the intensity should be as high as possible in order to maximize the signal-to-noise ratio. Because the homologous points travel, particularly, longitudinally during a tensile test, the illumination field should be spatially as uniform as possible. Moreover, temporal constancy of lighting is important.

Usually two incandescent spot lights, which illuminated from oblique angles (see Figure 23), were used for illumination. The uniformity of the illumination was tested indirectly in (Koljonen & Alander 2008). Some nonuniformity exists as expected, but, nevertheless, its influence on the accuracy should not be crucial. Manual scanning of images taken during a tensile test revealed that some temporal modulation of illumination also occur, probably due to the ambient light originating from fluorescent lamps.

## 3.3 Nonrigid registration and displacement field

Strain measurements based on digital image correlation adopt methods from nonrigid body registration. Once the displacement field is estimated using nonrigid registration, the strain field can be computed as will be shown in Section 3.4.

Basically, the transformation model distinguishes rigid and nonrigid body registrations from one another. Rigid body registration assumes that the shape and dimensions of the object remain constant. Nonrigid body transformation model include, in addition to some rigid body displacements, a deformation part. As to planar objects imaged from a fixed perspective, only translations, rotation, and their combinations always fulfill the requirements of rigid body transformation. Hence, every other type of transformations may implement a nonrigid body transformation.

In addition to the transformation model, issues concerning the objective function and search methods should be considered. These were already covered in Sections 2.2.3 and 2.2.5, respectively.

Image registration can be formulated as an optimization problem:

$$\mathbf{T}^* = \underset{\mathbf{T}_{\text{registration}} \in S}{\arg\min}\ h(g(\mathbf{F}_1, \mathbf{T}_{\text{registration}}), \mathbf{F}_2), \tag{53}$$

where $h$ is a *homology function*, $g$ is an operator that conducts a geometrical image transformation for image $\mathbf{F}_1$ using the given transformation function $\mathbf{T}_{\text{registration}}$ (see eq. 1), $\mathbf{F}_2$ is the other input image, and $S$ is the search space. The homology function is an idealized scalar function that summarizes the geometrical corres-

pondence of, ideally, all the homologous points of two images. In practice, $h$ cannot be estimated directly, except for some special cases, such as simulated test images. Therefore, it must be replaced by an objective function that correlates with $h$. The objective function can be, for example, correlation coefficient or the RMS difference of intensities (Zitova & Flusser 2003; Koljonen 2008).

The search space $S$ should be selected so that a good correspondence, according to $h$, can be achieved, while the complexity of the optimization problem is kept as low as possible. One extreme is to use a general transformation, eq. (1), where $\mathbf{T}$ is defined independently for each pixel or even with a higher sampling frequency. Such a general model does not contain any, explicit or implicit, assumptions of the deformation. On the other hand, its degree of freedom is infeasible for efficient optimization, and, moreover, such a model is sensitive to noise and prone to overfitting.

Another extreme is to estimate $h$ only by a sparse sampling grid. The intermediate methods model $\mathbf{T}$ explicitly or implicitly.

In many methods, a more or less sparsely sampled grid of control points is used and $\mathbf{T}$ is modeled between the control points by an appropriate method. Consequently, $h$ can be estimated between the control points, too. $\mathbf{T}$ can be modeled, e.g., by Taylor series (see e.g., Vendroux and Knauss 1998; Lu & Cary 2000), affine transformations (Hajnal, Hill & Hawkes 2001: 282), polynomials (Ibid.: 283), or using basis functions and splines of various types (Ibid.: 283–287).

Methods that use physical analogy also exist. These include elastic registration, fluid registration, finite element models (FEM), and mechanical models (Ibid.: 287–289).

Both the sparse sampling grid method and a method that uses bi-cubic interpolation of $\mathbf{T}$ were tested in this study.

In the sparse grid methods, landmarks are sampled with a uniform grid from the undeformed image. The landmarks are characterized by image templates that are localized from the sequence of deformed images using digital image correlation. The search space of the translations is limited by a first-order prediction based on the previous translations (see Koljonen, Kanniainen & Alander 2007a for details). Figure 26 shows four sample images of the search areas and the localized landmarks.

The homology is estimated only for the landmarks. In the 0th order approach, it is presumed that the deformation is negligible in the local neighborhood of the

landmark. In equal terms, the displacement field is considered to be locally constant. Consequently, only translations $t_x$ and $t_y$ of each template are subject to optimization, which implies efficient computation.

In the 1st order model, the scaling factors $c_x$ and $c_y$ are also varied, i.e., a linear (1st order) model of local displacements are used. In equal terms, the deformation field is assumed to be locally constant. Some increase in accuracy is also expected, but because the degree of freedom in optimization increases, the cost of computation increases, too.

One way to reduce complexity is to predict $c_x$ and $c_y$ instead of using full-scale optimization. Prediction can be done on the grounds of the previous strain states. All three options–0th order, 1st order with optimization, and 1st order with sole prediction–are available in DeforMEERI, but extensive tests of their performances still lack. Higher order models that use, e.g., Taylor series were already reviewed in Section 3.1.

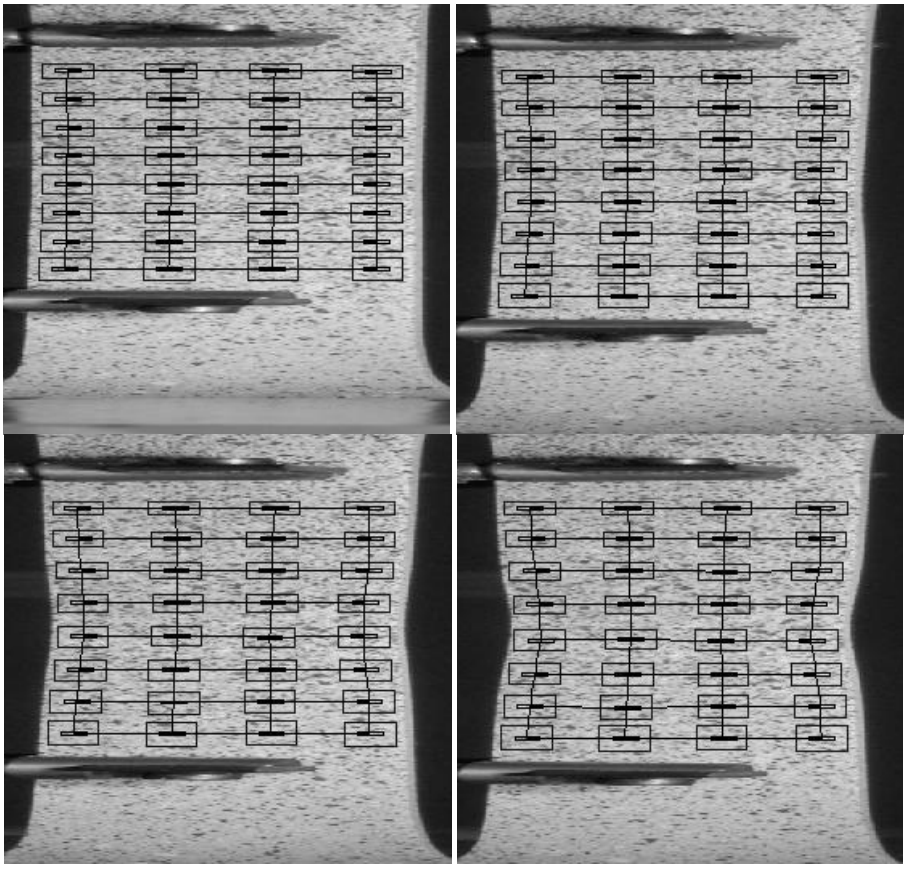


**Figure 26.** The sparse grid method using the 0th order model to localize landmarks. Four samples from a uni-axial tensile test. Smaller rectangles: landmark templates. Larger rectangles: search areas.

A 3rd order model for **T** was used in publications (Koljonen, et al. 2007; Koljonen 2008). **T** was modeled piece-wise by bi-cubic interpolation. Such a model assumes that the local deformations can be modeled approximately by quadratic polynomials. However, one should question whether the model is the most feasible one, as for physical equivalence, for the purposes of material testing.

The shape of **T** is controlled by the displacements **d** of the control points **o** that can be sampled, e.g., by a regular grid from the undeformed image (See Figure 27). For an arbitrary point $(x, y)$, the displacement $d_x$, or, similarly, $d_y$, is:

**(54)** $$d_x(x,y) = \sum_{i=0}^{3}\sum_{j=0}^{3} a_{i,j} x^i y^j \ .$$

The 16 coefficients $a_{i,j}$ are determined using the displacements of the 16 closest control points. Eq. (54) is used to determine **T** for each pixel. Consequently, $h$ can be estimated for each pixel using a similarity measure, as opposed to the sparse grid methods. On the other hand, the fitness landscape becomes more complicated due to interactions between the control points; thus complicating optimization. In (Koljonen 2008), it was demonstrated how the fitness function can be subdivided into local estimates of $h$ to boost optimization.

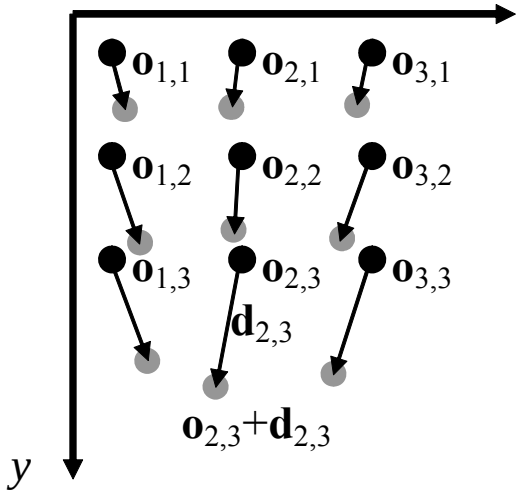



**Figure 27.** In the bi-cubic model of **T**, the image transformation is encoded as displacements **d** of the control points **o**.

## 3.4 Displacements to strains

Once the displacements field is estimated using nonrigid image body registration, strain can be computed. Suppose two landmarks whose original positions, in

world coordinates, are $\mathbf{r}_1^{(0)} = (x_1^{(0)}, y_1^{(0)})$ and $\mathbf{r}_2^{(0)}$. At time instant, or image, $k$, their positions are $\mathbf{r}_1^{(k)} = (x_1^{(k)}, y_1^{(k)})$ and $\mathbf{r}_2^{(k)}$, respectively. The displacement vectors $\mathbf{d}$ of the points are $\mathbf{d}_1(\mathbf{r}_1^{(0)}, k) = \mathbf{r}_1^{(k)} - \mathbf{r}_1^{(0)}$ and $\mathbf{d}_2(\mathbf{r}_2^{(0)}, k) = \mathbf{r}_2^{(k)} - \mathbf{r}_2^{(0)}$, respectively. By measuring the displacements of several points a displacement field is obtained.

The engineering strain ε is the change of length relative to the original length, i.e.:

**(55)** $$\varepsilon = \frac{\Delta l}{l^{(0)}}.$$

Without loss of generality, the $y$-component of strain $\varepsilon_y$ is discussed in what follows. Suppose that the $y$-component of the displacement field is $d_y(\mathbf{r}_1^{(k)})$. It can be shown that the engineering strain at landmark $\mathbf{r}_1$ at time instant $k$ is obtained by the difference/derivative of the displacement field with respect to the spatial coordinates.

**(56)** $$\varepsilon_y(\mathbf{r}_1, k) = \frac{\partial d_y(\mathbf{r}_1^{(0)}, k)}{\partial y}.$$

Let us study displacements in the direction of $y$ near landmark $\mathbf{r}_0$, while the $x$ coordinate is kept constant. Then eq. (56) can be rewritten as:

**(57)** $$\varepsilon_y(\mathbf{r}_1, k) = \lim_{\Delta y \to 0} \frac{d_y((x_1^{(0)}, \overbrace{y_1^{(0)} + \Delta y}^{y_2}), k) - d_y((x_1^{(0)}, y_1^{(0)}), k)}{\Delta y}$$

and, furthermore:

**(58)** $$\varepsilon_y(\mathbf{r}_1, k) = \lim_{y_2^{(0)} \to y_1^{(0)}} \frac{y_2^{(k)} - y_2^{(0)} - y_1^{(k)} + y_1^{(0)}}{y_2^{(0)} - y_1^{(0)}}.$$

If the original gauge length is denoted by $l^{(0)} = y_2^{(0)} - y_1^{(0)}$ and the corresponding gauge length at $k$ is denoted by $l^{(k)} = y_2^{(k)} - y_1^{(k)}$, then eq. (58) can be rewritten as:

**(59)** $$\varepsilon_y(\mathbf{r}_1, k) = \lim_{l^{(0)} \to 0} \frac{l^{(k)} - l^{(0)}}{l^{(0)}} = \lim_{l^{(0)} \to 0} \frac{\Delta l}{l^{(0)}}.$$

Eq. (59) evidently proves the connection between the derivative of displacements (eq. 56) and the engineering strain (eq. 55), namely, if a finite length is set to $\Delta y$, corresponding to $l^{(0)}$, the definition of engineering strain in eq. (55) is obtained.

In practical measurements, the gauge length $l^{(0)}$ has to be finite. In theory, the derivative of the displacement field could be computed analytically if the displacement field were determined as an analytical function, say, as a polynomial. However, even then the computation of $\Delta l$ is actually based on measurements that use a finite gauge length and interpolation of displacements. Obviously, there is always a trade-off between the spatial sampling frequency of the strain field and the accuracy of the strain estimates as described in (Koljonen, Kanniainen & Alander 2007a).


**Figure 28.** An automatic materials testing cell. An experiment, where strains are measured both with a mechanical extensometer and optically based on image.

In industrial material testing and quality control, it is a common practice to estimate the forming limit diagrams in automated testing cells, such as that in Figure 28. Usually only the longitudinal strains $\varepsilon_y$ are measured using a mechanical extensometer. Hence, the gauge length $l^{(0)}$ is fixed, although the preferred gauge length varies depending on the requirements of the customer or the type of the specimen. Conversions of elongation values are used to convert, say, the elongation of fracture $\varepsilon_y$ measured using $l^{(0)}$ to the corresponding strain value $\varepsilon_0^*$ of another gauge length $l^{(0)*}$. A feasible equation for the conversion is:

$$\varepsilon_y^* = \varepsilon_y \left( \frac{l^{(0)}}{l^{(0)*}} \right)^{\gamma}, \tag{60}$$

where the exponent $\gamma$ is determined experimentally. According to ISO2566–1:1984 standard, $\gamma = 0.4$ for certain steel alloys. However, the conversions of elongation values are problematic, because the exponent $\gamma$ seems to vary from alloy to alloy (ISO2566–1:1984; Personal communication with J. Tulonen from Rautaruukki Oyj).

When the displacement field has been obtained, say, optically, strains using gauges of arbitrary positions and length can be estimated by interpolation. The convex hull of the landmarks naturally restricts the measurement area. The possibility to place the gauge arbitrarily can be used to replace or complement the practice of conversions of elongation values. Optical measurements come with another advantage: the gauge can be placed symmetrically around the point of the fracture automatically *a posteriori*.

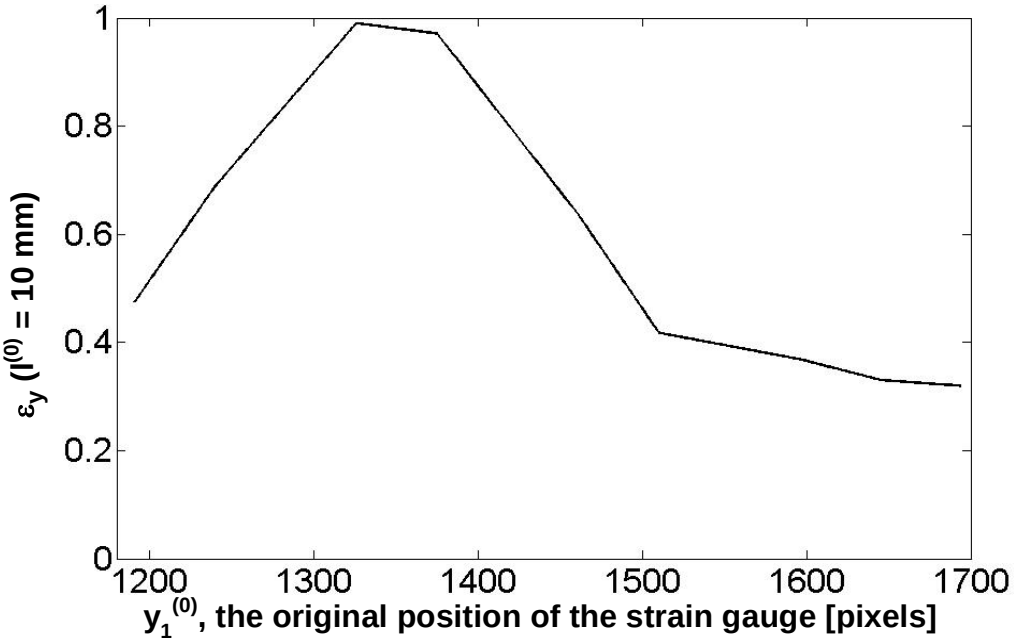



**Figure 29.** Longitudinal strain vs. the original longitudinal position of the strain gauge in affine image coordinates.

The aim of automation is achieved by simple optimization. First, the position of the strain gauge in the undeformed image, i.e., $y_0^{(0)}$ in eq. (57), is changed within an appropriate range while maintaining the gauge length $l^{(0)}$ constant. Figure 29 shows an example of the longitudinal strains obtained at different strain gauge positions. The position with the highest strain is selected and the corresponding strain value is regarded as the elongation of fracture, for instance. Figure 30 shows the position of the strain gauge at a deformed image captured just before the fracture. Note that the coordinates of Figures 29 and 30 refer to different images, between which large displacements have occurred.

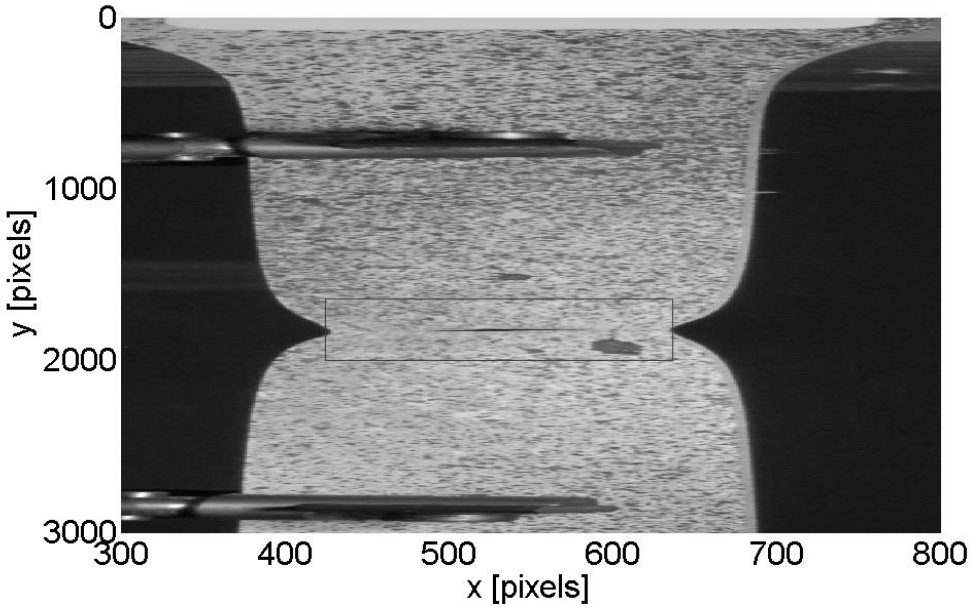



**Figure 30.** The position of the strain gauge (rectangle) at the last image before the fracture. Note that the *x*- and *y*-axes are in different scales.

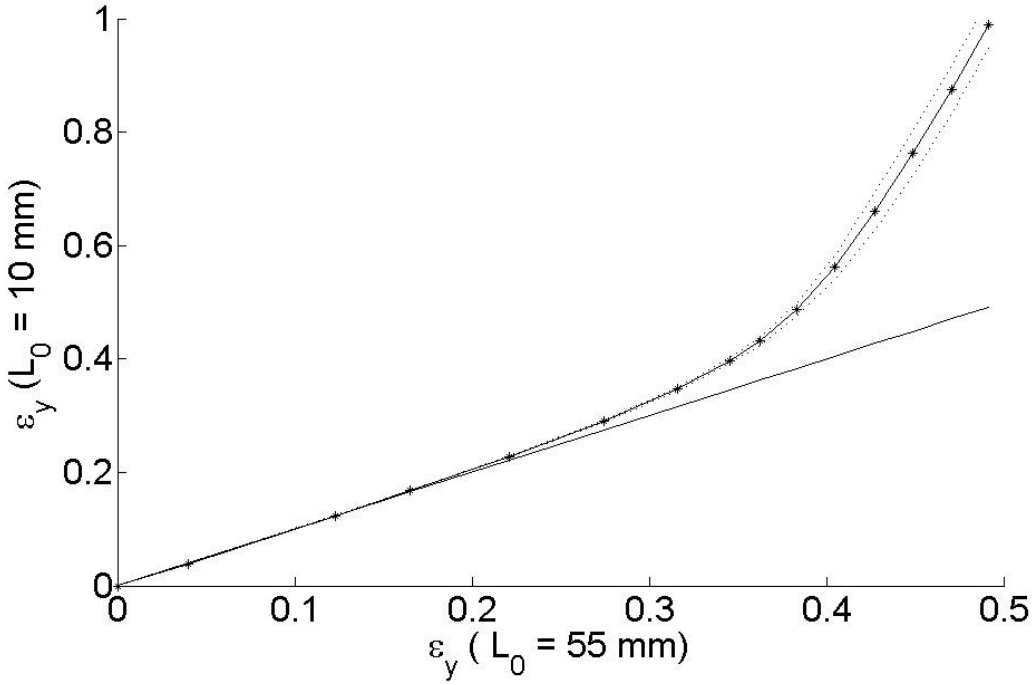



**Figure 31.** The development of the longitudinal strain during a uni-axial tensile test (solid curve) with the 3σ intervals of confidence (dotted lines). The straight line represents the unity mapping and the stars the strain samples.

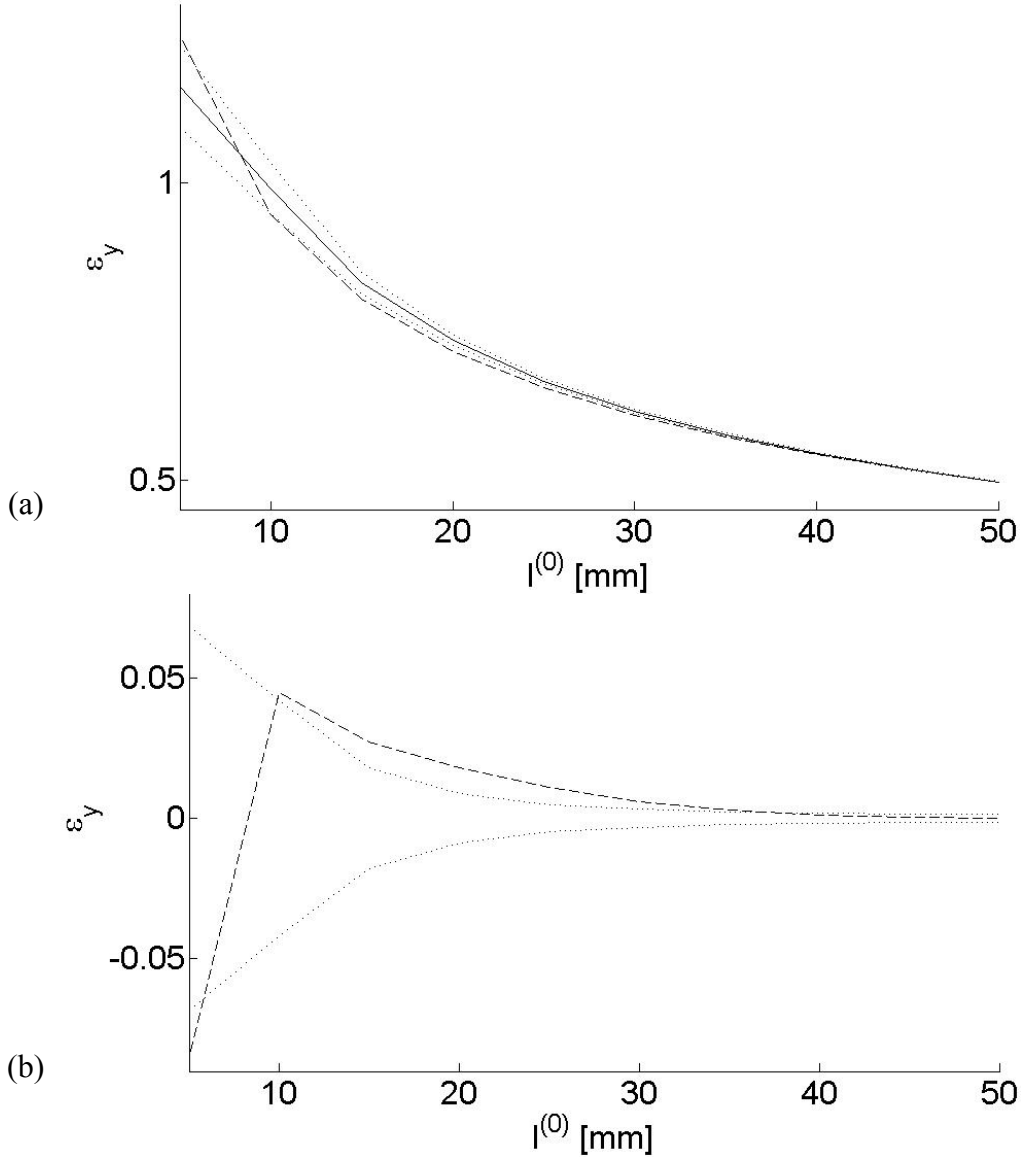



**Figure 32.** (a) Longitudinal strains measured with different gauge lengths (solid line), intervals of confidence (dotted lines), and predicted strains (dashed line). (b) The difference between the measured and predicted strains (dashed line) and the 3σ intervals of confidence of the measured strains (dotted lines). The predictions were obtained using eq. (60) and assigning: $l^{(0)} = 50$ mm and $\gamma = 0.4$.

Once the gauge position has been selected, the strain path using that strain gauge can be computed. A strain path describes the strain state during a tensile test. In Figure 31, the strain using $l^{(0)} = 10$ mm is plotted against the strain using $l^{(0)} = 55$ mm. Hence, the abscissa approximately represents the overall strain of the specimen, whereas the ordinate represents the local deformation. As a conclusion, the specimen deforms rather uniformly until the longitudinal strain exceeds app. 30

%, after which deformation starts to localize around the area of the forthcoming fracture.

In Figure 32, the accuracy of the strain conversion formula (eq. 60) is studied. Gauge length $l^{(0)} = 50$ mm is selected as the base in the conversions of elongation values. Strains using several other gauge lengths are obtained both by eq. (60) and using the method in Figure 29 directly. Comparison shows that the strains obtained by eq. (60) fit rather well the measured ones as long as the strain gauge is relatively long.

The longitudinal strains $\varepsilon_y$ shown in Figures 29, 31, and 32 are computed by averaging point-wise longitudinal strains along the $x$-axis.

# 4 EVOLUTIONARY ALGORITHMS

Evolutionary algorithms (EAs) constitute a group heuristic optimization algorithms inspired by Darwinian (Darwin 1859), Lamarckian (Lamarck 1809), and Baldwinian evolution (Baldwin 1896). In Lamarcian evolution, acquired properties of individuals can be inherited, whereas in Darwinian evolution, genetic variation is based only on recombination and mutation of the already existing building blocks of the genotype. Baldwinian evolution, also known as the Baldwin effect, assumes that individuals that have a superior learning capability have a better change to survive and reproduce in a changing environment. Instead of inheriting the learned capability, the ability to learn a certain thing is inherited and, eventually, the learned ability turns into an instinct. The effectiveness of applying these three mechanisms of evolution to evolutionary computation is compared by Julstrom (1999).

Tutorials and reviews of EAs are, e.g., (Bäck 1996; Whitley 2001).

The wide field of heuristics inspired by nature, including EAs, is still under extensive research and ongoing development. Despite their stochastic behavior, they have gradually been accepted as trustworthy tools for optimization in real world engineering characterized by "chaotic disturbances, randomness and complex non-linear dynamics" (Oduguwa, Twari & Roy 2005).

Oduguwa, Twari, and Roy (2005) survey the state-of-the-art of applying EAs in various areas of the manufacturing industry: metal forming, chemical industry, and paper industry. Moreover, two important domains of applications are recognized: scheduling and process planning, and curve optimization in computer aided design and manufacture (CAD/CAM). The latter application has some similarities to the search of the optimal deformation curve.

Next the subcategories of and related methods to EA are introduced (Section 4.1). In Section 4.2, the basic principles and important factors of EA design are discussed. Section 4.3 discusses advantages, disadvantages, and the applicability of EAs to different optimization and search tasks in general. Section 4.4 deals with multi-objective optimization and its application to the optimization of the software parameters of the strain analyzer (see also publication IV). Multimodality, real-coded EAs, and the search of the displacement field using EAs are discussed in Section 4.5.

## 4.1 Subcategories and related methods

Four originally distinct subcategories of EAs are commonly recognized: genetic algorithms (GA), evolutionary strategies (ESs), genetic programming (GP), and evolutionary programming (EP). Although GA, ES, and EP have been developed independently (according to Bäck 1996, where GP is not mentioned at all), they nowadays commonly borrow ideas from each other. Nevertheless, some distinctions and characterizations can be made.

Genetic algorithms (Holland 1975; Goldberg 1989; Forrest 1993) is perhaps the most commonly known and used evolutionary algorithm. GAs usually use binary genotype encoding and the genetic operators are applied in the genotype space.

Evolutionary strategies (Rechenberg 1971; Beyer & Schwefel 2002) are characterized by a phenotype problem representation and mutation as the primary search engine. However, recombination may also be used. If real-valued problem encoding is used, mutation is usually performed by normally distributed perturbations. Self-adaptation of the mutation parameters is used.

Genetic programming (Smith 1980; Koza 1992) attempts to evolve computer programs or algorithms. The computer program is represented in a high level using, e.g., a tree structure of operators.

In evolutionary programming, the structure of the computer program is fixed but the program parameters are optimized using the methodologies of ESs, i.e., phenotype representation and operators are used and self-adaptation of the mutation operators is preferred. Usually no recombination is applied. Sometimes EP is seen merely as a minor variant of ESs (Bäck, Rudolph & Schwefel 1993; Whitley 2001).

Other heuristic techniques related to EAs are, among many others, differential evolution (Storn & Price 1997), particle swarm optimization (Kennedy & Eberhart 1995), ant colony optimization (Dorigo, Maniezzo & Colorni 1996), and cultural algorithms (Reynolds & Sverdlik 1994).

## 4.2 Design principles

Different paradigms of EAs usually share some common features. A population of solution candidates (trials, individuals) evolves in a stochastic process by applying the principles of biological evolution to computation. After initialization of the population, new trials (offspring) are formed, e.g., by recombining and mutat-

ing the existing ones (parents). The solution candidates are evaluated by a fitness function. In the selection of the parents, a bias towards the fittest individuals is used while the worst individuals are pruned, following the principle of survival-of-the-fittest.

The flow of operations of a typical EA is summarized in Figure 33. Each iterative step is called a *generation*.

This stochastic process is prone to evolve the fitness of the candidates towards the global optimum, but convergence cannot be guaranteed. Although the conditions of convergence are difficult to study formally, attempts at a general convergence theory have been made (Rudolph 1996).

In EAs, a global exploration of the search space is accomplished implicitly. In the beginning of the optimization, the individuals cover the search space sparsely, provided that the diversity of the population is sufficient. In recombination, useful exchange of information of the global properties of the fitness landscape occurs; thus promoting global search.

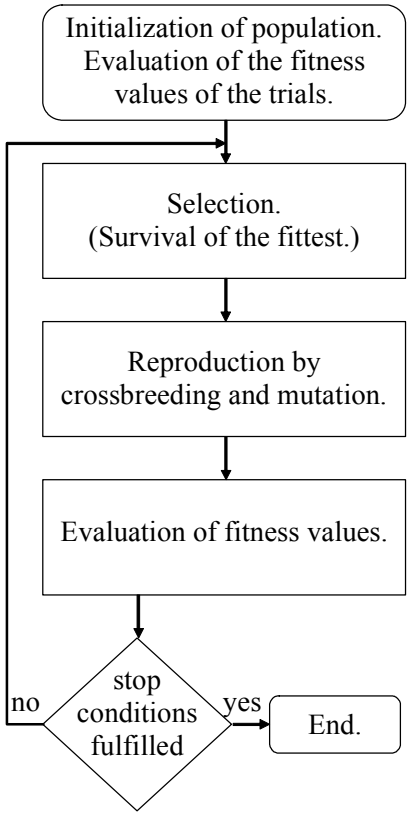



**Figure 33.** Flow chart of a typical evolutionary algorithm.

Sometimes local search methods are combined with EAs to hybrid methods in order to exploit the local properties of the fitness landscape more efficiently. These hybrids are sometimes called memetic algorithms (MAs). While the evolutionary part resembles biological evolution, the local search has an analogy to the learning of the individuals. For hybrid evolutionary algorithms and MAs see, e.g. (Moscato 1989; Goldberg & Voessner 1999; Krasnogor & Smith 2005).

The next subsections deal with the different components of EAs in more detail. In the discussion, the differences between the subtypes of EAs are mostly omitted for clarity and because the distinctions are fuzzy.

### *4.2.1 Genome*

Following the concepts of evolution in nature, most of the properties of the solution trials are encoded in their *genome*, although the concepts of memes, cultural inheritance, aging, and learning have also been utilized in evolutionary algorithms. The data structure of the genome is usually a string of variables called *chromosome*, or a tree in GP. The variables of the chromosome are called *genes*, the terminology being again borrowed from biology.

The genome is also known as the *genotype* representation of the problem. The genotype variables can be, e.g., bits, fixed-point numbers, or floating-point numbers. For example, bits are usually preferred in GAs.

### *4.2.2 Genotype-phenotype mapping*

The genotype representation does not necessarily tell directly what the solution is like. The *phenotype* of the problem in turn tells the appearance of the trials. The phenotype variables are used as the inputs to the fitness function. For instance, the genotype may be a bit string and the corresponding phenotype a displacement field.

Sometimes recognizing phenotype in the chain from the genotype to the fitness may be difficult or ambiguous. For example, instead of the displacement field, the image obtained by a geometrical image transformation could be regarded as the phenotype, because it can also be used as an input to the fitness function. However, those *free* variables, whose optimal values are searched, should compose the phenotype.

Returning to the example: Because the displacement field is the objective of the measurement and because the intensities of the transformed image are not free

parameters, the displacement field should be regarded as the phenotype while the computation of the transformation is a part of the fitness function.

The genotype-phenotype mapping can be deterministic or stochastic. A simple example of deterministic mapping is as follows: Let the bit string genome $\mathbf{b} = \{b_7, b_6, b_5, b_4, b_3, b_2, b_1, b_0\}$, $b_i \in \{0, 1\}$, represent two variables, $x_1$ and $x_2$, each with four bits using binary encoding. Then the phenotype would be:

$$\begin{cases} x_1 = \sum_{i=0}^{3} b_i 2^i \\ x_2 = \sum_{i=4}^{7} b_i 2^{i-4}. \end{cases} \tag{61}$$

The phenotype would change essentially if the mapping were done using, e.g., Gray coding. A simple example of a stochastic genotype-phenotype mapping can be found in (Koljonen, Mannila & Wanne 2007), where the genotype of a set of point were encoded using the expectation values, variances, and the correlation of 2D Gaussian distributions.

### *4.2.3 Fitness function*

Once the phenotype of an individual has been derived, its fitness can be evaluated. The fitness function should rank the individuals according to their optimality. A good fitness function correlates strongly with the underlying objectives of the optimization problem while being computationally inexpensive. Thus in complex problems, the fitness function may differ from the ideal objective function due to computational issues. The fitness function may also be vector-valued, e.g., in multi-objective optimization.

In simple cases, an analytical fitness function is available. For instance, the fitness of the phenotype in eq. (61) could be a function:

$$f : \mathbf{A} \times \mathbf{A} \mapsto \mathbf{R}, \tag{62}$$

where $\mathbf{A} = \{0, 1, \ldots, 15\}$.

On the other hand, the fitness function may be very complex or stochastic. For example, when the displacement field is optimized in image registration, the fitness function includes a geometrical image transformation and comparison of images. The similarity of images is finally summarized in a scalar fitness value

(Koljonen et al. 2007; Koljonen 2008) or sometimes in a matrix of fitness values (Koljonen 2008).

Overall, the selection of the genotype-phenotype mapping and the fitness function has a great influence on the convergence of the optimization problem. Usually several essentially dissimilar options exist. Hence, preliminary tests are usually done for comparison. Moreover, factors like the accuracy and robustness of the optimization as well as the computational efficiency affect the selection.

### *4.2.4 Genetic operators*

Genetic operators evoke the dynamics of the population – i.e., evolution. The dynamical properties of EAs are controlled by the selection of appropriate genetic operators as well as by the internal parameters related to the operators and, e.g., to the population size. Perhaps the most important factor to control is the genetic diversity of the population. In general, the more diverse the population is the more global the exploration of the search space. Vice versa, a lower diversity usually implies that the search is more local and thus faster but, on the other hand, prone to get stuck to a local optimum.

Basic genetic operators are initialization, selection, i.e., survival-of-the-fittest, and reproduction.

Initialization can be performed either randomly or using problem specific knowledge. Smart initialization can be based, for instance, on an algorithm giving approximate solutions of the displacement field (Koljonen et al. 2007; Koljonen 2008) or on the experience on some similar problems solved previously (Louis & McDonnell 2004). Nevertheless, adequate amount of diversity is needed to span the desired search space.

If the evolutionary algorithm tends to converge to near-optimal local optima, re-initialization may be used as an escape method. Using a different initial population and, perhaps, different internal parameters, the global optimum may be reached more efficiently than using a larger population and more extensive search. Re-initialization can be done either randomly or using a smart method. In a cultural genetic algorithm (CGA), a belief space gathers information of the near-optimal solutions. In re-initialization, the learned information is injected to the new population (Mantere & Koljonen 2008).

Selection is used to select parents, i.e., the individuals that are used in the reproduction process where new solution candidates, i.e., *offspring*, are generated

based on the genome of the parents. Selection is usually done following the principle of survival-of-the-fittest, borrowed from the evolution in nature. In other words, selection operators execute the pruning of the weak individuals; thus improving the average fitness of the population. However, sometimes it is possible to minimize and maximize the fitness function concurrently using reciprocal selection (Mantere 2005).

Blicke and Thiele (1996) give an overview of a variety of selection schemes: tournament selection, truncation selection, linear ranking selection, exponential ranking selection, and proportional selection. They also introduce mathematical analysis tools to evaluate and compare the selection methods. These tools analyze the distribution of the fitness values and the loss of diversity. In addition, concepts, such as reproduction rate and selection intensity, related to the selection methods are introduced.

In tournament selection, $t$ trials randomly drawn from the population are compared. Usually $t = 2$. The individual with the best fitness is selected to the next generation. These tournaments are held until an adequate number of parents have been selected. The tournament selection does not require sorting (Ibid.). However, avoiding sorting the population gives only a minor computational advantage, because the evaluation of the fitness function usually dominates the computational cost.

Truncation selection is elitist, i.e., only a prescribed fraction of the best-fit trials, i.e., the elite, are given a change to be selected. Hence, the weakest individuals are automatically pruned (Ibid.).

In ranking selection schemes, the population is sorted according to their fitness. Rank $N$ is assigned to the best trial and rank 1 to the worst one. The trials are assigned probabilities of selection proportional to their rank. The proportionality can be, e.g., linear or exponential (Ibid.).

In proportional selection, the selection probability is directly proportional to the fitness value. It is thus supposed that the fitness function is subject to maximization and that the fitness is non-negative. Moreover, the selection probabilities are sensitive to the bias component of the fitness function (Ibid.).

Reproduction has two main categories: sexual reproduction, i.e., crossbreeding or crossover; and asexual reproduction, i.e., mutation. Crossover and mutation operators can be applied both in the genotype and the phenotype space. Usually GAs use genotype operators whereas ESs use phenotype operators. Common reproduc-

tion operators are single-point crossover, uniform crossover, bit mutation, and Gaussian mutation.

In single-point crossover, two chromosomes are split into two using a common crossover point. Two new individuals are produced by recombining the split chromosomes. For example, let the parent chromosomes be $\mathbf{a} = \{a_7, a_6, a_5, a_4, a_3, a_2, a_1, a_0\}$, $a_i \in \{0, 1\}$, and $\mathbf{b} = \{b_7, b_6, b_5, b_4, b_3, b_2, b_1, b_0\}$, $b_i \in \{0, 1\}$. If the crossover point was, say, 5, then the new chromosomes **c** and **d** would be:

$$\textbf{(63)} \qquad \begin{cases} \mathbf{c} = \{a_7, a_6, a_5, b_4, b_3, b_2, b_1, b_0\} \\ \mathbf{d} = \{b_7, b_6, b_5, a_4, a_3, a_2, a_1, a_0\}. \end{cases}$$

Note that the crossover point can be inside a phenotype parameter.

In uniform crossover (Syswerda 1989), each gene is inherited from either of the chromosomes with an equal and independent probability. Uniform crossover can be represented by a crossover index, where 0 and 1 stand for the genes in the first parent and the second parents, respectively. Uniform crossover corresponds to the 'exclusive or' (XOR) operator.

Let **a** and **b** be the parents as previously. If the crossover index was, say, {0, 0, 1, 0, 1, 0, 0, 1}, the offspring **c** and the complementary offspring **d** would be:

$$\textbf{(64)} \qquad \begin{cases} \mathbf{c} = \{a_7, a_6, b_5, a_4, b_3, a_2, a_1, b_0\} \\ \mathbf{d} = \{b_7, b_6, a_5, b_4, a_3, b_2, b_1, a_0\}. \end{cases}$$

In bit mutation, random genes of an individual with a binary genotype encoding are inverted. Another common mutation operator changes the values of the phenotype parameters using a normal distribution. In general, mutation operators are used to enhance the diversity of the population whereas crossover operators recombine the existing building blocks of the parents in a new way.

Other genetic operators exist, too. For instance, aging of individuals, i.e., gradually reducing the fitness of old trials, may be useful when attempting to avoid premature converge and, particularly, when tracing a nonstationary fitness function (Ghosh, Tsutsui & Tanaka 1998).

## 4.3 Advantages and disadvantages

Evolutionary algorithms are considered to be primarily applicable to discrete, nonlinear, multidimensional, and multimodal problems. Genetic algorithms are

particularly suitable to solve discrete combinatorial problems. The ability of EAs to converge to the global optimum, provided that the EA is implemented properly, is a valuable feature. EAs have been applied successfully to multi-objective optimization and *Pareto* optimization, too (Horn, Nafpliotis & Goldberg 1994; Fonseca & Fleming 1995).

In general, EAs are multipurpose algorithms whose success relies chiefly on the representation of the problem. Consequently, many pitfalls concerning the selection of genetic encoding, genotype-phenotype mapping, and fitness function, and others, must be avoided. For example, Whitley (2001) gives a comparative review of the superiority of binary, Gray, and real-coded representations.

On the other hand, EAs are rarely the most efficient method for a given problem, but their flexibility makes them yet feasible for a variety of tasks, because the optimal optimization method is rarely known *a priori*. Whitley (Ibid.) gives some guidelines when to use EAs. Perhaps the most important advice is to exploit domain specific knowledge whenever it is possible and to test a local search method before implementing an EA.

The effectiveness of EAs can significantly be boosted by exploiting parallel and distributed computation, which greatly promotes their applicability to complex problems, which might otherwise be unsolvable in reasonable time. In addition to the computational advantage, parallelization of EAs is justified by its analogy with the counterpart in nature (Tomassini 1999). Recently, a powerful tool of computation called *grid computing* has been utilized to parallelize EAs (Lim et al. 2007).

Tomassini (1999) recognizes several ways of parallelizing evolutionary algorithms. The first approach is to run the same problem independently in several processors using different initial conditions and internal parameters of the algorithms. There is no exchange of information between the different processes, but after optimization the statistics of the solutions are gathered and processed. This approach has been utilized, e.g., in (Nordling et al. 2004).

The evaluation of the fitness is usually the most expensive operation of EAs. Hence, a natural approach is to parallelize the EA by a master–slave model, where the master process handles the population and performs the genetic operators, while the slaves are utilized only to evaluate the fitness of a given individual (Tomassini 1999). This approach has recently been studied, e.g., by Massa, Franceschini and Franceschini (2005).

Distributed EA approaches based on the *island* and *grid* models borrow ideas directly from nature. In these models, evolution occurs in subpopulations that have a loose coupling through the migration or diffusion of the individuals. In the island model, relatively large subpopulations have occasional exchange of individuals through migration (Whitley, Rana & Heckendorn 1998; Tomissini 1999). The grid approach in turn resembles cellular automata, and when applied to GAs it is also known as cellular genetic algorithm (cGA). Individuals are arranged in the form of a lattice where the probability of crossbreeding two individuals depends on their mutual distance (Tomissini 1999; Alba et al. 2007). The cGA lattice has a natural interpretation in certain application areas, for instance, in image processing (Mantere 2007).

## 4.4 Two-objective optimizing of the user parameters

The DeforMEERI software includes several parameters that should be tuned to have satisfactory performance as for accuracy and computational efficiency. Some user parameters obtain discrete values while the others are real-valued. EAs are suitable to operate in a search space with variables from different categories. Conversely, local hill-climber methods would stumble with the discrete variables when computing derivatives. Now mutation and crossover operators are performed in the phenotype space. Thus the algorithm resembles evolutionary strategies.

The global nature of EAs is efficiently utilized in this application. The objective is not to find the exact optimum but a parameter combination whose performance is close to the optimum and robust to changes in external conditions, such as SNR, is looked for. Furthermore, the optimization should be fast, at least in comparison to the manual search of the optimal parameters.

The optimization task has two objectives: minimize strain measurement error $\delta\varepsilon_z$ and complexity $t$. Pareto optimization would find all the points $(\delta\varepsilon_z, t)_p$ in the objective space that have a pre-image in the input space and are not *strictly dominated* by another point $(\delta\varepsilon_z, t)_q$, $p \neq q$. These points would form the Pareto frontier, from which a desired pareto-optimal solution could be selected. In general, vector $\mathbf{y}$ is strictly dominated by $\mathbf{x}$, i.e., $\mathbf{x}$ ) $\mathbf{y}$ if the objective is to minimize all dimensions and

$$\begin{cases} \forall i : x_i \leq y_i \\ \exists j : x_j < y_j . \end{cases} \tag{65}$$

Pareto optimization is computationally extensive. Hence, a different approach is used. The two objective measures are combined into a scalar fitness function as follows:

**(66)** $$f(\delta\varepsilon_z, t) = (\delta\varepsilon_z^p + at^p)^{1/p},$$

where $p > 0$ and $a \geq 0$ is determined using a pair of corresponding nominal error and complexity values:

**(67)** $$a = \delta\varepsilon_{z\,\mathrm{nom}} / t_{\mathrm{nom}}.$$

Values of $p$ and $a$ are selected *a priori*, and the single-objective fitness function (eq. 66) is optimized. The optimization should find a point on the Pareto frontier– which point, depends on the selection of $p$ and $a$. Theoretically, the Pareto-optimal point that has the lowest fitness measured by eq. (66) should be found.

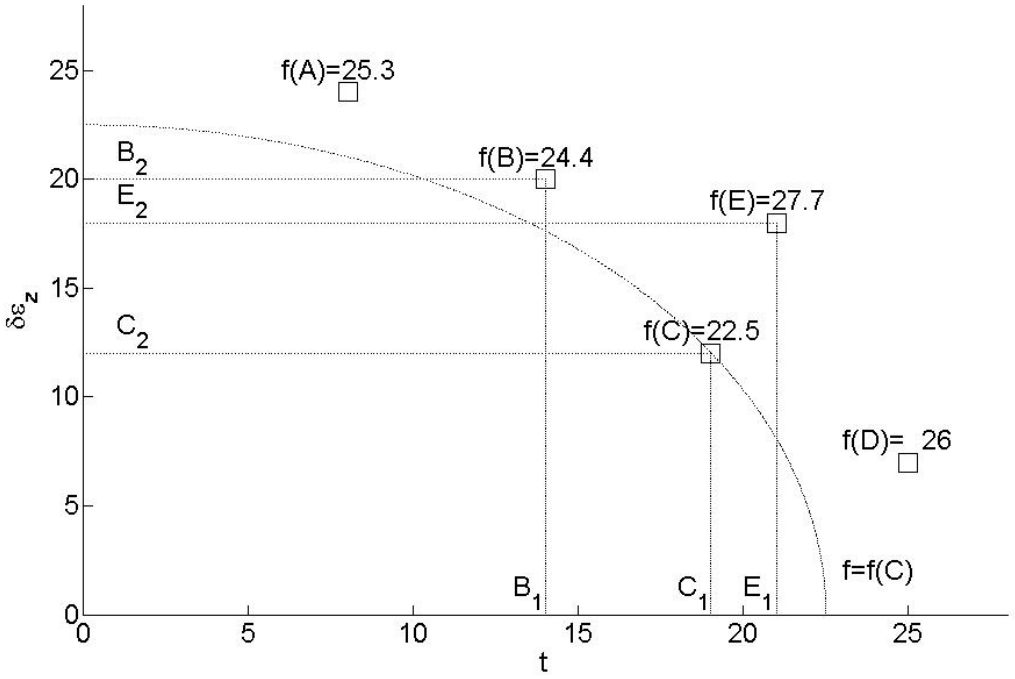



**Figure 34.** Four Pareto-optimal points in a two-objective space: **A**, **B**, **C**, and **D**. **E** is strictly dominated by **C**. Fitness values $f(\mathbf{X})$ according to eq. (66) using $p = 2$ and $a = 1$. The equi-cost arc $f = f(\mathbf{C})$ that intersects point **C**, which has the lowest fitness of the Pareto-optimal points.

Figure 34 clarifies Pareto-optimality and the use of the fitness function in eq. (66). Let point **A**, **B**, **C**, and **D** be the only Pareto-optimal points and **E** another point in $\mathbf{R}^2$ of arbitrary units. For instance, point **B** is not strictly dominated by **C**, because $C_1 > B_1$, and because is not strictly dominated any other point neither, it

is belongs to the Pareto frontier. Point **E** is in turn not a Pareto-optimal point, because it is strictly dominated by **C**. In other terms, $C_1 < E_1$ and $C_2 < E_2$.

Let us select $p = 2$ and $a = 1$ and compute the fitness values for point {**A**, …, **E**} (See Figure 34). Point **C** attains the lowest fitness in this case. Because the objective is to minimize the scalar fitness function in eq. (66), point **C** should be obtained by a perfect optimization. In general, the optimum should be one of the Pareto-optimal points as will be shown next. Without loss of generality, it is supposed that the optimum is at **C**. However, changing the values of $p$ and $a$ the optimal point changes, too.

First, if the current fitness was $f > f(\mathbf{C})$, it is evident that it can be improved, in one way or another, because the known minimum if $f(\mathbf{C})$.

Second, once attaining **C**, the fitness cannot be improved anymore, because all the other Pareto-optimal points have a worse fitness and any other point would, by definition, be strictly dominated by one of the Pareto-optimal points. Let **p** be a Pareto-optimal point and **p** ) **q**. Substituting **p** and **q** to eq. (66) and applying the inequalities of eq. (65) to $p_1$ and $p_2$ gives:

**(68)** $$f(\mathbf{p}) = (p_1^p + ap_2^p)^{1/p} < (q_1^p + aq_2^p)^{1/p} = f(\mathbf{q}).$$

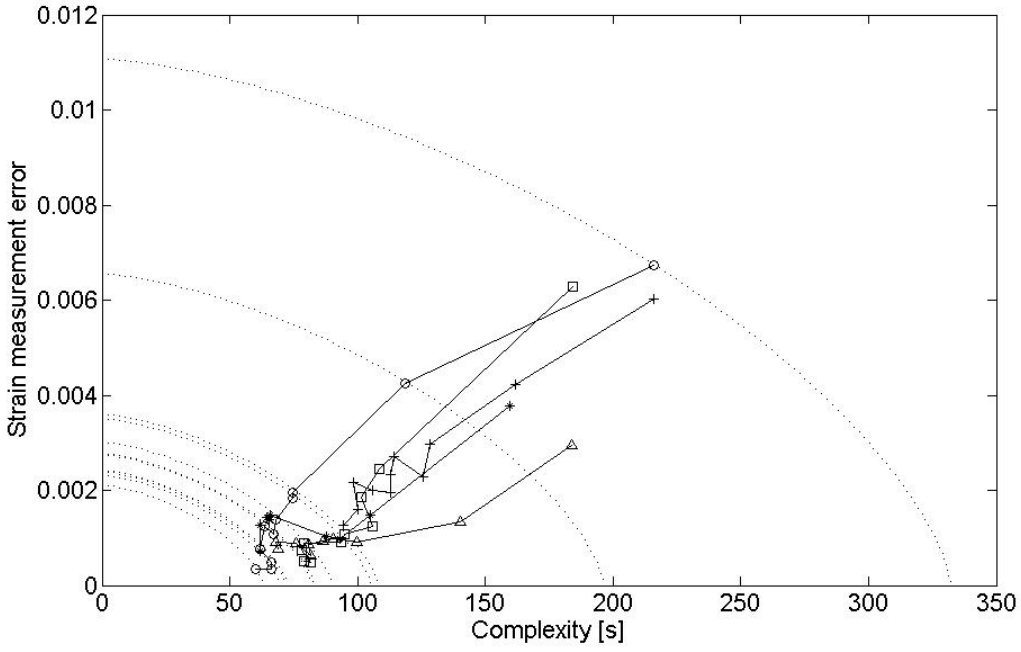



**Figure 35.** Evolvement of the two fitness components, error and complexity, in five optimization runs. Equi-cost arcs (dashed lines) with $p = 1.5$ and $a = 0.01/300$.

The user parameter optimization was first proposed and tested in publication IV. It was also utilized and tested in publication VII. Both publications reported significant improvements to the accuracy-complexity ratio in comparison to *ad hoc* tuning of parameters. In publication VII, it proved its usefulness in changing condition when the SNR was controlled manually.

Figure 35 shows examples of five optimization runs. The results show that both accuracy and complexity could be reduced and that the optimization runs converge near each other in the objective space due to the fixed selection of $p$ and $a$. On the other hand, the optimization runs were short, consisting of only 40 trials. Thus all runs had not converged yet.

## 4.5 Search of displacement field

A continuous third-order model using bi-cubic interpolation (eq. 54) of displacements was introduced in Section 3.4. In particular, Figure 27 depicted the principle of encoding the displacement field by the displacements of a finite number of control points. The objective is to minimize a fitness function that is supposed to correlate with the unknown homology function (eq. 53). In publication III, fitness was evaluated as the absolute difference of intensities whereas in publication VI it was estimated by the sum of squatted difference of intensities.

Searching for the optimal control point positions of the third-order model is a complicated task. First, the dimensionality of the input space is high. Each control point has two spatial dimensions. If a good spatial resolution and full-field measurements are needed, the total dimensionality may rise over a hundred. Second, the fitness function is potentially multi-modal, i.e., it has several local optima. Third, the input space is continues and it cannot be restricted strictly; hence, floating-point optimization with a rather high precision and a large dynamic range is needed.

The high dimensionality makes it computationally expensive to compute the derivatives that are needed in many numerical optimization methods. Multimodality in turn would cause local optimization methods to converge to a local optimum.

### *4.5.1 Multimodality*

From Figure 9, it can be seen that the fitness landscape with respect translation of templates is rugged with random speckle images. On the other hand, with larger

templates the multi-modality of the landscape is reduced. The third-order model does not contain separate templates but it form one large image with local deformations. If one control point is moved, it does not only cause a translation around the point but changes the local deformation, too. As a consequence, if one control point is translated, fitness tend to raise as can be seen from Figure 36

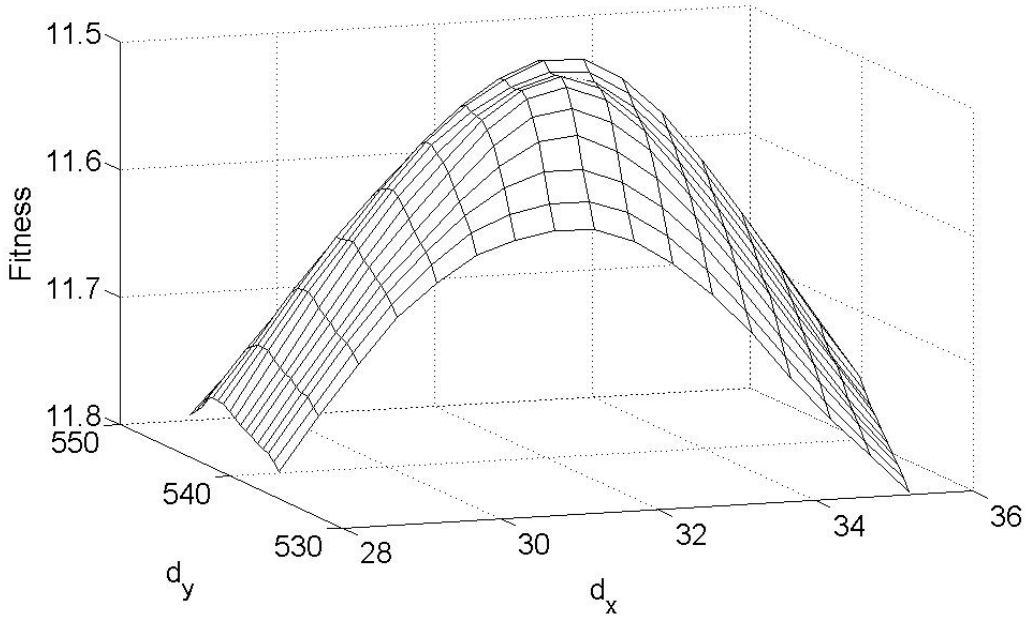



**Figure 36.** Fitness landscape when one control point is translated around its optimum while the others are fixed to their optima. Ordinate is reversed for clarity.

The locally monotony fitness landscape in Figure 36 suggests that each control point could be optimized independently. Unfortunately, the fitness landscape is affected by the other control points, too. In particular, it the control points neighboring **x** are translated, the two-dimensional optimum with respect to the displacements of **x** is moved as can be seen from Figure 37. Thus the independent optimization of each control point would lead to an iterative optimization in order to reach the global optimum.

In Figure 38, the fitness landscape of Figure 37 (a) is studied more closely around its two-dimensional optimum. A local optimum near the correct one is detected. However, this examination of does not reveal whether the point is a true local optimum. Instead, a descent towards the optimal fitness could be found in some other dimension of the control point space. Nevertheless, these flat areas of the fitness landscape are expected to make the local search inefficient.

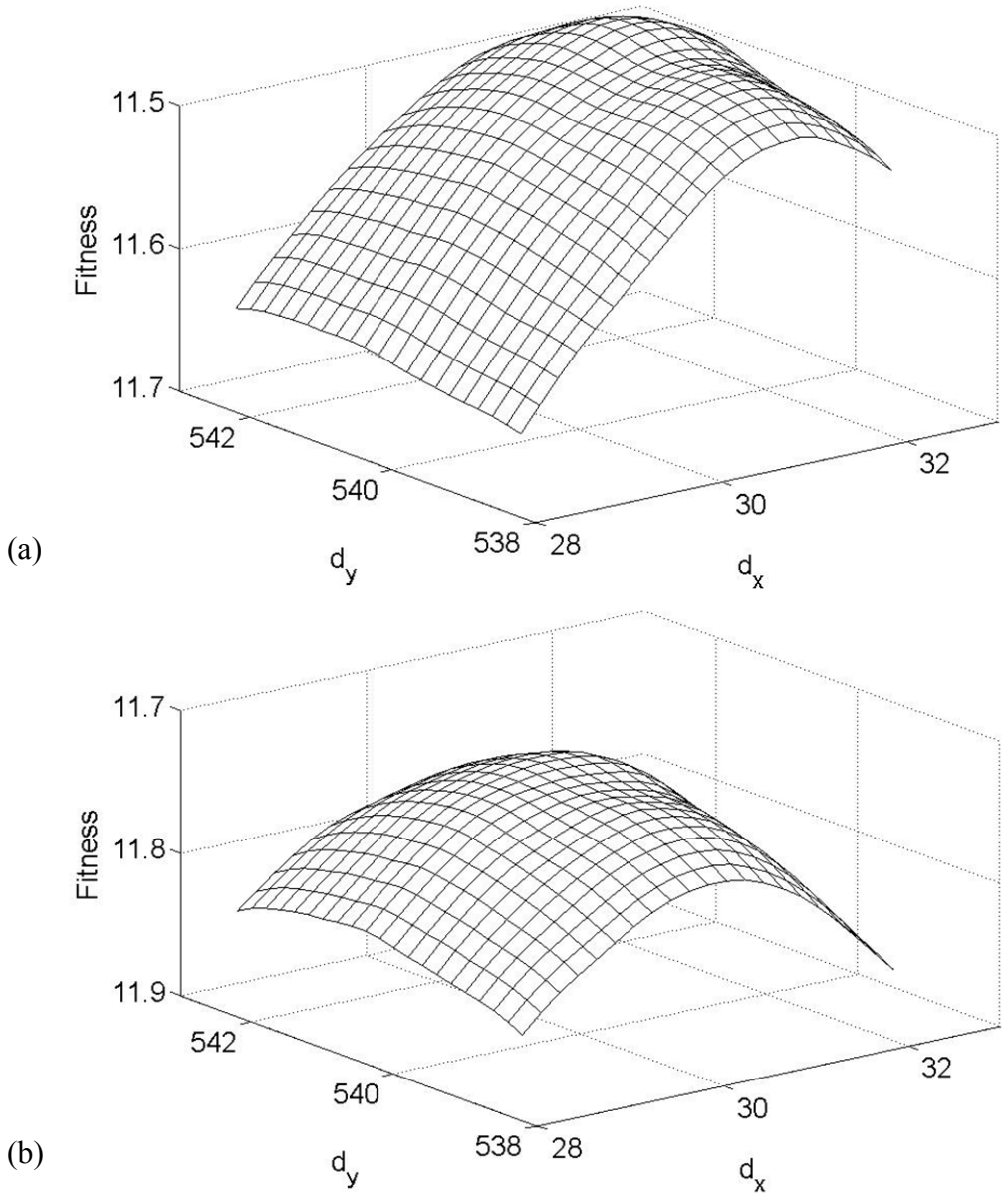



**Figure 37.** (a) Fitness landscape when one control point is translated around its optimum while the others are fixed to their optima. (b) The corresponding landscape when the eight neighboring control points are shifted by 1 pixel in *x*-direction from their optima. Ordinate in reversed for clarity.

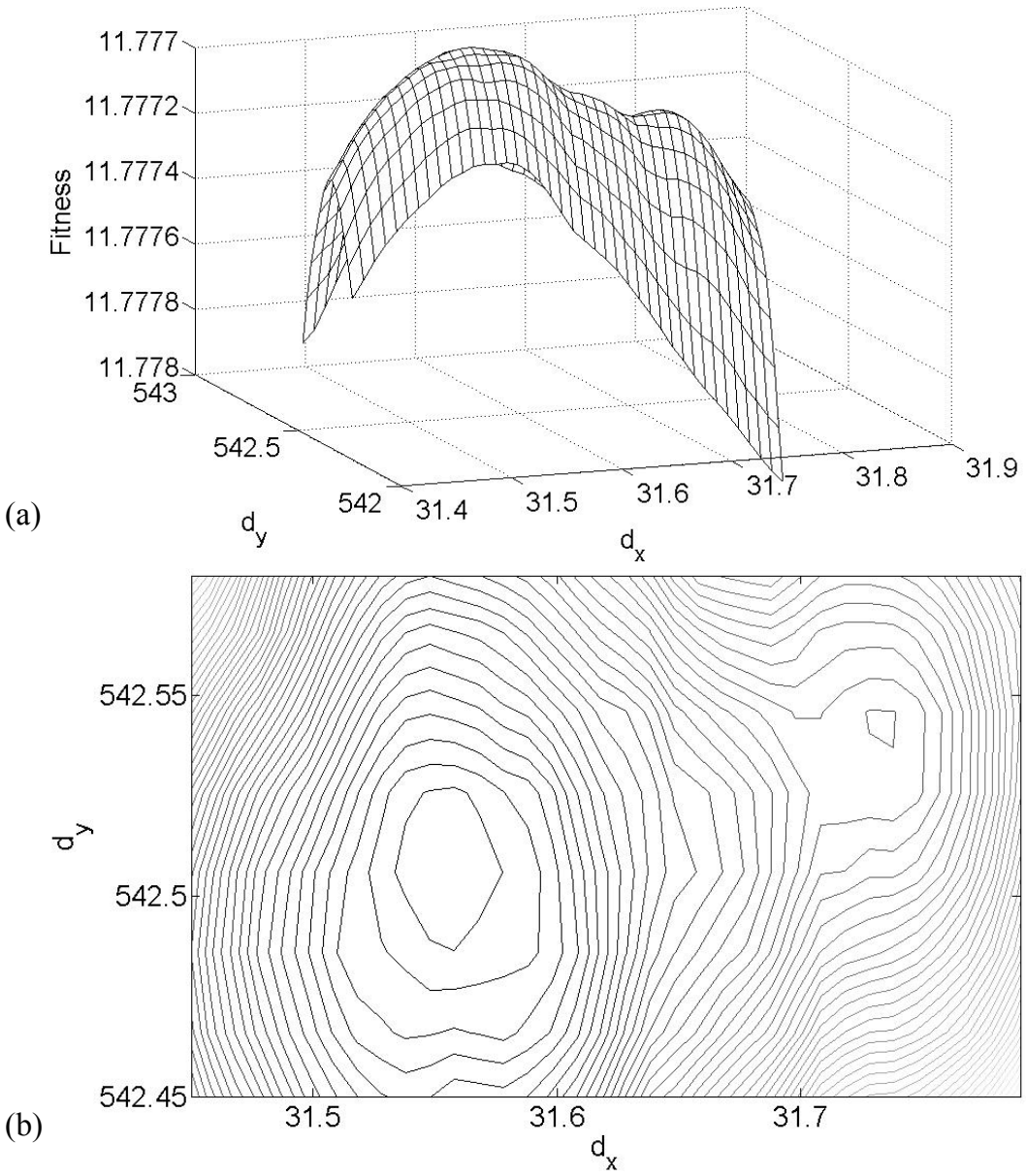



**Figure 38.** (a) Fitness landscape of Figure 37 (a) around its optimum. Ordinate in reversed for clarity. (b) Contour plot of the corresponding landscape. A local optimum near correct one is detected.

### *4.5.2 Binary and Real-coding of optimization problems*

Solving problems with a continuous domain by evolutionary algorithms differs from problems with a discrete input space. Herrera, Lozano, and Verdegay (1998) have written a comprehensive review where different coding schemes of genetic algorithms are compared as for suitability to discrete and continuous domains.

Originally genetic algorithms were proposed to use a binary alphabet and separate genotype and phenotype spaces. Genetic operators were performed only in the genotype space. The use of binary encoding has argued by two main benefits. First, it maximizes the level of *implicit parallelism* (Goldberg 1991). Second, binary GAs enable to use smaller populations than real-coded GAs, because the binary-coded chromosomes span the search efficiently in one important sense: any point of the search space can be reached by a crossover of two chromosomes of the initial population provided that the population is properly initialized (Reeves 1993).

If binary alphabets are used to encode continuous variables, the dynamic range $[a_i, b_i]$ is discretized using $L_i$ bits for each variable $i$. If the required precision is $\rho_i$, then

$$L_i = \left\lceil \log_2 \left( \frac{b_i - a_i}{\rho} + 1 \right) \right\rceil . \quad \textbf{(69)}$$

An advantage of this approach is the fact that precision can be determined *a priori*. On the other hand, in a multi-dimensional problem, where the dynamic range is large and a high numerical precision is required, the length of the bit string and the associated search space may be too large. For example, if 50 control points are used with a dynamic range of 10 pixels and $\rho_i$ = 0.01 pixels, $\Sigma L_i = 50 \lceil \log_2 1001 \rceil = 500$. The search space would subsequently have app. $10^{150}$ points.

The *Hamming cliff* effect related to the binary-coding may also complicate the search in a continuous domain. Hamming cliff occurs when two consecutive numbers in the phenotype space have complementary binary representations. The problem can sometimes be overcome using Gray coding (Mathias & Whitley 1994).

In real-coded evolutionary algorithm, each gene represents a variable of the problem. Thus the problem is usually encoded and the genetic operators are applied in the phenotype space. If the variables should be constrained, genetic operators usually take care of those restrictions (Herrera, Lozano & Verdegay 1998).

Real-coding has several advantages compared to binary-coding. The Hamming cliff effect is avoided intrinsically, and usually a slight change in the input parameters causes a slight change to the output function. This enables the local tuning of the solutions (Ibid.).

Further, since the genetic operators are applied in the phenotype, it is straightforward to utilize the domain-specific knowledge into genetic operators. In genotype-coded GAs, a decoding function from phenotype to genotype would be needed, which complicates the design of problem-specific genetic operators (Ibid.).

Herrera, Lozano, and Verdegay (Ibid.), review eleven crossover operators and six mutation operators suitable to real-coded GAs. Those operators are for general used without any domain-specific features.

In publication VI, a crossover and a mutation operator that utilize specific information related to image registration are introduced. In that case, the local fitness related to each control point was estimated using the *partial separability* property of the fitness function. In addition, uniform crossover, which is referred as *discrete crossover* in (Herrera, Lozano & Verdegay 1998), and uniform mutation were used.

Perhaps the worst drawback of real-coded EAs is related to the fine-tuning, i.e., the fine-tuning of the solution may be computationally expensive relative to the coarse search of a set of near-optimal solutions. Therefore, hybrid methods are commonly used. The EA is used for global exploration in order to avoid local optima while a local search strategy is used for fine-tuning (Ibid.).

As for the task to find the displacement field new local operators could be developed. One possibility could be to utilize the more or less parabolic shape of the fitness landscape around the optimum. The chromosomes of the population actually collect a lot of samples of the fitness landscape. By combining this information local derivatives or parabolic function estimates could be computed and utilized for more efficient search. However, too greedy search may lead to a local optimum anyway.

# 5 INTRODUCTIONS TO THE ORIGINAL PUBLICATIONS

Introductions to the seven articles reprinted in full at the end of this thesis are given in this chapter. Many references specific to the topics of each article are excluded from the bibliography of this thesis; thus the reader is referred to the bibliographies of the publications for additional relevant references.

## 5.1 Article I: Accuracy vs. resolution, implicit error estimate

This article introduces the imaging setup that was used in the measurements of strains during uni-axial tensile tests, including all subsequent articles. The article reviews different methods to measure strains. A method based on digital image correlation (DIC) of a regular rectangular lattice of fiducial points is introduced in more detail.

The theoretical relationship between the spatial sampling frequency of strains and the accuracy of the measurements is addressed and validated with experiments. It is proposed how the longitudinal and transverse strains can be computed with spatially coinciding sample points so that the relative change of thickness, i.e., the *z* strain, could be computed, too. A prediction scheme to make the search of the fiducial landmarks faster and more robust is introduced.

The article pays attention to the difficulties of validating strain measurements. An implicit error estimate based on deviations of strains along the transverse direction is proposed. The strain measurement principle and the implicit error estimate are tested and found feasible. The effects of some important parameters on the accuracy are tested.

## 5.2 Article II: Dynamic template size control

The focus of the second article is on improving the accuracy-complexity ratio of template matching, in general and in strain measurements. An algorithm to control the template size based on *confidence measures* is proposed: The dynamical template size control (DTSC) increases the template size individually for each template and target image if the results of the template matching are considered unreliable.

The DTSC method and the desired confidence measures are first considered theoretically and similar approaches reported in the literature are reviewed. Then several confidence measured are proposed.

Experiments are carried out using the strain measurement algorithm proposed in article I, accompanied with the DTSC algorithm. In contrast to the reported studies in the literature, both accuracy and computation complexity of the DTSC method are studied. Accuracy is estimated using the implicit approach similar to that in article I. Complexity is estimated both by the number of multiplications used in the computation of correlations and by the CPU time.

The Pareto frontier of the accuracy-complexity plane was estimated by random search. Two promising confidence measures showed up in the results.

## 5.3 Article III: Genetic algorithms applied to nonrigid body registration

This article was the first attempt to use another method, instead of direct template matching, to measure the deformation field from a pair of images. It was presumed that the deformation field was continuous. Moreover, it was assumed that an ideal geometrical transformation, which would transform the undeformed image to the deformed image, could be found.

A continuous model of displacements using cubic B-splines was used to transform an image spatially. The B-splines were encoded by their control points. A genetic algorithm was used to move the control points so that the mean absolute difference of intensities between the two images was minimized.

The template matching method was used as a reference when analyzing the accuracy of the proposed method. Results showed that GA scarcely outperformed random search and that the computational efficiency was unsatisfactory. As a conclusion, it was suggested, in order to boost the search, to use search operators that were more problem-specific.

## 5.4 Article IV: Optimization of algorithm parameters by genetic algorithms

Because the strain measurement algorithm included several free parameters that had probably also interactions, it became obvious that the parameters should be controlled more intelligently than using manual experimentation. Moreover,

sev00eral similar numerical methods require too much experience in the influence of different parameters in order to be effectively used by an occasional user.

In this article, it was proposed that the algorithm parameters could be optimized using genetic algorithms, because GAs are suitable for combinatorial problems where both continuous and discrete variables are involved. However, it is noted that extensive optimization aiming to find the global optimum would often turn out infeasible as to the CPU time. Hence, it was suggested that only sub-optimal results should be sought.

For sub-optimal results, the fitness function computed using the full registration algorithm could be replaced with an estimate of the fitness function. The estimate should be computed using a simplified, faster algorithm. Nevertheless, the fitness estimate should correlate with the true counterpart.

The strain measurement algorithm was used as a test case to study the concept of the accelerated fitness function estimation. Strain measurements were accelerated using a lower strain sampling resolution and measuring strains only in the beginning of the tensile-test. In the cost function, the implicit accuracy estimate and the CPU time were combined into a scalar using a general, adjustable approach proposed in the article. The results indicate that satisfactory sub-optimal combinations of the parameters can be found drastically faster than when using the method of trial-and-error.

## 5.5 Article V: Validation with an extensometer, the effect of lens distortions to strain measurements

Originally, the objective of this article was to validate the accuracy of the strain measurements using a widely applied and simple method – a mechanical extensometer. However, the results proved unreliable due to unknown reasons. Possible sources of errors are, at least: lack of camera calibration, unreliable or unskilled attachment of the extensometer, and wrong parameter setups of the extensometer. Later the validation measurements were repeated, but other difficulties related to the measurement range of the extensometer were encountered, and thus the error sources remain unknown.

Instead, the unexpected results lead to analyze the effects of imaging distortions analytically. Furthermore, reported methods and some conceptual aspects of validating strain measurements are discussed in general in the article.

Equations that describe how the radial distortion affects strain measurements are provided. The equations are used to calibrate a distortion model using the *apparent strains* measured from the images and the expected true strains given by the extensometer. However, it is questionable whether the results are reliable anyway. The findings of this article lead finally to implement the camera calibration procedure described in Section 2.3 in this thesis.

## 5.6 Article VI: Separable fitness function and smart genetic operators for nonrigid body registration

This article is a continuation of Article III presenting implementations of some future research ideas suggested in that article. In addition, it takes the theoretical and conceptual analysis of the proposed nonrigid body image registration method further.

The objective is to improve the computational efficiency of the registration method based on genetic algorithms. A fitness function based on partial separability and, subsequently, smart crossover and mutation operators based on the separable fitness function are proposed.

Test images were generated artificially by the method that was to be published later in Article VII. Hence, the true displacements of the test images were exactly known. Later other, more efficient smart genetic operators were developed, too, but the results have not been published yet.

## 5.7 Article VII: Artificial test images for deformation measurements

Validation of the strain measurements was discussed previously in, particularly, Articles I and V. In those articles, an implicit method and a method using mechanical extensometer were used, respectively, and other methods discussed. In general, it was recognized difficult to obtain reliable and accurate reference data to validate nonrigid body image registration methods.

In this article, the discussion of the validation methods is summarized and complemented, and a method to generate artificial test images is suggested and tested. The test image generation incorporates some dynamical properties that the true images bear:

1. The strain fields and paths measured from a true image series are used in the production of the artificial images.
2. The changes in saturation and brightness due to local deformations are modeled and applied in the image deforming process.
3. The effect of spatially nonuniform illumination is taken into account, because the test specimens move during a uni-axial tensile test.
4. Blurring of images due to, e.g., defocus of lens and vibrations, is modeled and applied to the test images.
5. The intensity of colored Gaussian pixel noise is varied.

The effect of different algorithm parameters is studied using the artificial test images. In particular, the implicit error estimate, DTSC, and the accelerated parameter optimization by GA, introduced in Articles I, II, and IV, respectively, are tested.

# 6 CONCLUSION

Computer vision methods to make deformation measurements based on image were introduced in this thesis. The introductory part put together concepts, theory, and practical details, particularly, of geometrical image transformations, rigid and nonrigid body image registration, subpixel image registration, camera calibration, strain measurements on planar objects, and evolutionary algorithms. Seven publications dealing with original research concerning nonrigid body registration, strain measurements, optimization, and testing and validation of deformation measurements based on image are attached at the end of the thesis.

The objective of this thesis was to evaluate the applicability of computer vision in measuring deformations from a specimen during a uni-axial tensile test. Both spatial and temporal characteristics of the deformation field were of interest. Thus a full-field method based on digital image correlation and random speckling of the specimen was adopted.

Overall, the objectives were achieved in many respects. An experimental setup and a program–i.e., an optical extensometer–capable to estimate strain fields from a series of images were developed. Despite the difficulties in the validation of the accuracy, the optical extensometer was found reliable and accurate enough in order to replace the mechanical one.

Although the mechanical extensometer is widely used both in laboratory and industry, it was found to have some shortcomings, specifically, an unreliable mechanical attachment leading to inaccurate target tracking, risk for mechanical wear, and the very limited spatial sampling resolution. The shortcoming can be overcome by the optical extensometer. Moreover, non-contacting measurements have no influence on the test specimen and can be used in a wide range of environmental conditions. Thus the answer to the first research question, *Can an accurate, fast, and easy-to-use optical extensometer superior to the mechanical extensometer be developed?*, is positive.

The speed and usability of the method was of primary concern. Furthermore, industrial online applications are in consideration, which sets high requirements for speed, reliability, and automation. Although there are commercial software and systems for strain measurements, they are probably inflexible and thus difficult to integrate to other data systems in an online process. However, the author has only second-hand knowledge of the commercial software; thus the comparison of the results is insufficient in this respect.

In terms of the second research question, *Which approach of nonrigid body image registration meets the requirements set for the method best?*, two basic approaches were developed. Currently, the simpler method based on distinct templates is considered superior to the continuous displacement model accompanied with global evolutionary optimization. The continuous model is still prone to overfitting and computationally too complex.

Later, however, the global optimization has been speeded up successfully by implementing more greedy genetic operators that utilize also local hill climbing. On one hand, greedy, local search strategies may lead to sub-optimal results in multimodal fitness landscapes. On the other hand, extensive optimization in noisy environments is prone to overfit the model to the data and the specific noise. Consequently, more attention should be paid to the registration model and, perhaps, regulation terms are needed in the optimization. Overall, the fitness landscapes should be studied more thoroughly in order to establish better understanding of the effect of the registration model and the optimization method on the registration accuracy and robustness.

The template based method can also utilize different search strategies. In the articles, exhaustive search and a coarse-to-fine search strategy are used. A region-of-interest is obtained by prediction in order to reduce complexity.

If no geometrical transformation is applied to the template, complexity can be reduced further and interpolation avoided. On the other hand, the templates are re-sampled, which lead to the accumulation of registration errors.

Two methods to transform the templates using scaling have been implemented, too. The first predicts the scaling factors using previous strain states. The other one uses prediction followed by exhaustive search for the longitudinal scaling factor. According to some preliminary, unpublished tests, the first one is computationally efficient and rather accurate while the other one is computationally much more complex but it reduces registration errors usually only little. With regard to the re-sampling method, transformed templates give comparable or better accuracy. However, with large deformations, the simple first-order transformation model seems to be insufficient and re-sampling is better in some cases.

The answer to the third research question, *Can compromises between computational complexity, accuracy, and resolution of strain measurements be avoided?*, is partly positive. It was argued, based on both analytical and experimental evidence, that resolution and accuracy are interconnected; hence, higher sampling frequency leads to worse accuracy. However, oversampling might be used to obtain high resolution measurements, but the computational complexity increases in

that case. Fortunately, methods to improve the accuracy-complexity ratio could be developed. These include dynamic template size control (DTSC) and accelerated two-objective optimization of the user parameters by evolutionary algorithms.

The fourth research question was: *How the accuracy of the deformation measurements can be evaluated?* Although three methods to estimate the accuracy were proposed, none of them fulfills all the requirements set for a decent validation method. Implicit error estimation was found practical when determining the intervals of confidence as well as in the control and optimization of the internal parameters. The method based on artificial test images is promising. It pays attention to the tonal changes of the images due to local deformations and nonuniform lighting. Nevertheless, it does not currently take the geometrical imaging distortions into account.

Validation using a mechanical extensometer was initially expected to give accurate reference data, because the method includes, intrinsically, all distortions and measurements noises. Nevertheless, the validation method proved unreliable. Probably some slipping of the knife-edges occurred or human mistakes were done. Moreover, one shortcoming always remains: optical and mechanical measurements cannot be taken using exactly the same measurement points.

As for the accuracy of the optical extensometer, the answer depends on many factors: accuracy of subpixel registration, accuracy of camera calibration, and the selected gauge length. In high SNR conditions, the accuracy of subpixel registration is about 0.1 pixels. Supposing that the back-projection error of camera calibration is negligible and the spatial resolution of strain measurements is 100 pixels, the relative accuracy of the strain measurement would be approximately 0.1 %. However, the practical accuracy in noisy conditions is 0.1–1 %.

This thesis also raises many new research questions and topics for the future: The accelerated parameter optimization method should be applied to other test cases. New smart genetic operators for problems that are partially separable have already been developed, but extensive tests should be done before publishing the results.

An accurate, two-step method, including direct template matching and GA based surface matching, could be developed for laboratory use. The first step could be used for rapid analysis, whereas the second step could be optional for cases, from which more accurate and high-resolution strain measurements are needed. Moreover, different methods for subpixel registration in template matching as well as different fitness functions should be tested and compared in attempts to improve the accuracy and robustness of the image registration methods.

The software and hardware still wait for prototyping as an integrated strain sensor as a part of an industrial process. It would require faster and more automated image capture and integration to other data systems.

One future direction could be to develop the testing method based on artificial images further. Can the test method using artificially generated images be made more realistic modeling, say, the imaging distortions? How could the method be applied to an iterative method called co-evolution (Floreano, Nolfi & Mondada 1998; Mantere & Alander 2002)? In competitive co-evolution, artificial species compete with each another: one could create test samples and try to search the weak points of the algorithm while the other evolves the algorithm.

# References

Alander, J. T. (1994). *An Indexed Bibliography of Genetic Algorithms in Optics and Image Processing* [Online report]. Vaasa, Finland: University of Vaasa, draft 23rd Aug. 2007. [Cited 13th Feb. 2009]. Available at: ftp://ftp.uwasa.fi/cs/report94-1/gaOPTICSbib.pdf.

Alba, E., Dorronsoro, B., Luna, F. Nebro, A. J., Bouvry, P. & Hogie, L. (2007). A cellular multi-objective genetic algorithm for optimal broadcasting strategy in metropolitan MANETs. *Computer Communications* 30, 685–697.

Amodio, D., Broggiato, G. B., Campana, F. & Newaz, G. M. (2003). Digital speckle correlation for strain measurement by image analysis. *Experimental Mechanics* 43: 4, 396–402.

Anuta, P. E. (1969). Digital registration of multispectral video imagery. *SPIE Journal* 7, 168–175.

Anuta, P. E. (1970). Spatial registration of multispectral and multitemporal digital imagery using fast Fourier Transform techniques. *IEEE Transactions on Geoscience Electronics* 8:4, 353–368.

Anwander, M., Zagar, B. G., Weiss, B. & Weiss, H. (2000). Noncontacting strain measurements at high temperatures by digital laser speckle technique. *Experimental Mechanics* 40:1, 98–105.

Arca, S., Casiraghi, E. & Lombardi, G. (2005). Corner localization in chessboards for camera calibration. In *Proceeding of International Conference on Multimedia, Image Processing and Computer Vision*. Madrid, Spain: IADAT.

Asundi, A. & Zhao, B. (2000). Optical strain sensor using position-sensitive detector and diffraction grating: error analysis. *Optical Engineering* 39:6, 1645–1651.

Bäck, T. (1996). *Evolutionary Algorithms in Theory and Practice. Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford, UK: Oxford University Press.

Bäck, T., Rudolph, G. & Schwefel, H.-P. (1993). Evolutionary programming and evolution strategies: similarities and differences. In D. B. Fogel & W. Atmar (eds.). *Proceedings of the Second Annual Conference on Evolutionary Programming*. La Jolla, USA: Evolutionary Programming Society. 11–22.

Baldwin, J. M. (1896). A new factor in evolution. *American Naturalist* 30, 441–451.

Barrow, H., Tenenbaum, J., Boles, R. & Wolf, H. (1977). Parametric correspondence and chamfer matching: Two new techniques for image matching. In *Proceedings of the 5th International Joint Conference on Artificial Intelligence*. Cambridge, San Mateo, USA: Morgan Kauffman. 659–663.

Berg, A. C. & Malik, J. (2001). Geometric blur for template matching. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Kauai, USA: IEEE. 607–614.

Berger, C. (2002). Strain measurement of moving periodic objects with optical sensors. *IEEE Transactions on Instrumentation and Measurement* 51:4, 616–621.

Bergonnier, S., Hild, F. & Roux, S. (2005). Digital image correlation used for mechanical tests on crimped glass wool samples. *The Journal of Strain Analysis for Engineering Design* 40:2, 185–197.

Beyer, H.G. & Schwefel, H.-P. (2002). Evolution strategies. A comprehensive introduction. *Natural Computing* 1, 3–52.

Blicke, T. & Thiele L. (1996). A comparison of selection schemes used in evolutionary algorithms. *Evolutionary Computation* 4:4, 361–394.

Bouchouicha, M., Khelifa, M. B. & Puech, W. (2003). A non-linear camera calibration with genetic algorithms. In *Proceedings of the Seventh International Symposium on Signal Processing and Its Applications*. Paris, France: IEEE. 189–192.

Brown, D. C. (1971). Close-range camera calibration. In *Proceedings of the Symposium on Close-Range Photogrammetry*, Urbana, USA: The American Society of Photogrammetry. 855–866.

Broyden, C. G. (1970). The convergence of a class of double-rank minimization algorithms. *Journal of the Institute of Mathematics and its Applications* 6, 76–90.

Bureay International des Poids et Mesures [Homepage]. [Cited 13th Mar. 2009]. Available at: http://www.bipm.org/.

Burger, W. & Burge, M. J. (2008). *Digital Image Processing: An Algorithmic Introduction Using Java*. Hagenberg, Austria: Springer.

Campbell, C. E. (2006). Generalized Coddington equations found via an operator method. *Journal of the Optical Society of America A* 23:7, 1691–1697.

Capel, D. & Zisserman, A. (1998). Automated mosaicing with super-resolution zoom. In *Proceeding of Computer Vision and Pattern Recognition*. Santa Barbara, USA: IEEE. 885–891.

Capel, D. & Zisserman, A. (2003). Computer vision applied to super resolution. *IEEE Signal Processing Magazine* May 2003, 75–86.

Castellanos, N. P., Angel, P. L. D. & Medina, V. (2004). Nonrigid medical image registration technique as a composition of local warpings. *Pattern Recognition* 37, 2141–2154.

Chang, R.-S. & Stavroudis, O. N. (1980). Generalized ray tracing, caustic surfaces, generalized bending, and the construction of a novel merit function for optical design. *Journal of the Optical Society of America* 70:8, 976–985.

Cheng, P., Sutton, M. A., Schreier, H. W. & McNeill, S. R. (2002). Full-field speckle pattern image correlation with B-spline deformation function. *Experimental Mechanics* 42:3, 344–352.

Choi, D., Thorpe, J. L. & Hanna, R. B. (1991). Image analysis to measure strain in wood and paper. *Wood Science and Technology*, 25:4, 251–262.

Choi, D.-h., Oh, S.-Y., Chang, H.-D. & Kim, K.-I. (1994). Nonlinear camera calibration using neural networks. *Neural, Parallel & Scientific Computations* 2, 29–42.

Chow, C. K., Tsui, H. T. & Lee, T. (2004). Surface registration using a dynamic genetic algorithm. *Pattern Recognition* 37, 105–117.

Chu, T. C., Ranson, W. F., Sutton, M. A. & Peters, W. H. (1985): Applications of digital-image correlation techniques to experimental mechanics. *Experimental Mechanics* 25:3, 232–244.

Clarke, T. A. & Fryer, J. G. (1998). The development of camera calibration methods and models. *Photogrammetric Record* 16:91, 51–66.

Coddington, H. (1829). *A Treatise on the Reflexion and Refraction of Light*. London, UK: Simpkin & Marshall.

Cohen, A. L. (1982). Anti-pinhole imaging. *Journal of Modern Optics* 29:1, 63–67.

Cotton, J. R., Winwood, K., Zioupos, P. & Taylor, M. (2005). Damage rate is a predictor of fatigue life and creep strain rate in tensile fatigue of human cortical bone samples. *Journal of Biomedical Engineering* 127, 213–219.

Cox, I. J., Kruskal, J. B. & Wallach, D. A. (1990). Predicting and estimating the accuracy of a subpixel registration algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12:8, 721–734.

Darwin, C. (1859). *On the Origin of Species*. Reprint of the 1st Edition (published 1998). UK: Wordsworth Limit.

DeJager, D. & Noethen, M. (1992). Gaussian beam parameters that use Coddington-based Y-NU paraprincipal ray tracing. *Applied Optics* 31:13, 2199–2205.

Do, Y. (1999). Application of neural networks for stereo-camera calibration. In *Proceedings of the International Conference on Neural Networks*. Washington, DC, USA: IEEE. 2719–2722.

Dorigo, M., Maniezzo, V. & Colorni, A. (1996). Ant system: Optimization by a colony cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics–Part B: Cybernetics* 26:1, 29–41.

Dubuisson, M.-P. & Jain, A.K. (1994). A modified Hausdorff distance for object matching. In *Proceedings of the 12th IAPR International Conference on Pattern Recognition*, Jerusalem, Israel: IEEE. 566–568.

Fan, Y. & Han-ling, Z. (2007). Multiresolution 3D image registration using hybrid ant colony algorithm and Powell's Method. *Journal of Electronics & Information Technology* 29:3, 622–625.

Faugeras, O. (1993). *Three-Dimensional Computer Vision. A Geometric Viewpoint.* Cambridge, USA: The MIT Press.

Feiel, R. & Wilksch, P. (2000). High-resolution laser speckle correlation for displacement and strain measurement. *Applied Optics* 39:1, 54–60.

Fitzpatrick, J. M., Grefenstette, J. J. & van Gucht, D. (1984). Image registration by genetic search. In *Proceedings of the IEEE SoutheastCon*. Louisville, USA: IEEE. 460–464.

Fletcher, R. (1970). A new approach to variable metric algorithms. *Computer Journal* 13, 317–322.

Floreano, D., Nolfi, S. & Mondada, F. (1998). Competitive co-evolutionary robotics: from theory to practice. In *From Animals to Animats V. Proceedings of the Fifth International Conference on Simulation of Adaptive Behavior*. Cambridge, USA: MIT Press. 515–524.

Fonseca, C. M. & Fleming, P. J. (1995). An overview of evolutionary algorithms in multiobjective optimization. *Evolutionary Computation* 3:1, 1–16.

Forliti, D. J., Strykowski, P. J. & Debetin, K. (2000). Bias and precision errors of digital particle image velocimetry. *Experiments in Fluids* 28, 436–447.

Foroosh (Shekarforoush), H. & Zerubia, J. B. (2002).Extension of phase correlation to subpixel registration. *IEEE Transactions on Image Processing* 11:3, 188–200.

Forrest, S. (1993). Genetic algorithms: principles of natural selection applied to computation. *Science* 261:5123, 872–878.

Freer, J. A., Beggs, B. J., Fernandez-Cangue, H. L., Chevrier, F. & Goryashko, A. (1997). Automatic intruder detection incorporating intelligent scenemonitoring

with video surveillance. In *Proceedings of Security and Detection*. London, UK: IEEE. 109–113.

Frischholz, R. W. & Spinnler, K. P. (1993). In *Proceedings of the International Symposium on Electronic Imaging Device Engineering*. Munich, Federal Republic of Germany: SPIE. 50–59.

Fuentes, L. M. & Velastin, S. A. (2006). People tracking in surveillance applications. *Image and Vision Computing* 24, 1165–1171.

Garcia, D. & Orteu, J.-J. (2001). 3D deformation measurement using stereo-correlation applied to experimental mechanics. In *Proceedings of the 10th FIG International Symposium on Deformation Measurements*. Orange, USA: International Federation of Surveyors (FIG). 50–60.

Garcia, D., Orteu, J.-J. & Penazzi, L. (2002). A Combined temporal tracking and stereo-correlation technique for accurate measurement of 3D displacements: application to sheet metal forming. *Journal of Materials Processing Technology* 125–126, 736–742.

Geiger, M. & Merklein, M. (2003). Determination of forming limit diagrams – a new analysis method for characterization of materials' formability. *CIRP Annals – Manufacturing Technology*, 52:1, 213–216.

Georgescu, B. & Meer, P. (2004). Point matching under large image deformation and illumination changes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26:6, 674–688.

Ghosh, A., Tsutsui, S. & Tanaka, H. (1998). Function optimization in nonstationary environment using steadystate genetic algorithms with aging of individuals. In *Proceedings of the 1998 IEEE International Conference on Evolutionary Computation.* Anchorage, USA: IEEE. 666–671.

Gleason, S. S., Hunt, M. A. & Jatko, W. B. (1990). Subpixel measurement of image features based on paraboloid surface fit. In *Proceedings of Machine Vision Systems Integration in Industry*. Boston, USA: SPIE. 135–144.

Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. New York, USA: Addison-Wesley.

Goldberg, D. E. (1991). Real-coded genetic algorithms, virtual alphabets, and blocking. *Complex Systems* 5, 139–167.

Goldberg, D. E. & Voessner, S. (1999). Optimizing global-local search hybrids. In *Proceeding of the Genetic and Evolutionary Computation Conference*. Orlando, USA: Morgan Kaufmann. 220–228.

Goldfarb, D. (1970). A family of variable metric updates derived by variational means. *Mathematics of Computing* 24, 23–26.

GOM GmbH [Homepage]. [Cited 13th Mar. 2009]. Available at: http://www.gom.com.

gPhoto2 [Homepage]. [Cited 13th Mar. 2009]. Available at: http://www.gphoto.org.

Hajnal, J. V., Hill, D. L. G. & Hawkes, D. J. (eds.) (2001). *Medical Image Registration*. USA: CRC Press LLC.

Haque, M. A. & Saif, T. A. In-situ tensile testing of nano-scale specimens in SEM and TEM. *Experimental Mechanics* 42:1, 123–128.

Harris, C. & Stephens, M. (1988). A combined corner and edge detector. In *Proceeding of the 4th ALVEY Vision Conference*. Manchester, UK: University of Manchester. 147–151.

Hati, S. & Sengupta, S. (2001). Robust camera parameter estimation using genetic algorithm. *Pattern Recognition Letters* 22, 289–298.

He, R.-S., Horn, C.-T., Wang, H.-J. & Hwang, S.-F. (2006). Deformation measurement by a digital image correlation method combined with a hybrid genetic algorithm. *Key Engineering Materials* 326, 139–142.

Heikkilä, J. (2000). Geometric camera calibration using circular control points. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22:10, 1066–1077.

Hemayed, E. E. (2003). A survey of camera self-calibration. In *Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance*. Miami, USA: IEEE. 351–357.

Hemley, R. J., Mao, H.-k., Shen, G., Badro, P. G., Hanfland, M. & Häusermann, D. (1997). X-ray imaging of stress and strain of diamond, iron, and tungsten at megabar pressures, *Science* 276:5316, 1242–1245.

Herrera, F., Lozano, M. & Verdegay, J. L. (1998). Tackling real-coded genetic algorithms: Operators and tools for behavioural analysis. *Artificial Intelligence Review* 12, 265–319.

Holland, J. (1975). *Adaptation in Natural and Artificial Systems*. Ann Arbor, USA: University of Michigan Press.

Horn, J., Nafpliotis, N. & Goldberg, D.E. (1994). A niched Pareto genetic algorithm for multiobjective optimization. In *Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence*. Piscataway, USA: IEEE. 82–87.

Hothmer, J. (1958). Possibilities and limitations for elimination of distortion in aerial photographs. *The Photogrammetric Record* 2:12, 426–445 and 3:13, 60–78.

Hu, W., Tan, T., Wang, L. & Maybank, S. (2004). A survey on visual surveillance of object motion and behaviors. *IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications and Reviews* 34:3, 334–352.

Hung, P.-C. & Voloshin, A. S. (2003). In-plane strain measurement by digital image correlation. *Journal of the Brazilian Society of Mechanical Science & Engineering* 25, 215–221.

Huttenlocher, D. P., Klanderman, G. A. & Rucklidge, W. J. (1993). Comparing images using the Hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Vision* 15:9, 850–863.

Instron [Homepage]. [Cited 6th May 2009]. Available at: http://www.instron.com.

Instron (2005). Easy set-up and consistent calibration. [News article] *Manufacturingtalk* [Cited 6th May 2009]. Available at: http://www.manufacturingtalk.com/news/ivx/ivx105.html.

Iqbal, S. & Asundi, A. (2006). Multipoint diffraction strain sensor: theory and results. *Measurement Science and Technology* 17, 2306–2312.

ISO2566–1:1984. Conversion of elongation values – part 1: Carbon and low alloy steels.

Jenkinson, M. & Smith, S. (2001). A global optimisation method for robust affine registration of brain images. *Medical Image Analysis* 5, 143–156.

Ji, Q. & Zhang, Y. (2001). Camera calibration with genetic algorithms. *IEEE Transactions on Systems, Man, and Cybernetics–Part A: Systems and Humans* 31:2, 120–130.

Jiang, J., Luk, W. & Rueckert, D. (2003). FPGA-based computation of free-form deformations in medical image registration. In *Proceedings of Field-Programmable Technology*. Tokyo, Japan: IEEE. 234–241.

Jin, H. & Bruck, H. A. (2005). Pointwise digital image correlation using genetic algorithms. *Experimental Techniques* 29, 36–39.

Julstrom, B. (1999). Comparing Darwinian, Baldwinian, and Lamarckian search in a genetic algorithm for the 4-cycle problem. In S. Brave & A. S. Wu (eds.). *Proceedings of the Late Breaking Papers Genetic Evolutionary Computation Conference*. Orlando, USA: Morgan Kauffman. 134–138.

Kadono, H. Bitoh, Y. & Toyooka, S. (2001). Statistical interferometry based on a fully developed speckle field: an experimental demonstration with noise analysis. *Journal of Optical Society of America A* 18:6, 1267–1274.

Kakarala, R. & Hero, A. O. (1992). On achievable accuracy in edge localization. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14:7, 777–781.

Keeler, S. (1968). Circular grid system – a valuable aid for evaluating sheet metal formability. *SAE Paper* 680092, 1–9.

Kennedy, J. & Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of the IEEE International Conference on Neural Networks*. Perth, Australia: IEEE. 1942–1948.

Kiesewetter, N. & Schiller, P. (1976). The acoustic emission from moving dislocations in aluminum. *Physica Status Solidi (a)* 38:2, 569-576.

Kim, D.-K., Jang, B.-T. & Hwang, C.-J. (2002). A planar perspective image matching using point correspondence and rectangle-to-quadrilateral mapping. In *Proceedings of the Fifth IEEE Southwest Symposium on Image Analysis and Interpretation*. Santa Fe, USA: IEEE. 87–91.

Kim, J. & Fessler, J. A. (2004). Intensity-based image registration using robust correlation coefficients. *IEEE Transactions on Medical Imaging* 23:11, 1430–1444.

Koljonen, J. & Alander, J. T. (2008). Deformation image generation for testing a strain measurement algorithm. *Optical Engineering* 47:10. 107202-1–13.

Koljonen, J. (2008). Partially separable fitness function and smart genetic operators for area-based image registration. In T. Raiko, P. Haikonen & J. Väyrynen (eds.). *AI and Machine Consciousness. Proceedings of the 13th Finnish Artificial Intelligence Conference*. Espoo: Finnish Artificial Intelligence Society. 4–14.

Koljonen, J., Kanniainen, O. & Alander, J. T. (2007a). An implicit validation approach for digital image correlation based strain measurements. In *Proceedings of the IEEE International Conference on Computer as a Tool*. Warsaw, Poland: IEEE. 250–257.

Koljonen, J., Kanniainen, O. & Alander, J. T. (2007b). Dynamic template size control in digital image correlation based strain measurements. In D. P. Casasent, E. L. Hall & J. Röning (eds.). *Intelligent Robots and Computer Vision XXV: Algorithms, Techniques, and Active Vision*. Boston, USA: SPIE. 67640L-1–12.

Koljonen, J., Katajarinne, T., Lönnqvist, A. & Alander, J.T. (2008). Validation of digital speckle correlation strain measurements with extensometer. In N. Asnafi (ed.). *Best in Class Stamping. Proceedings of the International Conference of International Deep Drawing Research Group*. Olofström, Sweden: IDDRG. 57–68.

Koljonen, J., Mannila, M. & Wanne, M. (2007). Testing the performance of a 2D nearest point algorithm with genetic algorithm generated Gaussian distributions. *Expert Systems with Applications* 32:3, 879–889.

Koljonen, J., Mantere, T. & Alander, J. T. (2007). Parameter optimization of numerical methods using accelerated estimation of cost function: a case study. In M. Niskanen & J. Heikkilä (eds.). *Proceedings of the Finnish Signal Processing Symposium* [CD-ROM]. Oulu, Finland: University of Oulu.

Koljonen, J., Mantere, T., Kanniainen, O. & Alander J. T. (2007). Searching strain field parameters by genetic algorithms. In D. P. Casasent, E. L. Hall & J. Röning (eds.). *Intelligent Robots and Computer Vision XXV: Algorithms, Techniques, and Active Vision*. Boston, USA: SPIE. 67640O-1–9.

Koza, J. R. (1992). *Genetic Programming*. Cambridge, USA: The MIT Press.

Krasnogor, N. & Smith, J. (2005). A tutorial for competent memetic algorithms: Model, taxonomy, and design issues. *IEEE Transactions on Evolutionary Computation* 9:5, 474–488.

Lamarck, J. B. (1809). *Philosophie zoologique*.

LaVision GmbH [Homepage]. [Cited 13th Mar. 2009]. Available at: http://www.lavision.de.

Lei, C., Wu, F., Hu, Z. & Tsui, H. T. (2002). A new approach to solving Kruppa equations for camera self-calibration. In *Proceedings of the 16th International Conference on Pattern Recognition*. Quebec, Canada: IEEE. 308–311.

Lester, H. & Arridge, S. R. (1999). A survey of hierarchical non-linear medical image registration. *Pattern Recognition* 32, 129–149.

Leung, Y. C., Chan, L. C., Tang, C. Y. & Lee, T. C. (2004). An effective process of strain measurement for severe and localized plastic deformation. *International Journal of Machine Tools & Manufacture* 44, 669–676.

Lim, D., Ong, Y.-S., Jin, Y., Sendhoff, B. & Lee, B.-S. (2007). Efficient hierarchical parallel genetic algorithms using grid computing. *Future Generation Computer Systems* 23, 658–670.

Lindberg, D. C. (1968). The theory of pinhole images from antiquity to the thirteenth century. *Archive for History of Exact Sciences* 5:2, 154–176.

Louis, S. J. & McDonnell, J. (2004). Learning with case-injected genetic algorithms. *IEEE Transactions on Evolutionary Computation* 8:4, 316–328.

Lu, H. & Cary, P. D. (2000). Deformation measurements by digital image correlation: Implementation of a second-order displacement gradient. *Experimental Mechanics* 40:4, 393–400.

Luck, J., Little, C. & Hoff, W. (2000). Registration of range data using a hybrid simulated annealing and iterative closest point algorithm. In *Proceedings of the 2000 IEEE International Conference on Robotics & Automation*. San Francisco, USA: IEEE. 3739–3744.

Maaskant, R., Alavie, T., Measures, R. M., Tadros, D., Rizkalla, S. H. & Guha-Thakurta, A. (1997). Fiber-optic Bragg grating sensors for bridge monitoring. *Cement & Concrete Composites*.

Maes, F. Collignon, A., Vandermeulen, D., Marchal, G. & Suetens P. (1997). Multimodality image registration by maximization of mutual information. *IEEE Transactions on medical imaging* 16:2, 187–198.

Malcolm, D. T. K., Nielsen, P. M. F., Hunter, P. J. & Charette, P. G. (2002). Strain measurement in biaxially loaded inhomogeneous, anisotropic elastic membranes. *Biomechanics and Modeling in Mechanobiology* 1:3, 197–210.

Mantere, T. & Alander, J. T. (2002). Developing and testing a structural light vision software by co-evolutionary genetic algorithm. In *Proceedings of the Second ASERC Workshop on Quantitative and Soft Computing Based Software Engineering*. Alberta, Canada: University of Alberta. 31–37.

Mantere, T. & Koljonen J. (2008). Solving and analyzing Sudokus with cultural algorithms. In *Proceedings of the 2008 IEEE Congress on Evolutionary Computation*. Hong Kong, China: IEEE. 4054–4061.

Mantere, T. (2005). A min-max genetic algorithm for min-max problems. In Y. Wang, Y.-M. Cheung, H. Liu (eds.). *Advances in Computational Intelligence and Security – The Workshop of 2005 International Conference on Computational Intelligence and Security*. Xi'an, China: Xidian University Press. 52–57.

Mantere, T. (2007). Object and pose recognition with cellular genetic algorithms. In D. P. Casasent, E. L. Hall & J. Röning (eds.). *Intelligent Robots and Computer Vision XXV: Algorithms, Techniques, and Active Vision*. Boston, USA: SPIE. 67640N-1–10.

Mäntyjärvi, K., Tulonen, J., Saarnivuo, T. & Karjalainen, J. A. (2008). Properties of grid patterns produced by laser for strain analysis. In N. Asnafi (ed.). *Best in Class Stamping. Proceedings of the International Conference of International Deep Drawing Research Group*. Olofström, Sweden: IDDRG. 145–152.

Marcellier, H., Vescolo, D., Varchon, P., Vacher, P. & Humbert, P. (2001). Optical analysis of displacement and strain fields on human skin. *Skin Research and Technology* 7:4, 246–253.

Martin, J. & Crowley, J. L. (1995). Experimental comparison of correlation techniques. In U. Remobold et al. (eds.). *Proceedings of the International Conference on Intelligent Autonomous Systems*. Karlsruhe, Germany: Ios Pr Inc.

Massa, A., Franceschini, D. & Franceschini, G. (2005). Parallel GA-based approach in microwave imaging applications. *IEEE Transactions on Antennas and Propagation* 53:10, 3118–3127.

Mathias, K. E. & Whitley, L. D. (1994). Transforming the search space with Gray coding. In *Proceedings of the First IEEE Conference on Evolutionary Computation*. Orlando, USA: IEEE. 513–518.

Maybank, S. J. & Faugeras, O. D. (1992). A theory of self-calibration of a moving camera. *International Journal of Computer Vision* 8:2, 123–151.

Memon, Q. & Khan, S. (2001). Camera calibration and three-dimensional reconstruction of stereo-vision using neural networks. *International Journal of Systems Science* 32:9, 1155–1159.

Moore, A. J. & Tyrer, J. R. (1990). An electronic speckle pattern interferometer for complete in-plane displacement measurement. *Measurement Science and Technology* 1, 1024–1030.

Moscato, P. (1989). *On Evolution, Search, Optimization, GAs and Martial Arts: Toward Memetic Algorithms*. California Institute of Technology. Caltech Concurrent Computation Program Report 826.

Nicolella, D. P., Nicholls, A. E., Lankford, J. & Davy, D. T. (2001). Machine vision photogrammetry: a technique for measurement of microstructural strain in cortical bone. *Journal of Biomechanics* 34, 135–139.

Nordling, T. E. M., Koljonen, J., Alander, J. T. & Geladi P. (2004) Genetic algorithms as a tool for wavelength selection. In J. T. Alander, P. Ala-Siuru & H. Hyötyniemi (eds.). *Proceedings of the 11th Finnish Artificial Intelligence Conference*. Vantaa, Finland: Finnish Artificial Intelligence Society. 99–113.

Oduguwa, V., Tiwari, A. & Roy R. (2005). Evolutionary computing in manufacturing industry: an overview of recent applications. *Applied Soft Computing* 5, 281–299.

Orteu, J. J., Garric, V. & Devy, M. (1997). Camera calibration for 3D reconstruction: application to the measure of 3D deformation on sheet metal parts. In P. Refregier & R.-J. Ahlers (eds.). *New Image Processing Techniques and Applications: Algorithms, Methods, and Components II*. Munich, Germany: SPIE. 252–263.

Périé, J.-N., Calloch, S., Cluzel, C. & Hild, F. (2002). Analysis of multiaxial test on a C/C composite by using digital image correlation and a damage model. *Experimental Mechanics* 42:3, 318–328.

Peters, W. H. & Ranson, W. F. (1982). Digital image techniques in experimental stress analysis. *Optical Engineering* 21, 427.

Pilch, A., Mahajan, A. & Chu, T. (2004). Measurement of whole-field surface displacements and strain using a genetic algorithm based intelligent image corre-

lation method. *Journal of Dynamic Systems, Measurement, and Control* 126:3, 479–488.

Press, W. H., Teukolsky, S. A., Vetterling, W. T. & Flannery, B. P. (1992). *Numerical Recipes in C – The Art of Scientific Computing*. Second Edition. Cambridge, USA: Cambridge University Press.

Rechenberg, I. (1971). *Evolutionsstrategie - Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Technical University of Berlin. Department of Process Engineering. PhD thesis.

Reeves, C. R. (1993). Using genetic algorithms with small populations. In S. Forest (ed.). *Proceedings of the Fifth International Conference on Genetic Algorithms*. San Mateo, USA: Morgan Kaufmann. 92–99.

Reynolds, R. G. & Sverdlik, W. (1994). Problem solving using cultural algorithms. In *Proceedings of the First IEEE Conference on Evolutionary Computation*. Orlando, USA: IEEE. 645–650.

Robert, L. (1994). Camera calibration without feature extraction. In *Proceedings of the 12th IAPR International Conference on Pattern Recognition*, Jerusalem, Israel: IEEE. 704–706.

Robinson, D. W. (1983). Automatic fringe analysis with a computer image-processing system. *Applied Optics* 22:14, 2169–2176.

Robinson, D. & Milanfar, P. (2004). Fundamental performance limits in image registration. *IEEE Transactions on Image Processing* 13:9, 1185–1199.

Rudolph, G. (1996). Convergence of evolutionary algorithms in general search spaces. In *Proceedings of IEEE International Conference on Evolutionary Computation*. Nagoya, Japan: IEEE. 50–54.

Rueckert, D., Sonoda, L. I., Hayes, C., Hill, D. L. G., Leach, M. O. & Hawkes, D. J. (1999). Nonrigid registration using free-form deformations: Application to breast MR images. *IEEE Transactions on Medical Imaging* 18:8, 712–721.

Ruoppa, R. [Personal communication 17th June 2009].

Saleem, Q., Wildman, R. D., Huntley, J. M. & Whitworth, M. B. (2003). A novel application of speckle interferometry for the measurement of strain distributions in semi-sweet biscuits. *Measurement Science and Technology* 14, 2027–2033.

Schowengerdt, R. A. (2007). *Remote Sensing. Models and Methods for Image Processing.* 3rd edition. USA: Elsevier.

Schreier, H. W., Braasch, J. R. & Sutton, M. A. (2000). Systematic errors in digital image correlation caused by intensity interpolation. *Optical Engineering* 39:11, 2915–2921.

Sen, M., Hemaraj, Y., Bhattacharyya, S. S. & Shekhar, R. (2006). Reconfigurable image registration on FPGA platforms. In *Proceedings of Biomedical Circuits and Systems*. London, UK: IEEE. 154–157.

Sen, M., Hemaraj, Y., Plishker, W., Shekhar, R. & Bhattacharyya, S. S. (2008). Model-based mapping of reconfigurable image registration on FPGA platforms. *Journal of Real-Time Image Processing* 3:3, 149–162.

Shanno, D. F. (1970). Conditioning of quasi-Newton methods for function minimization. *Mathematics of Computing* 24, 647–656.

Shaopeng, M. & Guanchang, J. (2003). Digital speckle correlation method improved by genetic algorithm. *Acta Mechanica Solida Sinica* 16:4, 366–373.

Shen, H., Stewart, C. V., Roysam, B., Lin, G. & Tanenbaum, H. L. (2003). Frame-rate spatial referencing based on invariant indexing and alignment with application to online retinal image registration. *IEEE Transactions on Pattern Analysis and Machine Vision* 25:3, 379–384.

Sirohi, J. & Chopra, I. (2000). Fundamental understanding of piezo-electric strain sensors. *Journal of Intelligent Material Systems and Structures* 11:4, 246–257.

Smith, S.F. (1980). *A Learning System Based on Genetic Adaptive Algorithms*. University of Pittsburgh. Computer Science Department. PhD Thesis.

Sonka, M, Hlavac, R. & Boyle, R. (2008). *Image Processing, Analysis, and Machine Vision*. 3rd edition. USA: Thomson Learning.

Spajer, M., Rastogi, P. K. & Monneret, J. (1981). In-plane displacement and strain measurement by speckle interferometry and moiré derivation. *Applied Optics* 20:19, 3392–3402.

Starink, J. P. P. & Backer, E. (1995). Finding point correspondence using simulated annealing. *Pattern Recognition* 28:2, 231–240.

Storn, R. & Price, K. (1997). Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization* 11, 341–359.

Sturm, P. F. & Maybank, S. J. (1999). On plane-based camera calibration: A general algorithm, singularities, applications. In *Proceeding of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Fort Collins, USA: IEEE. 432–437.

Su, C. & Anand, L. (2003). A new digital image correlation algorithm for whole-field displacement measurement. In *Innovation in Manufacturing Systems and Technology* [online]. Singapore-MIT Alliance. Available at: http://dspace.mit.edu/handle/1721.1/3749.

Suetens, P. (2002). *Fundamentals of Medical Imaging.* USA: Cambridge University Press.

Sullivan, G. D., Baker, K. D., Worrall, A. D., Attwood, C. I. & Remagnino, P. M. (1997). Model-based vehicle detection and classification using orthographic approximations. *Image and Vision Computing* 15, 649–654.

Sutton, M. A. & McFadden, C. (2000). Development of a methodology for non-contacting strain measurements in fluid environments using computer vision. *Optics and Lasers in Engineering* 32, 367–377.

Synnergren, P. Goldrein, H. T. & Proud, W. G. (1999). Application of digital speckle photography to flash x-ray studies of internal deformation fields in impact experiments. *Applied Optics* 38:19, 4030–4036.

Syswerda, G. (1989). Uniform crossover in genetic algorithms. In *Proceedings of the Third International Conference on Genetic Algorithms*. Fairfax, USA: Morgan Kauffman. 2–9.

Tan, Z., Melin, L. & Magnusson, C. (1992). Application of an image processing technique in strain measurement in sheet metal forming. *Journal of Material Processing Technology* 33, 299–310.

Tbakhi, A. & Amr, S. (2007). Ibn Al-Haytham: father of modern optics. *Ann Saudi Med* 27:6, 464–467.

The Mathworks, Inc. 2006. Help Manual of Matlab Version 7.3.0.267.

Thévenaz, P., Ruttimann, U. E. & Unser, M. (1998). A pyramid approach to subpixel registration based on intensity. *IEEE Transactions on Image Processing* 7:1, 27–41.

Tian, Q. & Huhns, M. N. (1986). Algorithms for subpixel registration. *Computer Vision, Graphics, and Image Processing* 35, 220–233.

Tomassini, M. (1999). Parallel and distributed evolutionary algorithms: A review. In K. Miettinen et al. (eds.). *Evolutionary Algorithms in Engineering and Computer Science*. New York, USA: Wiley. 113–133.

Tsai, R. Y. (1987). A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE Journal of Robotics and Automation* 3:4, 323–344.

Tulonen, J. [Personal communication 10th Dec. 2008].

Václavík, J., Minster, J. & Houha, R. (2002). Laser extensometer using stuck foil-embossed diffraction grating. In *Proceedings of the 40th International Conference of Experimental Stress Analysis*. Prague, Czech Republic: Czech Technical University in Prague. 259–260.

Vendroux, G. & Knauss, W. G. (1998). Submicron deformation field measurements: Part 2: Improved digital image correlation. *Experimental Mechanics* 38:2, 86–92.

ViALUX GmbH [Homepage]. [Cited 13th Mar. 2009]. Available at: http://www.vialux.de.

Vihtonen, L., Puzik, A. & Katajarinne, T. (2008). Comparing two robot assisted incremental forming methods: incremental forming by pressing and incremental hammering. *International Journal of Material Forming* 1, 1207–1210.

Vihtonen, L., Tulonen, J. & Tuomi J. (2008). Incremental sheet forming with an industrial robot as a method for sheet forming prototyping and small series manufacturing. In N. Asnafi (ed.). *Best in Class Stamping. Proceedings of the International Conference of International Deep Drawing Research Group* [CD-ROM]. Olofström, Sweden: IDDRG. 521–532.

Villa-Uriol, M.-C., Chaudhary, G., Kuester, F., Hutchinson, T. & Bagherzadeh, N. (2004). Extracting 3D from 2D: selection basis for camera calibration. In *Proceedings of Computer Graphics and Imaging*. Kauai, USA: Acta Press. 96–115.

Viola, P. & Wells III, W. M. (1997). Alignment by maximization of mutual information. *International Journal of Computer Vision* 24:2, 137–154.

Vrooman, H. A. & Maas, A. A. M. (1991). Image processing algorithms for the analysis of phase-shifted speckle interference patterns. *Applied Optics* 30:13, 1636–1641.

Wang, W.-C., Ho, C.-T., Lian, Y.-R. & Chuang, W.-C. (2006). Transducing mechanical force by use of a diffraction grating sensor. *Applied Optics* 45:9, 1893–1897.

Wei, G.-Q. & Ma, S. D. (1994). Implicit and explicit camera calibration: theory and experiments. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16:5, 469–480.

Weng, J., Cohen, P. & Herniou, M. (1992). Camera calibration with distortion models and accuracy evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14:10, 965–980.

Whitley, D. (2001). An overview of evolutionary algorithms: practical issues and common pitfalls. *Information and Software Technology* 43, 817–831.

Whitley, D., Rana, S. & Heckendorn, R. B. (1999). The island model genetic algorithm: On separability, population size and convergence. *Journal of Computing and Information Technology* 7:1, 33–47.

Yang, A., Meng, L., Luo J. & Lin, C. (2007). A rapid registration framework for medical images. In *Proceedings of Fourth International Conference on Image and Graphics*. Chengdu, China: IEEE. 731–736.

Yi, H. H., Kim, S. H. & Kwak, Y. K. (2000). A nanometric displacement measurement method using the detection of fringe peak movement. *Measurement Science and Technology* 11, 1352–1358.

Yoneyama, S. & Morimoto, Y. (2003). Accurate displacement measurement by correlation of colored random patterns. *JSME International Journal, Series A* 46:2, 178–184.

You, J. & Bhattacharya, P. (2000). A wavelet-based coarse-to-fine image matching scheme in a parallel virtual machine environment. *IEEE Transactions on Image Processing* 9:9, 1547–1559.

Zagar, B. G. & Kargel, C. (1999). A laser-based strain sensor with optical preprocessing. *IEEE Transactions on Instrumentation and Measurement* 48:1, 97–101.

Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22:11, 1330–1334.

Zhou, P. & Goodson, K. (2000). Thermomechanical diagnostics of BGA packages using digital image/speckle correlation. In G. B. Kromann, J. R. Culham & K. Ramakrishna (eds.). *Proceedings of the Seventh Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems*. Las Vegas, USA: IEEE. 240–245.

Ziemann, H. (1986). Thoughts on a standard algorithm for camera calibration. *International Archives of Photogrammetry and Remote Sensing* 26:5, 85–93.

Zitova, B. & Flusser, J. (2003). Image registration methods: a survey. *Image and Vision Computing* 21, 977–1000.

# REPRINTS OF THE PUBLICATIONS

## List of errata

Significant known errors of publications are listed below. Equation numbers in *italics* and in **bold-face** refer to the equations in the corresponding articles and this list of errata, respectively.

Article I:

Unfortunately, equations (*5*, *6*, *9*, *10*, and *12*), in article I, have a systematic erratum. In eqs. (*5* and *6*) the strain is defined wrong. The intention was to use engineering strain, i.e., $\varepsilon = \Delta l/l^{(0)}$, but due to a mistake the equations define the strain erroneously as $\varepsilon^* = l^{(k)}/l^{(0)}$. Evidently, $l^{(k)} = l^{(0)} + \Delta l$. Hence, $\varepsilon^* = 1 + \Delta l/l^{(0)} = 1 + \varepsilon$ and, after rearrangement of terms, $\varepsilon = \varepsilon^* - 1$. Consequently, eqs. (*5* and *6*) should be, respectively:

**(70)** $$\varepsilon_y = \frac{y_{i,j+1}^{(k)} - y_{i,j}^{(k)}}{y_{i,j+1}^{(0)} - y_{i,j}^{(0)}} - 1 \text{ and}$$

**(71)** $$\varepsilon_x = \frac{x_{i,j+1}^{(k)} - x_{i,j}^{(k)}}{x_{i,j+1}^{(0)} - x_{i,j}^{(0)}} - 1 .$$

Eqs. (*9* and *10*) can be corrected similarly. They should be, respectively:

**(72)** $$\varepsilon_{y,i,j} = \frac{\left(y_{i,j+1}^{(k)} - y_{i,j}^{(k)}\right) + \left(y_{i+1,j+1}^{(k)} - y_{i+1,j}^{(k)}\right)}{2\left(y_{i,j+1}^{(0)} - y_{i,j}^{(0)}\right)} - 1 \text{ and}$$

**(73)** $$\varepsilon_{x,i,j} = \frac{\left(x_{i+1,j}^{(k)} - x_{i,j}^{(k)}\right) + \left(x_{i+1,j+1}^{(k)} - x_{i,j+1}^{(k)}\right)}{2\left(x_{i+1,j}^{(0)} - x_{i,j}^{(0)}\right)} - 1 .$$

In eq. (*12*), $\Delta x$ should be replaced by $\Delta x - \Delta x_0$ that corresponds to $\Delta l$. Hence, eq. (*12*) should read:

**(74)** $$\varepsilon_x = \left(\Delta x \pm 2\delta_x - \Delta x^{(0)}\right)/\Delta x^{(0)} = \left(\Delta x - \Delta x^{(0)}\right)/\Delta x^{(0)} \pm 2\delta_x/\Delta x^{(0)} = \hat{\varepsilon}_x \pm 2\delta_x/\Delta x^{(0)},$$

where $\hat{\varepsilon}_x$ denotes the mean transverse strain.

Fortunately, the correct definition of engineering strain was used in the computer program. Consequently, all experimental results as regards the erroneous equations should be correct in the article.

In addition to the errors in equation, this article includes a somewhat inaccurate claim that "[…] no absolute camera calibration is required. However, the *differential* imaging distortions induce errors to the results, but these are typically small if the distances between the grid nodes are small". As late as in article V, it was studied formally how the lens distortions actually affect the strain measurements. In short, the result was that radial distortion changes the apparent original distance $l_0$, too. Hence, the claim that only the difference of the distortions of the nodes, i.e., the 'differential distortion', affected the strain measurements was too hesitated. Moreover, the magnitude of distortions obviously depends on many factors, but in many practical cases, it seems to be almost negligible as for strain measurements.

Article II:

Engineering strain is again defined wrong. Eq. (*2*) should read as eq. (**68**).

In section 3.1.3, two confidence measures, maximum correlation (MC) and discrimination gap (DG), have incorrect subscripts in their definitions (subscript *i* is used consistently instead of numerals 1 and 2). The paragraph in question should read:

"Using the correlation maxima two useful confidence measures are proposed:
- Maximum correlation (MC): $r_{\max(1)}$ and
- the difference of the first two peaks: $r_{\max(1)} - r_{\max(2)}$. Let us call this *discrimination gap* (DG). This has already previously been introduced by the authors […]."

Article III:

Engineering strain is again defined wrong. Eq. (*2*) should read as eq. (**68**).

Article V:

Digital image correlation is consistently referred using an erroneous abbreviation DIG instead of DIC.

In the last sentence of the fifth paragraph of introduction, word 'usual' should be 'useful', i.e., the implicit validation method referred is considered useful but not frequently used according to literature.

EUROCON 2007 The International Conference on "Computer as a Tool" Warsaw, September 9-12

# An Implicit Validation Approach for Digital Image Correlation Based Strain Measurements

Janne Koljonen, Olli Kanniainen, and Jarmo T. Alander
University of Vaasa/Dept. of Electrical Eng. and Automation, Vaasa, Finland, e-mail: *Firstname.Lastname@uwasa.fi*

***Abstract*—A digital image correlation algorithm for strain measurements with some improvements is introduced and tested. A major problem related to strain measurements with spatial resolution is the lack of simple validation of the results as the overall strain given by the test machine is not applicable as a local strain reference estimate. An implicit error estimation approach based on statistical analysis is introduced and applied to an image test set from a uniaxial tensile test of a planar aluminum test piece. The spatial strain sampling frequency and the measurement accuracy and errors are studied, when varying a few parameters of the algorithm. An intra-validation procedure is in turn used to validate the error estimates.**

***Keywords*—error estimation, deformation, material engineering, measurements based on image.**

## I. Introduction

Measurement of strains is an essential part when determining the mechanical properties of materials [1]. Typical material test machines are capable of measuring the overall strain of a test piece, but if a spatial distribution of the strains is required, other methods, typically optical ones, are used.

A common and simple optical strain measurement is digital image correlation [2]. A major problem related to strain measurements with spatial resolution is the lack of simple validation of the results as the overall strain given by the test machine is not applicable as a local strain reference estimate.

Typically, strains are measured with several methods, which are subsequently compared for validation purposes [3]. Another approach is to test algorithms by simulation and artificial or artificially deformed images [4]. Digital image correlation strain algorithms are sometimes tested also by measuring the localization accuracy instead of the strain accuracy as they are interconnected [4–6]. However, usually the comparisons are done using the overall strain, not the spatial distributions, and only a few points in the strain space are compared. Moreover, there does not seem to be any established measures of error in the literature and therefore the comparison of different studies is difficult.

Commonly used strain measurement methods are resistive and piezoelectric [7] strain gauges, acoustic emission [8], and optical methods. Optical strain measurements include digital image correlation [9, 10] and 3D image correlation [11], grating [12], moiré and holographic speckle interferometry [13–15], laser speckle [16–19], and electronic speckle pattern interferometry [20]. Each method has some benefits and drawbacks, and their resolution and strain range of operation differ significantly.

In this paper, a digital image correlation algorithm to measure principal strains is introduced in section II. The algorithm includes also some improvements. The accuracy estimation problem is tackled by an implicit validation approach based on statistical analysis in section III. The accuracy estimates are subsequently validated using another implicit method. In addition, the theoretical relation between spatial strain resolution and measurement accuracy is studied. The experimental setup and test runs are dealt in section IV. Section V deals with test results and discussion.

## II. Strain by Digital Image Correlation

Strain measurement by digital image correlation is based on tracking displacements of homologous landmarks (fiducial/reference points) using a static imaging configuration. Typically the fiducial points are nodes of a regular grid. By comparing the displacements of the grid nodes to each other the strains, relative to a reference image, can be computed.

The strain measurements are based on relative distances and thus no absolute camera calibration is required. However, the *differential* imaging distortions induce errors to the results, but these are typically small if the distances between the grid nodes are small. The imaging error can be compensated by a distortion model [21], but this is not done in this study, because it is technically challenging due to the non-fixed optics of the camera in use.

In the following, the algorithm used in this study is introduced. In this study, the emphasis is on the new unsupervised error estimates. Yet an improvement to enhance speed and reliability is obtained using prediction on the reference tracking. In addition to the algorithm, resolution and errors of digital image correlation strain measurements are studied.

### A. Template Sampling

Suppose that a test piece with a random pattern is imaged at approximately regular intervals of stretches. These images are denoted by $im_0$, $im_1$, ..., $im_K$, totally $K$ images of the sample with deformation and one before deformation ($im_0$), a non-deformed reference. The deformations of the others are calculated with respect to this reference. The longitudinal and transversal axes corresponding to the principal strain directions are denoted by y and x, respectively.

Finnish Funding Agency for Technology and Innovation (TEKES) and the industrial partners of the research project *Process Development for Incremental Sheet Forming* have supported this research.

1-4244-0813-X/07/$20.00 2007 IEEE. 

From $im_0$ a regular rectangular grid of reference point centers are selected. Let $\mathbf{c}_{0,0} = (x_{0,0}, y_{0,0})$ and $\mathbf{c}_{I,J} = (x_{I,J}, y_{I,J})$ be the locations of the first and last nodes of the grid, respectively. Now $I$ and $J$ are the number *strain* measurement samples in transversal and longitudinal directions, respectively. In other words, the numbers of reference points are $I + 1$ and $J + 1$, respectively. The distance between nodes in x and y directions are, respectively,

$$\Delta x = \left(x_{I,j} - x_{0,j}\right)/ I \tag{1}$$

and

$$\Delta y = \left(y_{i,J} - y_{i,0}\right)/ J\,. \tag{2}$$

The coordinates of node $(i, j)$ are subsequently

$$\mathbf{c}_{i,j} = \left(x_{0,0} + i\Delta x, y_{0,0} + j\Delta y\right). \tag{3}$$

Image templates called Zones-Of-Interest (ZOI) are sampled around the reference points. Let the $x$ and $y$ dimensions of a ZOI be $z_x$ and $z_y$, respectively. Then a $z_x \times z_y$ sub-image of $im_0$ is sampled for each grid node so that the reference point is located at the center of the ZOI (Fig. 1).

The templates are used to localize the reference points from the images of the deformed test piece. After localization, it is possible and sometimes advisable to update the ZOIs using the neighborhood of the localized points on the deformation image. This update procedure is beneficial when the deformation is so great that the ZOIs of the non-deformed image significantly differ from corresponding areas at the deformed images. Alternatively the ZOIs should be computationally deformed (scaled) according to the estimated deformation, but this method counters the difficulties related to pixel value interpolation [24].

The update procedure refreshes the template pixels but not the coordinates, i.e. the original reference points of the non-deformed image are still looked for after an update. Hence the strain fields are still computed with respect to $im_0$. The interval between updates is denoted by $\Delta k$.

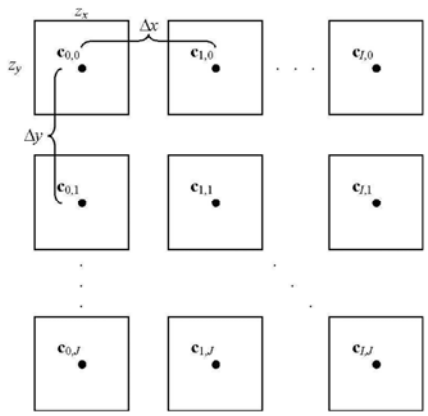



Fig. 1. ZOI grid and dimensions.

### B. *Reference Point Tracking*

The reference points selected from $im_0$ are located and tracked on images $im_1$, $im_2$, ..., $im_k$ using ZOIs sampled from $im_0$ or re-sampled from a later, deformed image. The localization of a ZOI is carried out by translating it over a Region-Of-Interest (ROI) and comparing the overlapping pixels using a similarity metric. In this study, correlation was used as the similarity measure. The position (translation vector) of the best match is regarded as the location of the reference point (Fig. 2).

To obtain sub-pixel resolution in localization the sampled correlation function interpolation method was applied [25]: On each pixel-grid point, the correlation coefficient between the ZOI and ROI was computed. A paraboloid was fit around the pixel with the maximum correlation using a 5×5 neighborhood. The maximum of the estimated continuous paraboloid function gives the sub-pixel location of the ZOI.

The resolution of the sub-pixel localization depends mainly on the template size as well as localization and interpolation method. Ref. [26] deals with theoretical limits of sub-pixel localization of edges. Ref. [24] in turn shows that another method, pixel interpolation, induces a systematic error. In this study, Pearson correlation was used as the similarity metrics, even though the Minimum Square Error (MSE) is also commonly used in strain analysis [27, 28]. Many other sub-pixel localization methods also exist, including maximum-likelihood template matching [29], augmented template matching [30], phase correlation [31], and de-aliasing [32].

ROI is a sub-image of the deformation image in concern. The origin of a ROI was determined by predicting the displacement of a reference point according to two previous images. Let $\left(\hat{x}_{i,j}^{(k)}, \hat{y}_{i,j}^{(k)}\right)$ be the estimated (measured) location of a grid point in $im_k$. By a linear optical flow model the predicted location of a grid node is

$$\left(\tilde{x}_{i,j}^{(k)}, \tilde{y}_{i,j}^{(k)}\right) = (a+1)\left(\hat{x}_{i,j}^{(k-1)}, \hat{y}_{i,j}^{(k-1)}\right) - a\left(\hat{x}_{i,j}^{(k-2)}, \hat{y}_{i,j}^{(k-2)}\right), \tag{4}$$

where $a \in [0, 1]$. Selecting $a = 0$ corresponds to the zero-order prediction i.e. using only the previous image, and $a = 1$ in turn to pure linear displacement. All combinations between these extremes are obtained selecting $0 < a < 1$.

The size of a ROI was determined to be a multiple of the ZOI dimensions. The ROI dimensions with respect to its origin are denoted by: $r_x$ in positive and negative transversal directions, $r_{y-}$ in the direction of negative longitudinal strain, and $r_{y+}$ in the direction of positive longitudinal strain. The origin of the ROI is not necessarily at the center. At least in this study, $r_{y+} > r_{y-}$ because the tensile test-

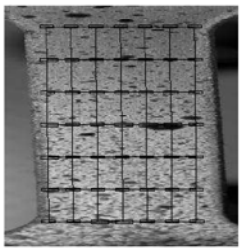


Fig. 2. Localized ZOI grid at a deformed image.

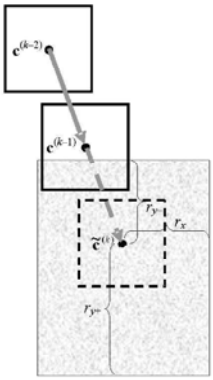


Fig. 3. Two previous localized ZOIs $\mathbf{c}^{(k-2)}$, $\mathbf{c}^{(k-1)}$, and the predicted ZOI location $\tilde{\mathbf{c}}^{(k)}$. The dimensions of ROI are shown around the predicted position.

ing machine caused the increment of stretch to increase after a certain limit (Fig. 6).

### C. *Principal Strains from Node Coordinates*

After obtaining the coordinates of the grid points the strains can be computed. In this study, only principal strains i.e. the relative changes of lengths in longitudinal (y) and transversal (x) directions are considered.

Two neighboring nodes ($x_{i,j}$, $y_{i,j}$) and ($x_{i,j}$, $y_{i,j+1}$) span a vertical line, whose length is $y_{i,j+1}-y_{i,j}$. The principal y strain $\varepsilon_y$ is obtained by comparing the deformed length to the original one:

$$\varepsilon_y = \left(y_{i,j+1}^{(k)} - y_{i,j}^{(k)}\right) / \left(y_{i,j+1}^{(0)} - y_{i,j}^{(0)}\right). \tag{5}$$

Similarly the $x$ strain is obtained by

$$\varepsilon_x = \left(x_{i+1,j}^{(k)} - x_{i,j}^{(k)}\right) / \left(x_{i+1,j}^{(0)} - x_{i,j}^{(0)}\right). \tag{6}$$

The measured strain is an average of the local strains on the area spun by the nodes that are used in the strain evaluation, if a somewhat continuous function of strains is presumed. In the case of (5), the y strain is an average of the strains on the line between two nodes. The center of the strain sample is at

$$(y_{i,j+1}+y_{i,j})/2. \tag{7}$$

Unfortunately, the x strain sample in (6) is measured at

$$(x_{i+1,j}+x_{i,j})/2, \tag{8}$$

which does not coincide with (7). Spatially coinciding evaluation of x and y strains is important if they are used to estimate the relative thickness change using the theorem of constant volume.

If four nodes $\mathbf{c}_{i,j}$, $\mathbf{c}_{i+1,j}$, $\mathbf{c}_{i,j+1}$, and $\mathbf{c}_{i+1,j+1}$ are used, both y and x strains are evaluated at the same spatial sample by

$$\varepsilon_{y,i,j} = \frac{\left(y_{i,j+1}^{(k)} - y_{i,j}^{(k)}\right) + \left(y_{i+1,j+1}^{(k)} - y_{i+1,j}^{(k)}\right)}{2\left(y_{i,j+1}^{(1)} - y_{i,j}^{(1)}\right)}, \tag{9}$$

and

$$\varepsilon_{x,i,j} = \frac{\left(x_{i+1,j}^{(k)} - x_{i,j}^{(k)}\right) + \left(x_{i+1,j+1}^{(k)} - x_{i,j+1}^{(k)}\right)}{2\left(x_{i+1,j}^{(1)} - x_{i,j}^{(1)}\right)}, \tag{10}$$

respectively. In practice, (9) and (10) compute the mean of two line samples, and these equations are used in this study when determining strains from displacements.

### D. *Resolutions and Errors of Strain Measurements*

Equations (9) and (10) imply that digital image correlation computes strains as an average from a finite area. The spatial resolution of the strain measurements is $\Delta x$ and the corresponding spatial sampling frequency (strain samples/pixel) is

$$f_{s_{\varepsilon_x}} = 1/\Delta x. \tag{11}$$

There is a relationship between the strain sampling frequency and the accuracy (error) of the strain measurements. Suppose that the maximum localization error of the grid nodes (one-dimensionally) is $\delta_x$ pixels. Interval arithmetic [33] gives that the distance between two adjacent nodes is $\Delta x \pm 2\delta_x$. The strain is thus

$$\varepsilon_x = \left(\Delta x \pm 2\delta_x\right) / \Delta x^{(0)} = \varepsilon_x \pm 2\delta_x / \Delta x^{(0)}, \tag{12}$$

where $\Delta x^{(0)}$ is the spatial resolution of the non-deformed grid. As a conclusion, the maximum strain measurement error is

$$\delta\varepsilon_x = 2\delta_x / \Delta x^{(0)} = 2\delta_x f_{s_{\varepsilon_x}}. \tag{13}$$

Equation (13) suggests that strain error is a linear function of localization accuracy and spatial strain sampling frequency. Inversely, (13) suggests that the localization accuracy can be estimated if the strain measurement error is known or estimated. There are experimental methods to determine the localization accuracy [6]. However in this study, the strain error is estimated directly but implicitly. This approach is introduced in the next section.

## III. Error Estimates and Intra-validation

To validate the performance of a digital image correlation algorithm explicitly an accurate reference method is required. However, many strain measurement methods have poor spatial resolution (e.g. strain gauge), are pointwise and thus slow (diffracting grating [12], digital laser speckle [19]), or are not independent from digital image correlation (optical methods using spatial patterns), and thus are not necessarily practical or advisable to use in validation.

Next an implicit approach to validate strain measurements is proposed. Generally, it is applicable with any method with a spatial resolution, although only digital image correlation is treated in this study.

### A. *Implicit Accuracy Analysis*

The implicit error estimation is based on statistics and the assumption that the principal strains in a transversal intersection are uniform. The assumption should approximately hold if the test piece is homogeneous, and more positively if the stretch is kept below the proportional limit, whence also the longitudinal strain distribution should be uniform.

For short, the strain error is estimated by the standard deviation of strains measured in a transversal row of the grid, i.e. the row-wise error estimate for y strain is

$$\hat{\delta}\varepsilon_{y,\cdot,j} = \sqrt{\sum_{i=0}^{I-1}(\varepsilon_{y,i,j} - \sum_{i=0}^{I-1}\varepsilon_{y,i,j}/I)^2/(I-1)}\,. \quad (14)$$

If the strain error is normally distributed, eq. (14) and its multiples give confidence intervals for the row-wise strain errors. The corresponding column-wise error (accuracy) estimate is

$$\hat{\delta}\varepsilon_{y,i,\cdot} = \sqrt{\sum_{j=0}^{J-1}(\varepsilon_{y,i,j} - \sum_{j=0}^{J-1}\varepsilon_{y,i,j}/J)^2/(J-1)}\,. \quad (15)$$

There are totally $I$ column-wise and $J$ row-wise error estimates. They all could be used, but it is more practical to select one of them, particularly when comparing e.g. parameter setups and in optimization. A cautious (worst-case) approach is to use maximum error: e.g. the maximum row-wise y strain error estimate is

$$\hat{\delta}_{\max}\varepsilon_{y,\cdot,j} = \max_j(\hat{\delta}\varepsilon_{y,\cdot,j})\,. \quad (16)$$

A more robust one is to use the median value

$$\hat{\delta}_{\mathrm{med}}\varepsilon_{y,\cdot,j} = \mathrm{median}_j(\hat{\delta}\varepsilon_{y,\cdot,j})\,. \quad (17)$$

### *B. Intra-validation*

The accuracy of results should be estimated somehow, even without explicit validation. Again an implicit, statistical approach is followed.

The intra-validation is carried out by translating the images by a few (now five) pixels. A small translation should not affect strain measurements significantly if the assumption of uniform strain field holds. Moreover, the error estimates should remain unaltered, unless the estimates are unreliable.

After translation the strain error estimates (e.g. (14)) are re-computed. After a few (now $n = 5$) different translations, standard deviations of the error estimates are computed. Again, standard deviations give confidence intervals for the error estimates.

The translation method is applicable to the error estimation itself, too. Nevertheless, it was spared for the intra-validation of the estimates. Yet another way to determine strain errors is to estimate the localization error and to use (13). However, the estimation of the localization error cannot be done directly from the images of the deformed sample.

## IV. Experiments

In order to test the digital image correlation algorithm and the implicit approach for error estimation an image series from a uniaxial tensile test was taken. An aluminum 'dog bone' tensile test sample was spray-painted to obtain a random pattern of black spots against the aluminum background. The dimensions of the narrow part of the test piece are approximately 47 mm × 8 mm × 2 mm.

The test sample was imaged using Nikon D200 digital camera with 2592 × 3872 pixels (3×8 bit RGB) and Sigma Macro objective. All imaging and camera setups were fixed during the tensile test pulling. Movements of the camera were minimized operating it by remote control. The experimental setup is shown in Fig. 4.

Fig. 4. Imaging setup.

The test sample was imaged before any deformation and always after a stretch of approximately 0.2 mm. After 59 images and a stretch of 9.8 mm (21 %), a rupture occurred.

Because the reliable use of the implicit accuracy analysis assumes locally uniform strain of material, only the 27 first images of the data set were used in the computations. The last image has thus a strain of 11 %. The narrow part of the test piece i.e. the area of interest was about 330 × 1900 pixels. No pre-processing was done to the images, except transforming the images to gray-scale.

Already a visual inspection of the images reveals that the upper clamp of the tensile testing machine moves downward in the first images (Fig. 5). The notice is also confirmed in the results (Fig. 6).

The movement is due to the mechanism of the clamp. It is designed to tighten the jaws when the pulling begins. Unfortunately, this movement induces a bias of about 3 mm to the stretch measure of the tensile test machine, which is usually compensated by a minor pre-stretch. On the contrary, the bow down does not (in theory) affect optical strain measurements.

When the image series was flicked through, it was noticed that the material had stretched longitudinally non-uniformly between the images. On the contrary, it seems that the deformation occurs in local twitches, particularly when proportional limit of the material is exceeded. The non-uniformity of deformation may be present at the results.

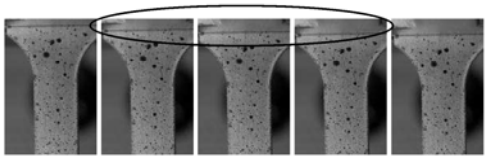


Fig. 5. Magnifications of the 1st, 5th, 9th, 13th, and 17th image. The movement of the upper clamp caused by tightening is visible in the 5th, 9th and 13th image (circled), after which it practically stopped.

The digital image correlation strain algorithm was tested using the data set and varying some parameters. For computational reasons only some one-dimensional cross-sections of the parameter space were evaluated. The test runs and the algorithm parameters tested are summarized in TABLE I.

The other parameters were fixed: in location prediction $a = 0$, ROI size parameters were $r_x = z_x$, $r_{y-} = z_y$, and $r_{y+} = 6z_y$. The ROI dimension $r_{y+}$ was so wide, because the prediction part of (4) becomes zero for the first deformed image. Therefore, a larger search area is needed, particularly if the update step is large. The sudden increase of strain increments (Fig. 6) is another reason. Adaptive $r_{y+}$ could speed up the algorithm.

## V. Results and Discussion

The results of the four test runs (TABLE I) are now given and discussed. Due to high amount of data only a part of the results can be shown and discussed.

The implicit error estimates are used to study the accuracy of the strain measurements with respect to different parameters of the algorithm. These results are compared to theoretical predictions.

The 95% confidence intervals obtained by intra-validation are shown by dotted lines, unless otherwise mentioned. Intra-validation is used to examine, how well the error estimates hold. The confidence intervals are now

$$\overline{X} \pm t_{0.025} S / \sqrt{n} = \overline{X} \pm 1.15 S \,, \tag{18}$$

where $\overline{X}$ is a mean estimate, $t_{0.025} = 2.571$ is obtained from the $T$ distribution, $S$ is standard deviation estimate, and $n = 5$ is the sample size [34].

First, the overall relative strain measured by the digital image correlation algorithm is plotted against the image rank index $k$ (Fig. 6). The results confirm the finding that the clamps bow down at the beginning of the pull. The deviation of the estimate (intra-validated) is found small.

In Fig. 7 the estimated y strain errors (median and maximum) are plotted against the overall strain and image index $k$ using different template sizes. The errors increase almost linearly with respect to $k$, i.e. at each strain evaluation (now $\Delta k = 1$) an almost fixed amount of error is induced and accumulated, although according to Fig. 6 the strain increment is lower at the beginning. An exception occurs, when $z_x = z_y = 15$, whence at some $k$ more errors occur. The accumulation of errors and the result of Fig. 6 also explain why the error with respect to overall strain increases drastically at low strains.

The error accumulation mostly originates from the ZOI localization. Every localization includes some small error (usually less than a pixel). If no update is done this error occurs only at that measurement. On the contrary, if the

TABLE I.
Test runs and objective parameters

| Run | Template size ($z_x = z_y$) | Update interval, $\Delta k$ | Number of x nodes, $I+1$ | Number of y nodes, $J+1$ |
|---|---|---|---|---|
| 1 | {11, 13, …, 33} | {1} | {7} | {7} |
| 2 | {15, 21} | {1, 2, …, 13, 25} | {7} | {7} |
| 3 | {25} | {5} | {7} | {3, 5, …, 31} |
| 4 | {25} | {5} | {3, 4, …, 17} | {7} |

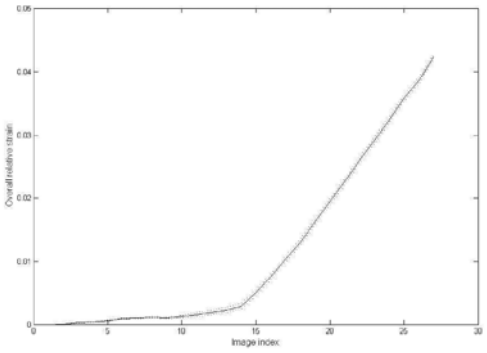



Fig. 6. Estimated overall strains vs. the rank index of the image. The results expose that at the beginning of the pull the test piece did not stretch much, because the clamp of the tensile testing machine has a mechanism that bows down and tightens up automatically when pulling.

template is updated, the localization coordinates are rounded to the nearest integer pixels, around which a template is then sampled from. The rounding error is recorded, and the accumulative rounding errors are always compensated in localizations. Because localization errors are mutually independent they accumulate in updates. Nevertheless, the localization errors are transferred at least to the rounding error account and sometimes to the template re-sampling coordinates.

If the local strain error estimate is divided by the overall strain, relative error estimate is obtained. Relative error (Fig. 8) exposes the pleasant fact that strain measurement error increase slower than the overall strain. Henceforth the relative error approaches zero in great strains.

The effect of the template size on strain errors was already visible in Fig. 7. The effect was studied more closely in Fig. 9, inclusive confidence intervals. It was found that after 17 × 17 pixels, the marginal benefit of

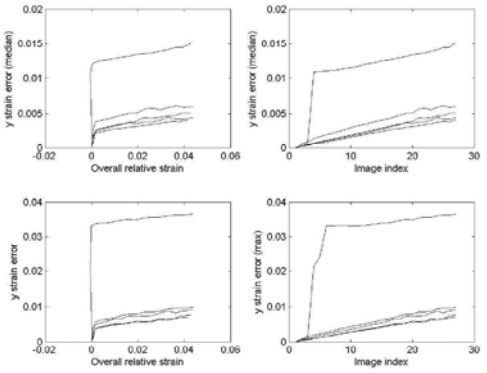



Fig. 7. Error estimates (upper: median, lower: maximum) of longitudinal strain measures. Errors vs. overall strain (left) and image rank index (right) with template sizes top to down being: 15, 19, 23, 27, and 31. The right sub-figures show that errors accumulate mostly due to multiple evaluations of strains, although at each evaluation step new image templates are sampled and rounding errors are taken into account at each evaluation step.

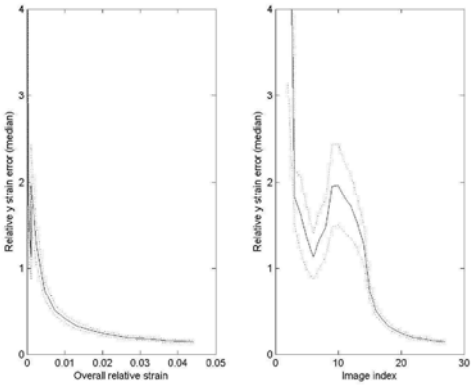



Fig. 8. Relative y strain median error vs. overall strain (left) and image rank index (right). Fortunately, overall strain grows faster than errors accumulate and thus relative error approaches zero as strain increases.

increasing the template was reduced significantly. The results also indicate that the intra-validation deviation usually increases as the error increases.

The results so far suggest that the update step $\Delta k = 1$ should be increased in order to avoid accumulation of errors. In run 2, different update steps were tested. The results are shown in Fig. 10, where the update steps are given in strain units (relative change of length) and the error is measured at $k = 27$, i.e. at the last image.

The results are in line with the presumption of accumulating errors. However, with the longest update step, which yielded only one update, the strain error increases. This is perhaps explained by the fact that the localization of deformed patterns with a fixed scale template becomes difficult. A minimum error is in this case obtained when the update step was approximately 0.02 strain units.

In the 3rd and 4th runs, the theoretical relationship (13) between the spatial strain resolution and accuracy was empirically tested. In Fig. 11, the accuracy estimates are shown against the spatial resolutions; the results confirm

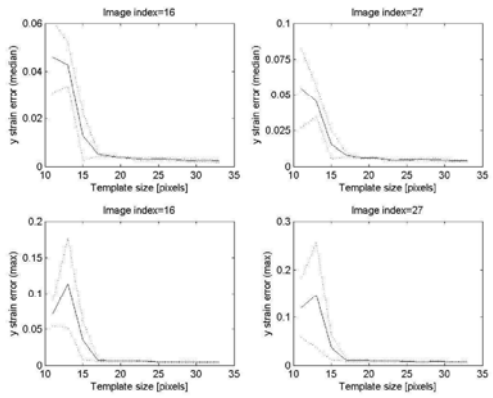



Fig. 9. Both median (upper) and maximum (lower) error estimates decrease as the template is grown. Results are consistent for small deformation (left) and a greater one (right).

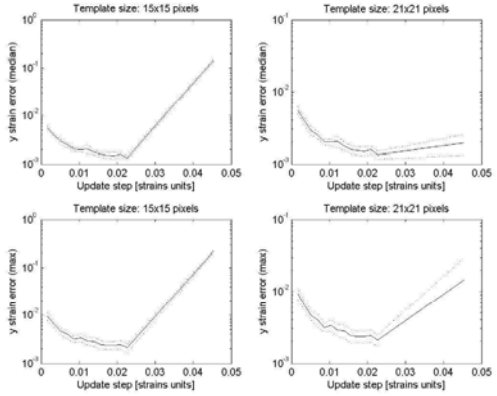



Fig. 10. Influence of the update interval on the estimated strain error (log scale). Results indicate that the minimum error is in this case achieved, when the update is done after 2 % strain.

the linear relationship.

The max (16) and median (17) estimates are compared as for y strain, and they both give similar results, except for magnitude. The two possible directions of error estimates, row-wise and column-wise, are compared with x strains. Row-wise estimates seem to be greater than the column-wise estimates, but qualitatively similar.

The ranges of spatial resolution for x and y directions were different due to the shape of the test specimen. When x and y row-wise strain errors are combined in the same plot (Fig. 12), it is seen that the errors lie on a single straight line. As a conclusion, row-wise errors in both directions behave similarly with respect to spatial strain resolution, and the algorithm does not cause any systematic error for any direction. Moreover, this discovery can be regarded as a proof of correctness for the implicit accuracy estimation method: x and y accuracy estimates give similar results, although the y strain is of order 0.04 and the x strain only of order -0.015.

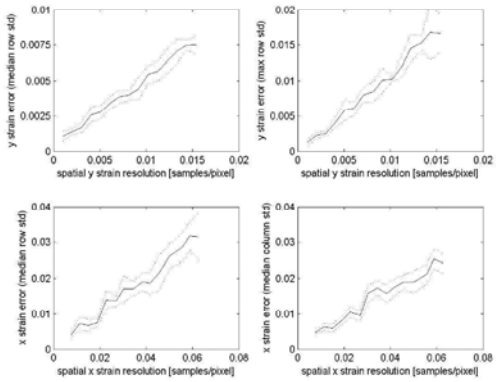



Fig. 11. Strain accuracy is a linear function of spatial strain resolution as expected. Median (upper left) and maximum (upper right) error of y strain are similar except as for magnitude. Row-wise x strain errors estimates (lower left) are a little greater than the column-wise (lower right) estimates.

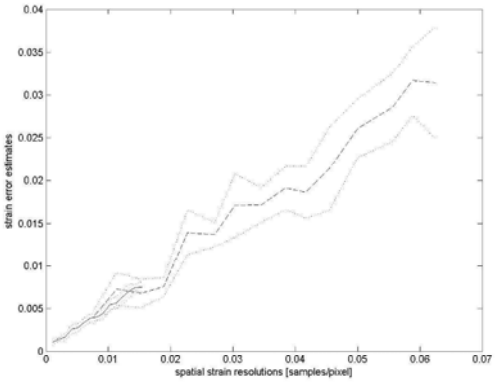



Fig. 12. Combination of y strain median error vs. y resolution (solid) and x strain row-wise median error (dashed) shows that errors in both directions lie on the same straight line, i.e. errors in both direction behave similarly with respect to spatial strain resolution.

## VI. Conclusion

The accuracies and resolutions of an image based strain measurement algorithm were studied in unsupervised, implicit statistical means. The results were validated using another implicit intra-validation method. The theoretical relationships between spatial strain resolution and accuracy of strain measurements were studied.

An image series of an aluminum test sample covered by random black painting pattern was taken during uniaxial tensile test. Strains were measured using the digital image correlation algorithm with varying parameters. According to the results, the template size should be at least $17 \times 17$ pixels and it was optimal to update the templates after about every 2 % increase of strain. The linear relationship between spatial strain sampling frequency and the measurement accuracy was also confirmed empirically.

In the future, the digital image correlation based strain measurement algorithm could be developed to be more accurate and faster by using e.g. camera calibration, dynamic template size control, and multi-resolution strain measurements. By estimating the reliability of correlation, the unreliable localizations could be recomputed with a larger template. The multi-resolution approach should detect the area of plastical deformation. Hence, the strain maxima near the fracture point can be measured automatically with a high resolution and a reasonable computation time. The algorithm could be optimized using a multi-objective function, and perhaps also tested by artificial images generated e.g. by genetic algorithms.

## Acknowledgment

We would like to thank Mika Billing for his assistance with the tensile testing machine. Seppo Kivivuori, Tuomas Katajarinne, and Lotta Vihtonen are acknowledged for collaboration in material engineering issues, and Pekka Isto for technical assistance.

## References

[1] R. C. Juvinall, and K. M. Marshek, *Fundamentals of Machine Component Design*, 4th ed., John Wiley & Sons, 1983.

[2] R. C. Conzalez and R. E. Woods, *Digital Image Processing*. 2nd ed., New Jersey: Prentice-Hall, 2002, p. 703.

[3] P.-C. Hung and A. S. Voloshin, "In-plane strain measurement by digital image correlation," *J. of the Braz. Soc. of Mech. Sci. & Eng.*, vol. 25(3), pp. 215-221, July-Sept. 2003.

[4] S. Bergonnier, F. Hild, and S. Roux, "Digital image correlation used for mechanical tests on crimped glass wool samples," *J. Strain An. Eng. Design*, vol. 40, pp. 185-197, 2005.

[5] S. Yoneyama and Y. Morimoto, "Accurate displacement measurement by correlation of colored random patterns," *JSME Int. J.*, Series A, vol. 46(2), pp. 178-184, 2003.

[6] T. C. Chu, W. F. Ranson, M. A. Sutton, and W. H. Peters, "Applications of digital-image-correlation techniques to experimental mechanics," *Exp. Mech.*, vol. 25(3), pp. 232-244, 1985.

[7] J. Sirohi, and I. Chopra, "Fundamental understanding of piezoelectric strain sensors," *Proc. of SPIE.*, vol. 3668, pp. 528-542, June 1999.

[8] N. Kiesewetter, and P. Schiller, "The acoustic emission from moving dislocations in aluminum," *Physica Status Solidi (a)*, vol. 38(2), pp. 569-576., Dec. 1976.

[9] S. R. McNeill, W. H. Peters, W. F. Ranson, and M. A. Sutton, "A study of fracture parameters by digital image processing," in *Proc. of 18th Midw. Mech. Conf.*, vol. 12 , pp. 267-271, 1983.

[10] G. Vendroux, and W. G. Knauss, "Submicron deformation field measurements: Part 2. Improved digital image correlation," *Exp. Mech.*, vol. 38(2), pp. 86-92, 1998.

[11] T. Schmidt, J. Tyson, and K. Galanulis, "Full-field dynamic displacement and strain measurement using advanced 3D image correlation photogrammetry," *Exp. Tech.*, vol. 27(3), pp. 47-50, 2003.

[12] A. Asundi, and B. Zhao, "Optical strain sensor using position-sensitive detector and diffraction grating: error analysis," *Opt. Eng.*, vol. 39(6), 2000.

[13] Y. Y. Hung, and C. Y. Liang, "Image-shearing camera for direct measurement of surface strains," *Applied Opt.*, vol. 18(7), pp. 1046-1051, April 1979.

[14] M. Spajer, P. K. Rastogi, and J. Monneret, "In-plane displacement and strain measurement by speckle interferometry and moiré derivation," *Applied Opt.*, vol. 20(19), pp. 3392-3402, Oct. 1981.

[15] D. W. Robinson, "Automatic fringe analysis with a computer image-processing system," *Applied Opt.*, vol. 22(14), pp. 2169-2176, July 1983.

[16] R. Feiel, and P. Wilksch, "High-resolution laser speckle correlation for displacement and strain measurement," *Applied Opt.*, vol. 39(1), pp. 54-60, Jan. 2000.

[17] B. G. Zagar, and C. Kargel, "A laser-based strain sensor with optical preprocessing, *IEEE Instrumentation and Meas.*, vol. 48(1), 97-101, Feb. 1999.

[18] B. G. Zagar, and S. C. Schneider, "A laser-optical strain gauge and its application in material testing," *IEEE Instrumentation and Meas.*, pp. 733-736, 2002.

[19] M. Anwander, B. G. Zagar, B. Weiss, and H. Weiss, "Noncontacting strain measurement at high temperatures by digital laser speckle technique," *J. Pressure Vessels and Piping*, vol. 80(6), pp. 361-365, 2003.

[20] R. K. S. Raman, and R. Bayles, "Detection of decohesion/failure of paint/coating using electronic speckle pattern interferomety," *J. Eng. Fail. Anal.*, vol. 13, pp. 1051-1056, 2006.

[21] R.Y. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses", *Robotics and Automation*, vol. 3(4), pp. 323-344, 1987.

[22] S. Iqbal, and A. Asundi, "Multipoint diffraction strain sensor: theory and results," *Meas. Sci. Technol.*, vol. 17. 2006.

[23] S. Roux, F. Hild, and S. Pagano, "A stress scale in full-field identification procedures: A diffuse stress gauge," *Eur. J. Mech. A/Solids*, vol. 24, pp. 442-451, 2005.

[24] H. W. Schreier, J. R. Braasch, and M. A. Sutton, "Systematic errors in digital image correlation caused by intensity interpolation," *Opt. Eng.*, vol. 39(11), 2000.

[25] S. S. Gleason, M. A. Hunt, and W. B. Jatko, "Subpixel measurement of image features based on paraboloid surface fit," in *Proc. of Machine Vis. Sys. Integ. in Ind.*, 1990.

[26] R. Kakarala, and A. O. Hero, "On achievable accuracy in edge localization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14(7), pp. 777-781, 1992.

[27] C. Su, and L. Anand, "A new digital image correlation algorithm for whole-field displacement measurement," in *Proc. of Innovation in Manuf. Sys. and Tech.* (IMST), 2003.

[28] H. Lu, and P. D. Cary, "Deformation measurements by digital image correlation: Implementation of a second-order displacement gradient", *Exp. Mech.*, vol. 40(4), pp. 393-400, 2000.

[29] C. F. Olson, "Maximum-likelihood template matching," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 52-57, 2000.

[30] W. Ruyten, "Subpixel localization of synthetic references in digital images by use of an augmented template", *Opt. Eng.*, vol. 41(3), pp. 601-607, 2002.

[31] H. Foroosh, J. B. Zerubia, and M. Berthod, "Extension of phase correlation to subpixel registration," *IEEE Transactions on Image Processing*, vol. 11(3), pp. 188-200, Mar. 2002.

[32] H. Stone, M. Orchard, and E.-C. Chang, "Subpixel registration of images," in *Proc. of Signals, Systems, and Computers*, vol. 2, pp. 1224-1452, 1999.

[33] S. Rocco, M. Claudio, and N. Guarata, "The Use of Interval Arithmetic as an Alternative Method to Evaluate Uncertainty in Input-Output Models," in *Proc. 14th Int. Conf. Input-Output Techniques*, 2002.

[34] J. S. Milton, and J. C. Arnold, *Introduction to probability and statistics: Principles and applications for engineering and the computing sciences*, 4th ed., Boston: McGraw-Hill, 2003.

# Dynamic template size control in digital image correlation based Strain measurements

Janne Koljonen, Olli Kanniainen, Jarmo T. Alander

Department of Electrical Engineering and Automation, University of Vaasa,
P.O. Box 700, FIN-65101, Vaasa, Finland. Email: *Firstname.Lastname*@uwasa.fi

## ABSTRACT

Image matching is a common procedure in computer vision. Usually the size of the image template is fixed. If the matching is done repeatedly, as e.g. in stereo vision, object tracking, and strain measurements, it is beneficial, in terms of computational cost, to use as small templates as possible. On the other hand larger templates usually give more reliable matches, unless e.g. projective distortions become too great. If the template size is controlled locally dynamically, both computational efficiency and reliability can be achieved simultaneously. Adaptive template size requires though that a larger template can be sampled anytime.

This paper introduces a method to adaptively control the template size in a digital image correlation based strain measurement algorithm. The control inputs are measures of confidence of match. Some new measures are proposed in this paper, and the ones found in the literature are reviewed. The measures of confidence are tested and compared with each other as well as with a reference method using templates of fixed size. The comparison is done with respect to computational complexity and accuracy of the algorithm. Due to complex inter-actions of the free parameters of the algorithm, random search is used to find an optimal parameter combination to attain a more reliable comparison. The results show that with some confidence measures the dynamic scheme outperforms the static reference method. However, in order to benefit from the dynamic scheme, optimization of the parameters is needed.
**Keywords:** adaptivity, machine vision, optimization, pattern matching, tensile testing.

## 1 INTRODUCTION

Many signal processing techniques include a window in one or another way. Common window methods are filtering and pattern matching. In most of the cases, window size and shape are fixed. The selection of these key parameters can be done *ad hoc*, using experience, by making a compromise, or after learning or optimization. Adaptive window shape and size control strategies have been developed to overcome the drawbacks of fixed parameters. In this paper, adaptive (or dynamic) parameter control means local control decisions and not e.g. global optimization, although optimization methods can be used to train the controller. Dynamic parameter control can help avoid compromises and attain more optimized performance, particularly in a heteroskedastic environment.

Adaptive window methods have been applied at least in median filtering[1] and template matching[2-9]. In several research papers, adaptive window size template matching has been applied to disparity map creation in stereo vision algorithms[2-7]. Another typical application is target tracking[8-9]. On the contrary, there does not seem to be any applications on strain measurements, although a lot of papers on digital image correlation based strain measurements have been published.

The adaptive template size strategies typically rely on measures of confidence (reliability, (un)certainty) of match and local sequential search. In many papers[3, 5] the original template size is $3 \times 3$, almost the smallest possible, which is then gradually increased. Kanade and Okutomi[3] expands the template by one pixel, individually to all four principal directions, until is a measure of uncertainty begins to increase. Izquierdo[5] in turn use a model of elastic contour and an energy function approach.

Usually only accuracies but not the computational costs are taken into account when designing and testing adaptive window size algorithms. Actually, Giachetti[10] used a fixed window size in image motion measurement due to a relevant speed reduction when comparing to using an adaptive window size algorithm. A fixed window is selected to give a trade-

Intelligent Robots and Computer Vision XXV: Algorithms, Techniques, and Active Vision
edited by David P. Casasent, Ernest L. Hall, Juha Röning
Proc. of SPIE Vol. 6764, 67640L, (2007) · 0277-786X/07/$18 · doi: 10.1117/12.732355

off between the good properties of large and small windows. On the other hand, Giachetti introduced several methods, such as mask sub-sampling and fast search strategies, to reduce complexity.

Gong and Yee-Hong[7] in turn use genetic algorithms to optimize stereo matching and a multi-resolution approach to represent possible disparity maps. They claim that multi-resolution matching and adaptive window size have a similar influence. However, if the multi-resolution quad-tree is always computed exhaustively, an adaptive window size approach can be computationally more efficient.

This paper proposes confidence measures for evaluation of pattern localization reliability and accuracy. The confidence measures are tested in a digital image correlation based strain measurement algorithm. The objective of this study is to find out whether the dynamic template size control improves the measurement accuracy without an increment to the overall computational complexity and which confidence measures are applicable for this purpose. In Section 2, the strain measurement algorithm and its accuracy analysis is introduced in short. Section 3 discusses the general requirements for confidence measures, reviews the confidence measures found in the literature and proposes new ones. Section 4 includes the experimental setup and test run details, whereas in Section 5, the results are given and discussed.

## 2 DIGITAL IMAGE CORRELATION BASED STRAIN MEASUREMENTS

Digital image correlation is a full-field method to measure strains of materials. The principle is to track homologous points of material using images taken before and after deformation. The strains are calculated from the displacements of these points. Next the strain measurement algorithms and an implicit method to estimate the measurement errors are introduced.

### 2.1 Algorithm

Points on a regular rectangular grid are selected from the non-deformed image. Let $\mathbf{c}_{0,0} = (x_{0,0}, y_{0,0})$ and $\mathbf{c}_{I,J} = (x_{I,J}, y_{I,J})$ be the locations of the first and last nodes of the grid, respectively. The distance between nodes in x is thus

$$\Delta x = \left(x_{I,j} - x_{0,j}\right)/I \,. \tag{1}$$

Templates called Zones-Of-Interest (ZOI) are sampled around the grid points. The dimensions of the ZOI are denoted by $Z_x$ and $Z_y$. The template size is varied locally in this study. Details of the dynamic template size control are given in Section 3. The templates are used to localize the points to be tracked in the images taken from a test piece after deformation. If the deformation is noticeable, it causes scale distortion, which can be taken into account either scaling or updating the template after a certain amount of strain. In the algorithm used in this paper, the template is updated after each localization step, i.e. the pixels of the ZOI are replaced by the values around the matched position. Unfortunately, the update procedure induces nonsystematic error accumulation of the localization errors.

The localization of a grid points is carried out by translating the ZOIs over a Region-Of-Interest (ROI) and comparing the overlapping pixels using a similarity metric. In this study, normalized correlation was used. ROI is determined by linear prediction, i.e. the predicted position is extrapolated using the two previous locations of each grid point. For the first deformed image no prediction can be done. The ROI dimensions with respect to the predicted position are denoted by: $R_x$ in positive and negative transversal directions, $R_{y-}$ in the direction of negative longitudinal strain, and $R_{y+}$ in the direction of positive longitudinal strain.

The principal engineer strains transversally (x) and longitudinally (y) are computed from the grid coordinates. In this study, four neighboring nodes are used so that the x and y strains can be evaluated at the same spatial position. The x strain is (y strain is defined similarly):

$$\varepsilon_{x,i,j} = \frac{\left(x_{i+1,j}^{(k)} - x_{i,j}^{(k)}\right) + \left(x_{i+1,j+1}^{(k)} - x_{i,j+1}^{(k)}\right)}{2\left(x_{i+1,j}^{(1)} - x_{i,j}^{(1)}\right)} \,. \tag{2}$$

### 2.2 Implicit accuracy analysis

If no explicit validation method for the strain measurements (or grid point localization) is available, the following implicit approach can be used. The implicit method is based on statistical analysis and on the assumption that the transversal distribution of strains is uniform. The assumption should approximately hold as long as the test piece is homogeneous, and more positively if the stretch is low.

The strain errors are estimated by the standard deviations of strains measured in a transversal row of the grid. The error estimate for x and y strain are respectively

$$\hat{\delta}\varepsilon_{x,\cdot,j} = \sqrt{\sum_{i=0}^{I-1}(\varepsilon_{x,i,j} - \sum_{i=0}^{I-1}\varepsilon_{x,i,j}/I)^2/(I-1)} \tag{3}$$

and

$$\hat{\delta}\varepsilon_{y,\cdot,j} = \sqrt{\sum_{i=0}^{I-1}(\varepsilon_{y,i,j} - \sum_{i=0}^{I-1}\varepsilon_{y,i,j}/I)^2/(I-1)}\,. \tag{4}$$

Now there are $J$ error estimates for x and y strain errors. To be able to compare error the error estimates are compressed using e.g. mean, median, or max of the row-wise estimates. Moreover, to obtain an overall accuracy estimate and to easily compare small differences of small error a product of inverse x and y errors is computed. E.g. maximum row error can be used to have a cautious accuracy estimate:

$$Accuracy = 1/\left[\max_j\left(\hat{\delta}\varepsilon_{x,\cdot,j}\right)\max_j\left(\hat{\delta}\varepsilon_{y,\cdot,j}\right)\right]. \tag{5}$$

## 3 DYNAMIC TEMPLATE SIZE CONTROL

Dynamic template size control is based on reliability or confidence measures of correlation matches. Confidence measures can be in turn based on the information of the template, correlation or similarity measure surface and on the previous match values, coordinates, or other previous states of the algorithm. Since there can be several reliability measures (features), an overall reliability decision have to be done by combining the inputs into an output value or values that control template shape parameters. A strategy to control the template shape and size as well as probably other algorithm parameters also has to be selected.

### 3.1 Confidence measures

A confidence measure, as meant here, evaluates the certainty that a localized pattern is the correct one. Namely, the more the features used in pattern matching include noise, the more probable it is that a noise pattern is localized instead of the right one. In image pattern matching, the localization accuracy is reduced, in addition to the increasing change of mismatch, when the amount of noise increases. Confidence measures should also reflect the localization accuracy.

The confidence measures reported in this paper can be divided into three categories: measures using information of the template, measures studying correlation surface, and measures based the previous states of the algorithm. The confidence measures that use only the template estimate the matching certainty without actually evaluating matches or localizing patterns. Thus they have an advantage as for computational complexity, but on the other hand, the information content is inadequate as the information of the Region of Interest (ROI), including its noise level etc., is not included in the confidence estimate.

Confidence measures based on the correlation surface are on the contrary computationally more complex as the whole correlation surface and usually the sub-pixel correlation maximum peak are computed. On the other hand, the correlation surface reflects many effects caused by the information of the template and ROI as well as the correlation or similarity

metric. Hence, e.g. the increase of noise level of ROI should change the shape of the correlation surface. The previous states that can be utilized are e.g. the localized coordinates using a smaller template or confidence values of previous matches.

Confidence measures based on the correlation surface can be regarded as a sub-domain of evaluation of fitness landscapes. More specifically a general problem can be stated: Given a noisy fitness landscape, evaluate the certainty that an optimum found is the global optimum, and evaluate how accurate its value is. Measures using only the template can in turn be compared with approaches that evaluate an optimization problem *a priori* in order to e.g. select the best optimization method to be used[11]. Now the purpose is to select the best template shape and size, and possibly the correlation method. However, there is no one-to-one analogy between optimization and pattern matching: In terms of optimization, the *a priori* evaluation methods usually study a sample of fitness landscape. In pattern matching there is no analogues approach as the shape of the correlation surface depends on the template, ROI, and correlation method, whereas in optimization the fitness landscape is given, may it though be stochastic.

### 3.1.1 General conditions

Some general conditions for an efficient confidence measure can be set. The underlying requirement is that it should correlate with the certainty and accuracy of match. However, these fundamental properties cannot be measured directly in the approach described in this paper, although some statistical methods using controlled imaging setups or artificial images could be used to study them, too. Nevertheless, some indirect conditions and examination of the confidence measure efficiency are proposed and studied instead.

In the strain measurement approach, there is no method to evaluate the accuracy of a single pattern localization. Instead, the accuracy of the strain measurement can be estimated. Now if the strain estimation accuracy correlates with the template size, as it does, the confidence measure should subsequently correlate with the template size. Furthermore, the confidence measure should also correlate with the strain estimation accuracy, which however can be tested only indirectly by comparing confidence measure threshold requirements with the resulting strain estimation accuracies.

The existence of the correlations between strain template size and confidence measure does not solely suffice to guarantee an efficient confidence measure. In addition, a given template size should result in varying confidence measure values depending on the noise level and quality of images and its features i.e. the confidence measure should correlate with the image quality properties that have an influence on pattern matching quality.

Actually, the potential existence of these correlations is what makes the use of dynamic template size control attractive. Specifically, if the pattern matching is done in varying conditions and with varying image and template quality, the template size and shape can be optimized, with respect to localization certainty and accuracy as well as computational complexity, adaptively and individually for each case. The computational advantage requires that the matching cases include enough variation. In that case the computational cost of first evaluating the confidence measures and then doing the match with varying template size can be less than the cost of using a large enough template at each localization.

### 3.1.2 Reported measures

There various approaches for confidence or reliability measures for template matching reported in the literature. Examples from the different categories (template based, correlation function, algorithm state) also exist. As cited by Izquierdo[5] and Gong and Yang[6] the earliest work concerning adaptive window size was carried out by Levine et al. (1973)[12], who used intensity variation as a *a priori* confidence estimate.

In stereo matching, Kanade and Okutomi[3] uses an algorithm state approach, where the change of disparity and image intensity fluctuations are monitored in order to stop the template expansion, when the uncertainty begins to increase. Izquierdo[4-5] in turn uses a linear combination of two reliability measures. The first term measures stereo-motion consistency and the second one analyzes correlation surface. The latter is based on surface curvature $K$ and minimum Sum-of-Squared-Distance ($SSD_{min}$)[4]:

$$f = K/(K + SSD_{min} + S), \tag{6}$$

where $S$ is a proper constant.

### 3.1.3 Proposals

Now some elementary (primary) confidence measures are proposed. These measures are supposed to fulfill at least partly the general conditions dealt in Section 3.1.1. In order to achieve better performance the elementary measures can be combined arithmetically to a compound (secondary) measures. Alternatively, primary measures can span a multi-dimensional feature space that is used as an input for the controller.

Predictive confidence estimates derived from a template (Zone-Of-Interest, ZOI) should evaluate the strength and quality of texture concerning pattern matching; a larger template should be sampled from a low-textured or noisy region. Simple measures of template texture quality are:

- Standard deviation of ZOI intensities ($STD_{ZOI}$) and
- Entropy or normalized entropy of intensities ($E_{ZOI}$).

Each of these should increase with increasing strength of texture. However, noise increases them, too, which is an undesired property. More complex template analysis should separate noise e.g. by filtering and analyze noise intensity and the remaining texture separately. The template could also be subdivided into local regions, whose properties were studied and compared to analyze texture more closely.

Next confidence measures calculated from the correlation function ($r$) are proposed. Alternatively the correlation function can be a surface obtained by matching with a similarity or dissimilarity metric such as Sum-of-Squared-Distance. First of all the confidence measures proposed for ZOI (standard deviation ($STD_r$) and entropy ($E_r$) computed either globally or locally around the correlation peak) apply for correlation surface, too, although they are rather rough estimators of the localization reliability and accuracy.

A generic approach is to study the local correlation maxima. Let $r_{max(i)}$ be the i$^{th}$ highest correlation peak. How to define and detect the local optima is another problem. In this study, they are detected recursively. The largest correlation value is looked for first ($r_{max(1)}$) from the whole correlation surface. The search region for the next peak ($r_{max(2)}$) is the correlation surface without a region around the first peak; pixels closer than half of the template edge ($Z_x$) are neglected. The third peak is localized neglecting the neighborhoods of the first two peaks and so on.

Using the correlation maxima two useful confidence measures are proposed:

- Maximum correlation ($MC$): $r_{max(i)}$ and
- the difference of the first two peaks: $r_{max(i)}-r_{max(i)}$. Let us call this discrimination gap ($DG$). This has already previously been introduced by the authors[13].

Figure 1 shows a one-dimensional example of the determination $r_{max(1)}$ and $r_{max(2)}$ the proposed confidence measures. It is intuitively obvious that the closer from one another the first two peaks are, the more uncertain it is that the highest peak is the correct match. On the other hand, the noise level of the correlation function should be taken into account. The more noise it includes, the more probable it is that the matcher has found an artificial noise pattern instead of the correct object. The noise level could be approximated e.g. by local standard deviation or by fitting a parabola to the correlation peak and computing the root-mean-square-error from the residuals. The discrimination gap could then be normalized by the noise estimate to obtain a variable for statistical tests.

An approach already mentioned in Section 3.1.2 is to measure the curvature of the highest correlation peak. The Gaussian curvature of a surface is the product of the principal curvatures, $k_1k_2$, which are respectively the maximum and minimum 1D curvature values of all possible direction. Sometimes the principal curvatures are replaced for simplicity by the horizontal and vertical curvatures, $k_x$ and $k_y$, respectively, and the Gaussian curvature is replaced by

$$K = k_x k_y . \tag{7}$$

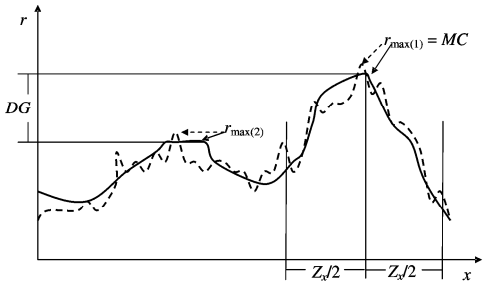



Figure 1: One-dimensional correlation function and the principle to determine maximum correlation (*MC*) and discrimination gap (*DG*). Two correlation functions: without noise (solid) and with noise (dashed).

For explicitly defined functions the curvature is:

$$k_x = \frac{\partial^2 f}{\partial x^2} \Big/ \left(1 + \left(\frac{\partial f}{\partial x}\right)^2\right)^{3/2}. \tag{8}$$

If a parabola is fit to the correlation peak, the horizontal and vertical curvatures are obtained as follows. The equation of a parabola is:

$$f(x,y) = a_{x^2}x^2 + a_{y^2}y^2 + a_{xy}xy + a_x x + a_y y + a_0. \tag{9}$$

Applying (8) to (9), the curvature $k_x$ of the parabola becomes:

$$k_x = 2a_{x^2} \Big/ (1 + 4a_{x^2}^2 x^2 + 4a_{x^2}a_{xy}xy + 4a_{x^2}a_x + 4a_{xy}^2 y^2 + 2a_{xy}a_x y + a_x^2)^{3/2}. \tag{10}$$

Curvature measures the steepness of the peak. Izquierdo[4] presumes that a flat correlation surface with low curvature in the peaks is related to a low textured region and a clear peak to reliable match. However, again a random match with a high-frequency noise pattern may bring a narrow peak to the correlation, whereas a clear pattern with lower frequency generates a flatter pattern with only little noise. Consequently, an improvement to the pure curvature might be to estimate the frequency contents of the template and ROI and to study the residuals of the fitted parabola. Another problem is that curvature seems to decrease as template is enlarged. Therefore the curvature based confidence measure used in this study is formed by multiplying curvature by the template size:

$$f = KZ_xZ_y. \tag{11}$$

Equation (6) is used in this study, but it is modified for normalized correlation:

$$KMC = K/(K + 1/(MC + 1) + S), \tag{12}$$

For the last category of confidence measures, namely the algorithm state approach, the following simple methods are proposed: Compute the difference of translation vectors from localizations with different template sizes, shapes or different matching metrics. Let $\mathbf{v}_1$ and $\mathbf{v}_2$ be two translation vectors obtained with different parameters. If the translation difference (*TD*)

$$TD = \|\mathbf{v}_2 - \mathbf{v}_1\| \tag{13}$$

is small, the matching is considered reliable. Averaging locations measured with different template sizes, shapes, or different matching metrics may also improve the localization accuracy. The confidence value $TD_{AVG}$ is equal to (13) but the translation vector becomes

$$\mathbf{v}_{avg} = (\mathbf{v}_1 + \mathbf{v}_2)/2. \tag{14}$$

### 3.2 Template size control strategy

The template model used in this study is a square template with every pixel included in the computation of correlation. The template has one free parameter: size ($Z_x = Z_y$). The confidence measure obtained from a pattern match is compared to a threshold (*thres*) value to make a binary decision, whether to increase the template or not. The aim is to result in an almost equal confidence value in each node match.

The control algorithm has three static parameters: the original template size ($Z_0$), template size increment ($Z_{inc}$), and the maximum number of increments ($N$). All these parameters have an influence on the control path and subsequently strain computation speed and accuracy. They should be selected roughly so that the ranges of possible template sizes and template size resulting in the confidence threshold are equal. The following pseudo-code implements the control algorithm and returns the resulting template size:

```
function templateSize(Z0, Zinc, N, thres)
      Zx ← Z0; i ← 0;
      while (i<N & computeConfidence(Zx)<thres)
            i ← i+1; Zx ← Zx + Zinc;
      end while;
      return Zx;
end function;
```

The control algorithm looks for the smallest number of increments $n = 0, 1, \ldots, N$ so that

$$confidence(Z_x^{(n)}) = confidence(Z_0 + nZ_{inc}) \geq thres, \tag{15}$$

where *confidence*() computes the confidence measure, and uses $Z_x^{(n)}$ or the maximum template size

$$Z_x^{(\max)} = Z_x^{(N)} = Z_0 + NZ_{inc} \tag{16}$$

to the grid node localization. However, note that as for e.g. *TD* the objective is to go under the given threshold.

### 3.3 Control of search area

In case of correlation surface or algorithm state based confidence measures, there are at least two potential reasons for a low confidence value: the template is too small or ROI does not contain the point that is looked for. In the latter case, the solution is to increase ROI. However, it is difficult to determine, which one of the reasons is in question[1]. A ROI growth parameter was added to the algorithm. Each time the template is grown by $Z_{inc}$, ROI is grown by $R_{inc}$ into all four directions.

## 4 EXPERIMENTS

The objective of the experiments was to estimate the strain measurement accuracy as a function of computation complexity for each confidence measure. The fixed size template approach is used as a reference method. In addition, correlations between template size, confidence measures, and accuracies are studied.

[1] If the point was just outside the ROI, the correlation surface could give an indication of an existing peak.

For each confidence measure, the strain measurement algorithm was run 200 times with some randomized parameters (see Table 1 and Section 4.2.1). The reference method with fixed template size was run 400 times to achieve a more reliable comparison. The last three runs used pipelined confidence measures. The confidence values were recorded at template match. The strain accuracies and computation complexities were evaluated as described in Section 4.2.

### 4.1 Dataset and fixed parameters

The data set consisted of three images taken from a typical aluminum 'dog-bone' test piece used in uni-axial tensile tests. The dimensions of the narrow part of the test piece are approximately 47 × 8 × 2 mm. Nikon D200 digital camera with 2592 × 3872 pixels (3×8 bit RGB) was used. The narrow part of the test piece i.e. the area of interest was about 330 × 1900 pixels.

Random black speckle patterns were spray-painted to the test piece before loading it in a tensile test machine. The first image is a reference image without deformation. Before the second and third images the test piece was subjected to a strain of approximately 2 % and 4%, respectively.

Because the test runs with high spatial strain resolution would require excessive computation time, the strain measurement grid was limited 4 × 4 nodes. The sub-pixel interpolation in localization was carried out using 3 × 3 neighborhood of the correlation maximum. The same fitted parabola was used to calculate the curvature according to (7).

### 4.2 Complexity and accuracy measures

The complexity of a run of the strain measurement algorithm is estimated by the number of multiplications in the computation of the cross-correlation, which is presumed to dominate the overall computation time, even though is does not include for example the computations of the template based confidence measures. This approach is beneficial, because it is deterministic and thus the implementation and running environment issues do not affect the result. The actual computation time using fixed implementation and environment is though also recorded for comparison.

The accuracies of the individual pattern localizations cannot be estimated directly. On the contrary, the accuracy of the strain measurement is estimated using the approach presented in Section 2.2 and particularly in (5). The strain accuracy implies how accurate results the combination of the free objective parameters and the confidence measure in use give. Next these objective parameters and the Monte Carlo optimization of them as well as the post-processing of the results are introduced.

#### 4.2.1 Objective parameters and random search

To obtain a reliable comparison of the different confidence measures the free parameters are varied randomly and the maximum accuracy for a given complexity is obtained by post-processing (Section 4.2.2) of the recorded pairs of accuracy and complexity. The following parameters are randomized between the test runs: ROI dimensions ($R_x$, $R_{y-}$, and $R_{y+}$), ROI size increment $R_{inc}$, original template size ($Z_0$), template size increment ($Z_{inc}$), the maximum number of increments ($N$), and the confidence threshold. The ranges of these parameters in the test runs are given in Table 1.

Table 1: Setups of the test runs. Test run 10 computes the mean according to (14).

| Test run | Confidence measure | Samples | *Thres* | ($R_x$, $R_{y-}$, $R_{y+}$, $R_{inc}$) | ($Z_0$, $Z_{inc}$, $N$) |
|---|---|---|---|---|---|
| 1 | none | 400 | none | (1–20, 1–20, 1–40, 0) | (5–79, 0, 0) |
| 2 | $STD_{ZOI}$ | 200 | 2–47 | (1–20, 1–20, 1–40, 0) | (5–25, 2–8, 2–8) |
| 3 | $E_{ZOI}$ | 200 | 3–8 | (1–20, 1–20, 1–40, 0) | (5–25, 2–8, 2–8) |
| 4 | $E_r$ | 200 | 0.005–0.05 | (1–20, 1–20, 1–40, 0–8) | (5–25, 2–8, 2–8) |
| 5 | *MC* | 200 | 0.3–1.0 | (1–20, 1–20, 1–40, 0–8) | (5–25, 2–8, 2–8) |
| 6 | *DG* | 200 | 0–0.85 | (1–20, 1–20, 1–40, 0–8) | (5–25, 2–8, 2–8) |
| 7 | *K* (11) | 200 | 0–10 | (1–20, 1–20, 1–40, 0–8) | (5–25, 2–8, 2–8) |
| 8 | *KMC* (12) ($S = 2$) | 200 | 0–1 | (1–20, 1–20, 1–40, 0–8) | (5–25, 2–8, 2–8) |
| 9 | *TD* (13) | 200 | 1.01–0.01 | (1–20, 1–20, 1–40, 0–8) | (5–25, 2–8, 2–8) |
| 10 | $TD_{avg}$ (13&14) | 200 | 1.01–0.01 | (1–20, 1–20, 1–40, 0–8) | (5–25, 2–8, 2–8) |

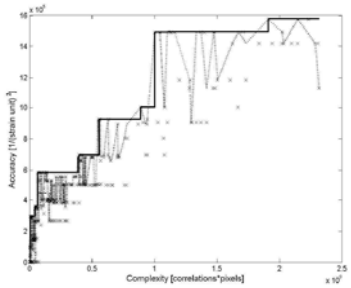



Figure 2: Example of maximum envelope filtering: ($complexity_i$, $accuracy_i$) pairs are (×) from run 1, dashed line is the function (15), and solid line is the corresponding filtered function (16).

#### 4.2.2 Post-processing

The obtained ($complexity_i$, $accuracy_i$) pairs form a set, where the majority of points are from runs with an inappropriate parameter combination. The accuracy is supposed to increase monotonically as the complexity increases. Hence, the interesting part of the point cloud is the monotonically increasing envelope of the maximum accuracies. Let $f(x)$ be a function $complexity \rightarrow accuracy$:

$$f(x) = \begin{cases} \max\limits_{i,\ complexity_i = x} (accuracy_i), \exists i \text{ such that} complexity_i = x \\ \text{undefined} \qquad\qquad \text{otherwise.} \end{cases} \tag{17}$$

The maximum envelope $f_{max}$ is thus obtained by filtering (17) (see also Figure 2):

$$f_{\max}(x) = \max_{i \le x} f(i) \cdot \tag{18}$$

## 5 RESULTS

In this section the results of the test runs are shown and discussed. First, the correlations between confidence measures and template size as well as template size and accuracy are studied to see, whether the general conditions of efficient confidence measures are fulfilled. Finally, the confidence measures and the fixed template method are compared as for complexity and accuracy.

### 5.1 Correlations between confidence values, template size, and accuracy

Figure 3 shows how the confidence values correlate with the template size. Confidence values are recorded at 16 different locations of the image, namely at the grid nodes. The results indicate that all of the confidence measures correlate with the template size and have also some sample position dependent cross-variation. However, the measures seem to have significant differences.

$STD_{ZOI}$ includes amply cross-variation, but the template size dependence is not always monotonous and sometimes even negative. $EZ_{ZOI}$ usually follows monotony but the variation is perhaps minor. These template based confidence measures have no inner-variation in contrast to the others, which depend also on the selection of ROI parameters.

$E_r$ seems to obey the conditions of monotony and cross-variation. *MC* in turn seems to saturate to its maximum at too small templates. Hence its cross-variation stays inferior and no discrimination of the match reliability can be done. The cross-variation of *DG* is promising, although it violates the monotony requirement. *K* has great cross-variation, but its nonmonotony is striking. *KMC* in turn has a similar shape as *K* and the modulation effect by *MC* is evident. *TD* decreases logarithmically when template size is increased linearly.

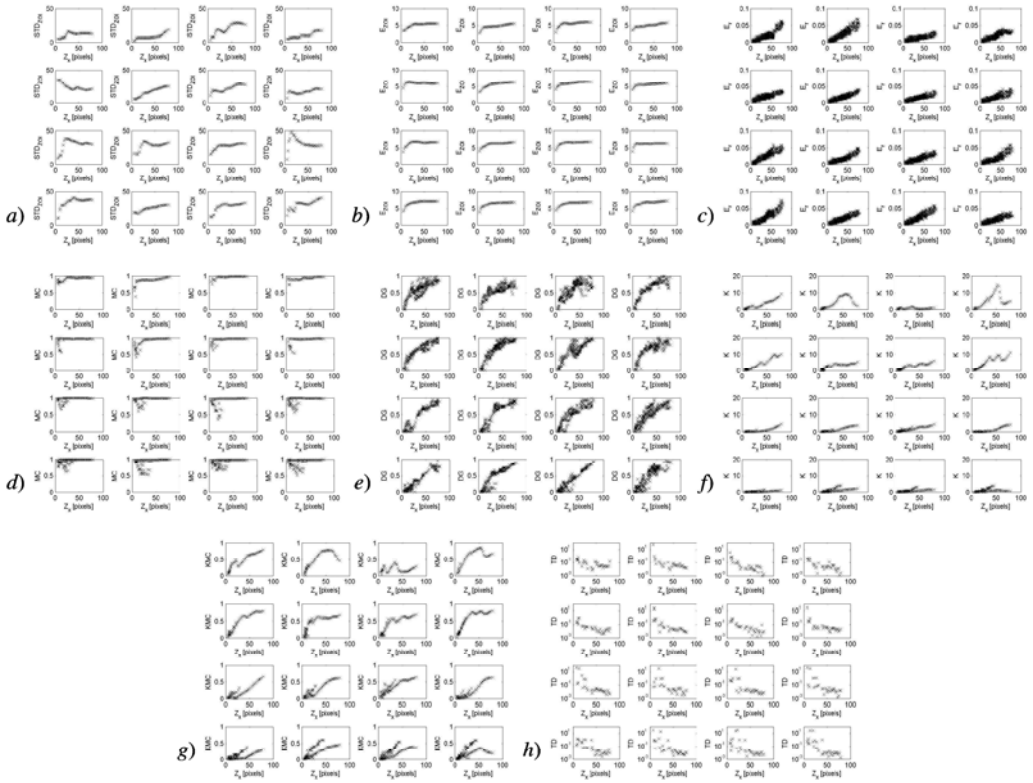


Figure 3: Confidence values acquired from the 16 grid nodes at the first deformed image plotted against template size (*a*: $STD_{ZOI}$, *b*: $E_{ZOI}$, *c*: $E_r$, *d*: *MC*, *e*: *DG*, *f*: *K*, *g*: *KMC*, *h*: *TD* (log scale)).

Figure 4 *a*) compares how the strain accuracies of run 1 correlate with the corresponding (fixed) template size used. As there is a positive correlation, it is justified to enlarge the template in order to obtain better accuracy. Figure 4 *b*) shows that the accuracy is gradually improved as the threshold of $STD_{ZOI}$ is increased. The results in Figure 3 and Figure 4

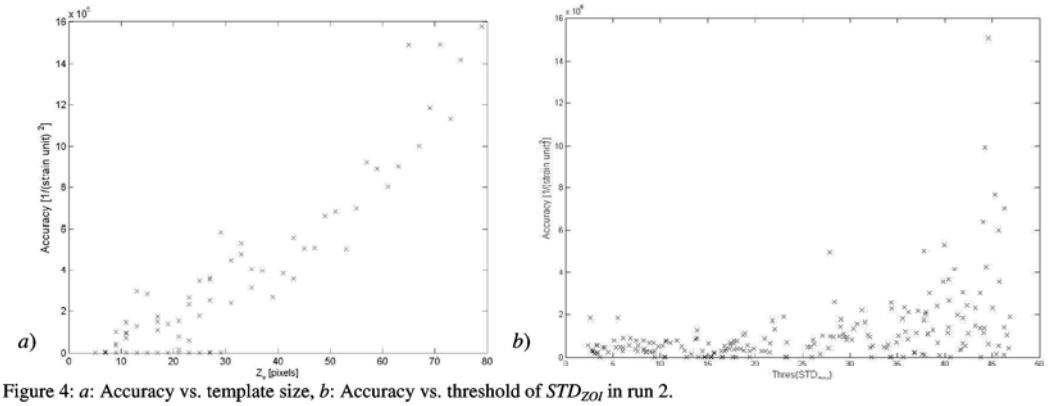



Figure 4: *a*: Accuracy vs. template size, *b*: Accuracy vs. threshold of $STD_{ZOI}$ in run 2.

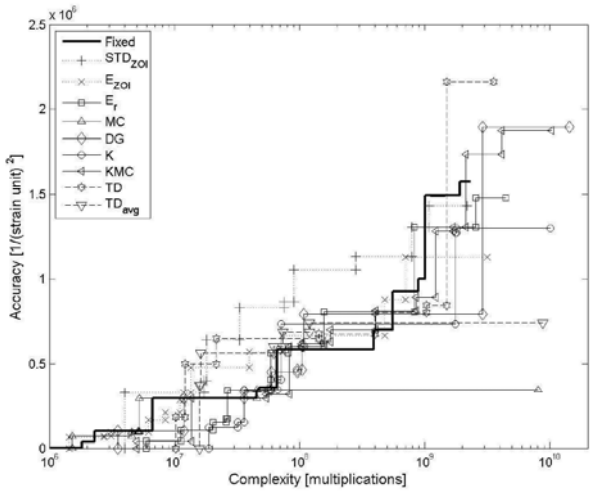



Figure 5: Accuracy vs. deterministic complexity (log scale).

indicate that the general conditions (Section 3.1.1) are at least partly fulfilled for the confidence measures proposed. These results are promising so that good accuracy-complexity efficiency could be achieved.

### 5.2 Complexities and accuracies

The comparison between the confidence measures is shown in Figure 5. The abscissa uses logarithmic scale of the number of multiplications in pattern matching. The results indicate that at the middle of the complexity range all of the confidence measures outperform the fixed template at least slightly. The most promising confidence measures seem to be $STD_{ZOI}$. For example, the accuracy requirement was $10^6$, $STD_{ZOI}$ would require only about one tenth of the number of multiplications in comparison to the fixed template method. The modified version of the method used by Izquierdo (*KMC*) seems to be comparable to template size modulated curvature *K*. The pure maximum correlation on the contrary actually seems to be the most inefficient of the tested method as was expected on the basis of the confidence template size correlations.

When the accuracy was plotted against the actual computation time the differences between the dynamic template and the fixed template size method were smaller. $STD_{ZOI}$ was still the best method. The computation times of $STD_{ZOI}$ and the fixed method at accuracy $10^6$ were about 48 and 160 s, respectively. Thus the runtime requirement of $STD_{ZOI}$ was about 30% of the fixed template method, a significant difference. The range of actual computation times was approximately 10 to 2000 s.

## 6 CONCLUSION

A method to dynamically control template size, a key parameter in pattern matching, was introduced. To control is accomplished by comparing a confidence value to a threshold value and by enlarging the template is the threshold requirement is not met. New confidence measures were proposed and tested. A material strain algorithm based on image correlation was used as a test case and an implicit strain measurement error estimate as the objective function as for pattern localization accuracy. Standard deviation of the template intensities proved to be the most efficient confidence measure of the tested ones as for accuracy and computation time. Sometimes it used even 90 % less multiplications to reach the same accuracy as the fixed template size method.

### 6.1 Future and suggestions

Dynamic template size control could be improved by creating more efficient confidence measures, more complex template models and control strategies, and by reducing computation time e.g. by sampling. Instead of a rectangular template more flexible shapes or even pixel-wise defined templates could be used. The control complex templates could be done e.g. by genetic algorithms, fuzzy logic or artificial neural networks.

E.g. neural networks could be used to control the algorithm parameters. A training set for a neural network could be obtained by looking for an optimal template size (and possibly other parameters) for each grid point and image, and by recording the corresponding confidence measures. Speed reduction could be achieved by computing a confidence measures estimate from a sample instead of computing the whole correlation surface or using the whole template.

## ACKNOWLEDGMENTS

Finnish Funding Agency for Technology and Innovation (TEKES) and the industrial partners of the research project *Process Development for Incremental Sheet Forming* have supported this research.

## REFERENCES

1. W. Zhou, J.A.R. Blais, "Adaptive median filtering using local entropy", in *Proc. of the Geoscience and Remote Sensing Symposium*, pages 285-288, 1990.
2. S. Lewicki, L. Meemong, P. Chodas, E. DeJong, "Stereo processing of Magellan SAR imagery performed on a transputerarchitecture", in *Proc. of the Geoscience and Remote Sensing Symposium*, Vol. 4, pages 1786-1788, 1993.
3. T. Kanade, M. Okutomi, "A stereo matching algorithm with an adaptive window: theory and experiment", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **16**(9): 920-932, 1994.
4. E. Izquierdo, "Stereo matching for enhanced telepresence in three-dimensional videocommunications", *IEEE Trans. on Circuits and Systems for Video Technology*, **7**(4): 629-643, 1997.
5. E. Izquierdo, "Disparity/segmentation analysis: matching with an adaptive window and depth-driven segmentation", *IEEE Trans. on Circuits and Systems for Video Technology*, **9**(4): 589-607, 1999.
6. M. Gong, Y-H Yang, "Multi-resolution stereo matching using genetic algorithm", in *Proc. of the IEEE Workshop on Stereo and Multi-Baseline Vision*, pages 21-29, 2001.
7. M. Gong, Y-H Yang, "Fast stereo matching using reliability-based dynamic programming and consistency constraints", in *Proc. of the Conf. on Computer Vision*, Vol. 1, pages 610-617, 2003.
8. D. A. Montera, S. K. Rogers, D. W. Ruck, M. E. Oxley, "Object tracking through adaptive correlation", *Optical Engineering*, **33**(1): 294-302, 1994.
9. S-I Chien, S-H Sung, "Adaptive window method with sizing vectors for reliable correlation-based target tracking", *Pattern Recognition*, **33**: 237-249, 2000.
10. A. Giachetti, "Matching techniques to compute image motion", *Image and Vision Computing*, **18**: 247-260, 2000.
11. C.R. Reeves, "Predictive measures for problem difficulty", in *Proc. of the Congress on Evolutionary Computation*, Vol. 1, pages 736-743, 1999.
12. M. Levine, D. O'Handley, G. Yagi, "Computer determination of depth maps", *Computer Graphics and Image Processing*, **2**(4): 131-150, 1973.
13. J. Koljonen, J.T. Alander, "Genetic algorithm for optimizing fuzzy image pattern matching", in *Proc. of the Ninth Scandinavian Conference on Artificial Intelligence*, pages 46-53, 2006.

# Searching strain field parameters by genetic algorithms

Janne Koljonen, Timo Mantere, Olli Kanniainen, and Jarmo T. Alander

Department of Electrical Engineering and Automation, University of Vaasa,
P.O. Box 700, FIN-65101, Vaasa, Finland. Email: *Firstname.Lastname*@uwasa.fi

## ABSTRACT

This paper studies the applicability of genetic algorithms and imaging to measure deformations. Genetic algorithms are used to search for the strain field parameters of images from a uniaxial tensile test. The non-deformed image is artificially deformed according to the estimated strain field parameters, and the resulting image is compared with the true deformed image. The mean difference of intensities is used as a fitness function. Results are compared with a node-based strain measurement algorithm developed by Koljonen et al. The reference method slightly outperforms the genetic algorithm as for mean difference of intensities. The root-mean-square difference of the displacement fields is less than one pixel. However, with some improvements suggested in this paper the genetic algorithm based method may be worth considering, also in other similar applications: Surface matching instead of individual landmarks can be used in camera calibration and image registration. Search of deformation parameters by genetic algorithms could be applied in pattern recognition tasks e.g. in robotics, object tracking and remote sensing if the objects are subject to deformation. In addition, other transformation parameters could be simultaneously looked for.
**Keywords:** deformation, genetic algorithms, machine vision, search, tensile testing.

## 1 INTRODUCTION

Problems and applications, where some kind of (spatial) image transformations are applied or measured, are abundant in the broad field of image processing[1]. For instance in camera calibration[2, 3], transformation matrices to map image coordinates to world coordinate as well as to compensate for distortions are required. In image registration[4, 5], rigid or non-rigid body transformations are applied to 2D images or 3D voxel images in order to align (stitch) images visually or find corresponding points. Image transformations are used in image pattern recognition, too. In material engineering, measurements based on image are used to obtain deformation (strain) fields, which can also be regarded as a transformation. Actually, all these applications use same kinds of techniques (transformations and search methodologies); only the objectives may be somewhat different.

There are two principal methods to obtain a transformation mapping from acquired images: point matching and surface matching. The former uses templates that are obtained e.g. by training or syntactically. The templates are used as landmarks that are located from different images using some similarity metric. The relative positions of the located landmarks are subsequently used to determine the transformation mapping. The selection of the landmarks can be done regularly (e.g. a grid), by hand, or by some landmark selection algorithm.

Templates use local intensity patterns and they may be subject to transformations if the local patterns are assumed to be distorted. However, with complex transformations the search space grows rapidly, whence the matching process may slow down dramatically. Fortunately, with a proper similarity metric, there is no need for complex transformations; usually only translations and scales are varied.

The former method, surface matching, uses a larger surface and some similarity metrics. The surface can be regarded as a large template that is matched to another image with different transformations. Surface matching has a few potential advantages compared to template matching: it uses all the information in the selected image area, there is no need for landmark selection, only one transformation mapping is looked for. The latter feature may be beneficial as for computational complexity in situations, where complex transformation had to be individually searched for every template.

Intelligent Robots and Computer Vision XXV: Algorithms, Techniques, and Active Vision
edited by David P. Casasent, Ernest L. Hall, Juha Röning
Proc. of SPIE Vol. 6764, 67640O, (2007) · 0277-786X/07/$18 · doi: 10.1117/12.751725

There are a lot of studies of measuring strains of materials using imaging (Koljonen et al.[7] and the references therein). Some commercial products also exist for 3D strain measurements[6]. Usually the strain measurement algorithms use templates and regular grids. We have also studied and developed such an algorithm[7], and it is used as a reference method in this study. As far as we know, there is no surface matching based strain algorithm reported in the literature. We wanted to develop one, because surface matching seems to be common and efficient e.g. in medical image registration.

Genetic algorithms[8] (GAs) is an optimization method inspired by evolution. In the context of computer vision, genetic algorithms have been applied e.g. to the search of internal and external camera parameters[9], which is a nonlinear, multimodal optimization problem. Moreover, genetic algorithms have been applied to optimize the placement of cameras[10] and structural light vision[11, 12] with respect to the accuracy of 3D measurements and object reconstruction. In[13] a hybrid GA was used to determine the camera displacement using the background texture.

Genetic algorithms have also been extensively used in materials technology. A. Mahajan et al. have used GA and other soft computing methods to improve displacement and strain measurements of spray painted specimen. They tried to find an optimal subset of features and templates to be used in pattern localization. Moreover, their cost function included other component in addition to correlation. For more literature, the reader should refer to a comprehensive bibliography of genetic algorithms in materials science and engineering[14] with ca. 500 references.

This study uses B-splines in displacement field and intensity interpolation. B-splines and genetic algorithms have been applied e.g. to design antenna arrays[15] and in font outline description learning[16]. Intrinsic camera parameters have been interpolated by B-splines when varying focal length in[17]. The reader is kindly referred e.g. to refs. [15–17] for mathematical details and properties of cubic B-splines.

The surface matching based strain algorithm introduced in this paper stretches the image taken from a non-deformed test object using a deformation model consisting of 2D B-splines. A genetic algorithm is used to search the control point values that correspond to the deformation field of the deformed object that is imaged with the same camera parameters than the non-deformed one (Section 3). The results of the GA surface matching methods are compared with the ones obtained by our template matching algorithm, which is introduced in Section 2. The novel method gives promising results, but the computational complexity is still a limiting factor. However, we believe that the improvements suggested at the end of this paper (Section 4) will significantly reduce the computation time. The suggestions may also have more general use in the broad field of applications of genetic algorithms.

## 2 THE TEMPLATE BASED REFERENCE METHOD

The principle of template based strain measurements is to track homologous points of material using images taken before and after deformation. The strains are calculated from the relative movements of these points. Only the spatial derivatives of optical distortions of the images have an influence on the strain measurement results; usually camera calibration can be omitted. Now the algorithm is introduced in short.

Points on a regular rectangular grid are selected from the non-deformed image. Let $\mathbf{c}_{0,0} = (x_{0,0}, y_{0,0})$ and $\mathbf{c}_{I,J} = (x_{I,J}, y_{I,J})$ be the locations of the first and last nodes of the grid, respectively. The distance between the adjacent nodes in x direction is thus:

$$\Delta x = \left(x_{I,j} - x_{0,j}\right) / I \,. \tag{1}$$

Templates called Zones-Of-Interest (ZOI) are sampled around the grid points. The templates are used to localize the points to be tracked in the images taken from a test piece after deformation.

The localization of grid points is carried out by translating the ZOIs over a Region of Interest (ROI) and comparing the overlapping pixels using a similarity metric. Now correlation coefficients are used. Sub-pixel accuracy is achieved by correlation sampling and least-mean-square parabola fitting. ROI is determined by linear prediction, i.e. the predicted position is extrapolated using the two previous locations of each grid point. For the first deformed image no prediction can be done.

The principal engineer strains transversally (x) and longitudinally (y) are computed from the grid coordinates. In this algorithm, four neighboring nodes are used so that the x and y strains can be evaluated at the same point. The x strain is (y strain is defined similarly):

$$\varepsilon_{x,i,j} = \frac{\left(x_{i+1,j}^{(k)} - x_{i,j}^{(k)}\right) + \left(x_{i+1,j+1}^{(k)} - x_{i,j+1}^{(k)}\right)}{2\left(x_{i+1,j}^{(l)} - x_{i,j}^{(l)}\right)}. \tag{2}$$

## 3 GENETIC ALGORITHM AND SURFACE MATCHING

The node based strain measurement algorithm introduced in Section 2 and the surface matching method seems similar for the first glimpse, because the cubic B-spline based surface fitting introduced in this section also uses control points that almost correspond to the nodes of the reference method. However, there is a fundamental difference: while the node based algorithm uses only local intensity information, the surface based method uses global information, not only in the very neighborhood of a cubic B-spline control points but also beyond the adjacent nodes due to the continuity constraints of values and the first two derivatives of the cubic B-splines.

### 3.1 Strain field coding, image morphing, and fitness function

Genetic algorithms consist of a few fundamental parts: problem coding, fitness function, and genetic operators. Problem coding i.e. genotype-phenotype mapping determines, how the chromosomes (genotype) are interpreted as a trial (phenotype). A trial is a solution candidate. Genetic algorithms also have a population i.e. a pool of solution candidates. Fitness function (cost function, objective function) is used to evaluate the trials. Genetic operators are e.g. initialization, selection, crossover, and mutation. They are used to reproduce new trials, either totally or partly randomly using the existing trials as parents.

The strain field is coded as displacements of the B-spline control points with respect to their positions in the image without deformation. The control points form a uniform grid on the non-deformed image, similar to the grid in the template matching strain measurement algorithm introduced in Section 2. The displacements are coded using floating point genotype presentation. Note that there is a unique mapping between strains and displacements. The horizontal (x) and vertical (y) displacements are given in separate $(I+1)\times(J+1)$ matrices $\mathbf{D}_x$, $\mathbf{D}_y$.

It was presumed to obtain an advantage in the reproduction phase of the genetic algorithm when using displacements instead of strains in the genotype coding. With displacements, the influence radius of a single small control point translation is rather small, whereas if strains were used, the influence would be almost global. The same hold for crossover operations, too. Actually, displacement coding gives a good opportunity to utilize a building block approach.

The genotype (control points) is mapped into phenotype (stretched image) in five steps that were implemented using Matlab functions. The following procedure is also common in image morphing:

1. Interpolate displacements by cubic B-splines to obtain x and y displacements for every pixel.
2. Translate pixels of the non-deformed image $\mathbf{M}_0$ according to the interpolated displacements matrices, i.e. the intensities of the non-deformed image remain unchanged but their locations change. The locations are represented as floating point numbers and they are not rounded. Thus translation results in an image, whose pixels locate on a non-uniform grid.
3. Interpolate intensities of the translated pixels using cubic B-splines.
4. Sample intensities using a regular grid with the sampling interval that the non-deformed image had.
5. Round intensities e.g. to 8 bits. The resulting image is an artificially deformed image (ADI) $\mathbf{M}_{\mathrm{ADI}}$.

The artificially deformed image is evaluated using a cost function that compares it to a true deformed image $\mathbf{M}_k$, whose deformation field is to be estimated. If the artificially stretched image is equal to the true one, cost should be zero. Now the fitness function is the *mean absolute difference of intensities* of $\mathbf{M}_{\mathrm{ADI}}$ and the $\mathbf{M}_k$.

$$f = \sum_{m=1}^{M}\sum_{n=1}^{N}\left|\mathbf{M}_{\mathrm{ADI}}(m,n) - \mathbf{M}_{k}(m,n)\right| / MN \cdot \tag{3}$$

The objective of the genetic algorithm is to minimize the fitness function.

### 3.2 Genetic operators, internal parameters

The initialization step was considered important when applying genetic algorithms to deformation parameter search, because by smart initialization the search process can be speeded up significantly. Smart initialization was implemented using rough estimates of strains instead of random displacements. Because the strain field is many times somewhat continues (at least in the tensile test image series that were used), a rough estimate (obtained e.g. by visual comparison of images) of the overall strain is given to the initialization operator. Additionally, the displacement of the (0, 0) control point is estimated. The population of trials is initialized by randomly varying the given strain and displacement estimates. The initialized strains are converted to displacements, starting from the (0, 0) control point (top-left corner).

The population size is denoted by $n$ and the maximum number of trial by $N$. Parent for crossover and mutation are selected randomly from the entire population, but the fittest chromosomes are set more probable by drawing the parent index from a skewed distribution. The indices of the first and second parents are obtained by multiplying three and two random instances from a uniform [0, 1] distribution, respectively. The resulting index selection frequencies are shown in Figure 1.

Two operators are used in reproduction: uniform crossover of control points and Gaussian mutation of the control points. The probabilities of crossover and mutation as well as the mutation intensity (standard deviation) are changed during the optimization: crossover probability and mutation intensity are decreased while mutation probability is increased. Specifically, the probability to apply only mutation is increased linearly during the optimization from 0 to 80%, while the probability for plain crossover is decreased from 80 to 0%. Thus the probability to apply crossover and mutation subsequently and respectively is 20%. Mutation intensity i.e. the standard deviation of the Gaussian mutation was decreased from 4 to 0.2 pixels.

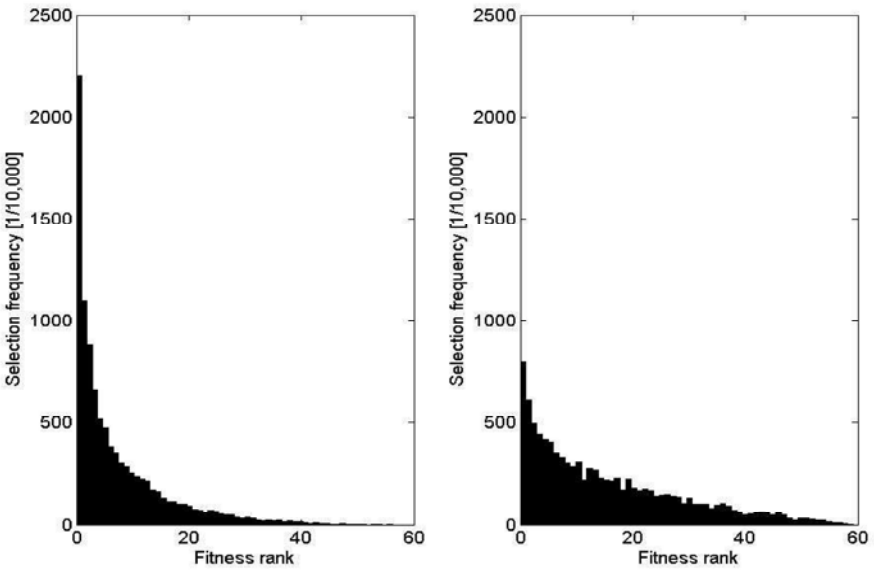



Figure 1. Selection frequencies out of 10,000 instances of different chromosomes for the first parent (left) and second parent (right) when $N = 60$. The fittest chromosomes have a greater probability to become a parent.

The reason for dynamic operator probabilities and mutation intensity is the assumption that crossover is efficient to combine good building blocks of trials i.e. one parent has some control points nearly right while the other one has some others. Hence, it is probable that the offspring has more good building blocks than its parents. Mutation is though required to add variation in the beginning and to fine tune the control points at the end.

## 4 EXPERIMENTS AND RESULTS

A spray-painted random speckle uniaxial tensile test 'dog bone' was stretched and imaged during deformation. The first image and an image with an overall longitudinal strain of approximately 1% were used in the experiments.

The genetic algorithm was run with three different parameters setup, three times with each setup. The first set of runs applied pure random search, where only the initialization operator was in use but the best trial so far was spared. In the second set of runs, population size was 60, and in the third 180. All other parameters were kept unaltered. The region of interest was $180 \times 400$ pixels and a grid of $4 \times 8$ control points was used. The total number of trials $N = 6000$. Figure 2 shows image samples from an optimization run.

### 4.1 Optimization efficiency

The three setups of genetic algorithm were compared as for optimization efficiency by studying the fitness development during the optimization runs. The mean values and standard deviations of the best fitness values are shown in Figure 3. The means suggest that each setup performs almost equally well, and random search sometimes even outperforms setups that utilize crossover and mutation operators. However, the variation figure indicates that crossover and mutation operator make the exploration more robust.

Adjustment of the internal GA parameters might improve the performance. On the other hand, genetic algorithms are claimed to be quite robust with respect to parameter selection. Improvements to the genetic operators could be more

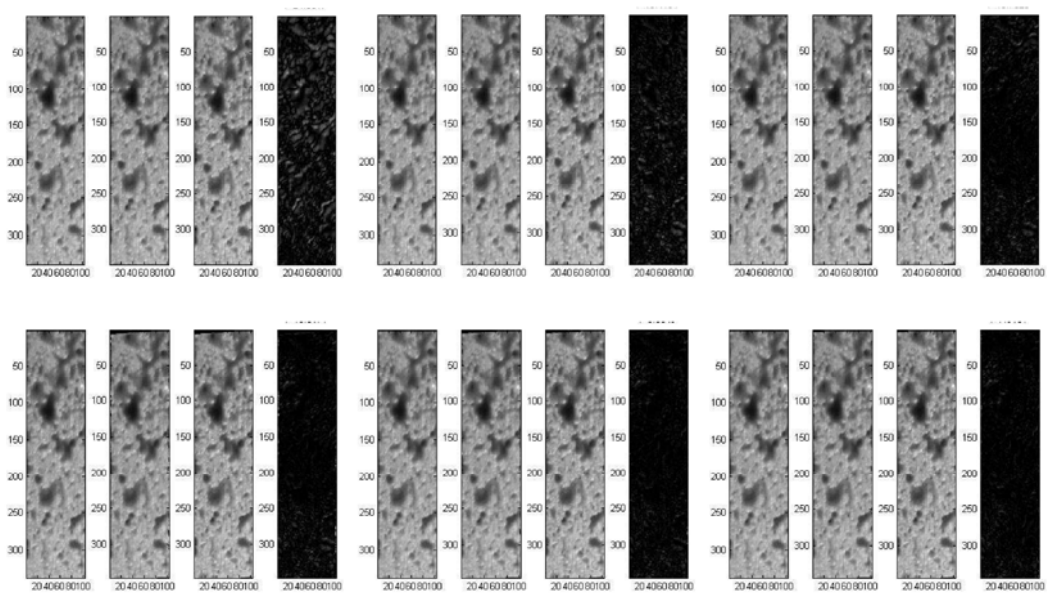


Figure 2. Samples from an optimization run. The four sub-images of each size image are: original image, artificially deformed image ($\mathbf{M}_{\mathrm{ADI}}$), true deformed image ($\mathbf{M}_k$), difference image of $\mathbf{M}_{\mathrm{ADI}}$ and $\mathbf{M}_k$. The fitness values of the images are: 24, 15, 11, 10, 9, and 8, left to right, top to down. For comparison, when all displacements were set to zero, fitness was 35.

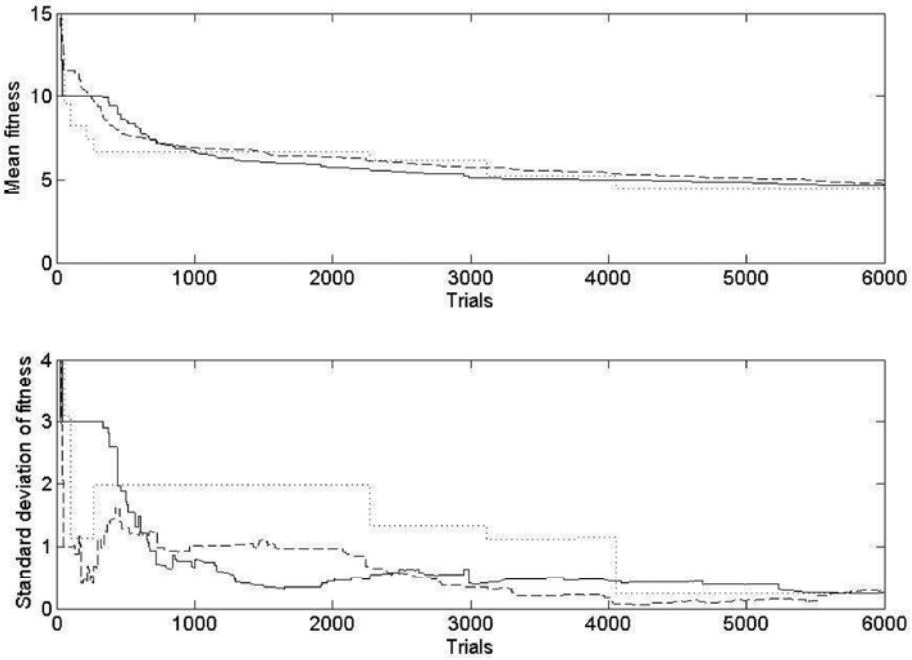



Figure 3. Development of the fitness of the best trial so far. Upper: mean fitness of three runs. Lower: standard deviation of three runs. Three different GA setups: random search (·····), $n = 60$ (----), and $n = 180$ (—).

fruitful. Actually, the reason for the good performance of the random search might be in the smart initialization operator, which is the sole genetic operator in the random search.

Initialization could be further improved e.g. by using coarse template based search for the control points. Another coarse-to-fine approach could such that control points were dynamically added between the existing ones when the fitness improvement saturates. Hence the spatial measurement resolution would be incrementally increased.

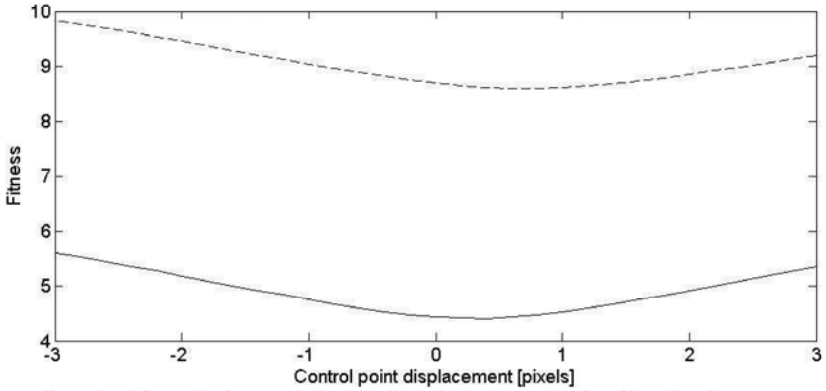



Figure 4. Two one-dimensional fitness landscapes obtained by displacing one control point of in y direction.

A hybrid genetic algorithm that utilized local hill climbing exploration could be beneficial as for complexity reduction. By studying fitness landscapes the usefulness of local optimization methods can be evaluated. Figure 4 shows two one-dimensional fitness landscapes. Each landscape was obtained by moving one control point of a trial one-dimensionally while keeping the other control points unchanged. The local fitness landscapes around the optimum suggest that hill climbing and even curve fitting could be efficient and robust. If local search and curve fitting were solely used, the approach resembles template based strain measurement algorithms with the distinction that global intensities are used. Nevertheless, the one-dimensional view does not expose if there are interactions that shift the local optima of single control points. In conjunction with genetic algorithms, hill climbing could be used e.g. as an intelligent mutation operator.

### 4.2 Comparison to the reference method

The template based method introduced in Section 2 was used to compute reference displacements for comparison. The initial control point positions were used as the initial node grid positions. After obtaining the displacements of the nodes on the deformed image the displacements were also interpolated with B-splines. When the node displacements of the reference method were used to evaluate the fitness function (eq. 3), the resulting absolute mean error of intensities was 4.1, which is 0.3 units better than the best fitness achieved by GA.

The displacements were compared pixel by pixel and a root-mean-squared (rms) value of the differences was computed. The resulting rms differences were computed after every 60$^{th}$ trial evaluation for the best trial obtained so far. The results for one random search run, one $n = 60$ run, and one $n = 180$ run are shown in Figure 5. The results indicate that there is a difference of order a half pixel between the displacements of the two methods (GA and reference). The reference method can be considered to produce better results because its fitness value (eq. 3) was lower. The measurement accuracy can be improved, at least to some extend, by adding control points. On the other hand, the more control points there are the longer the computation time becomes.

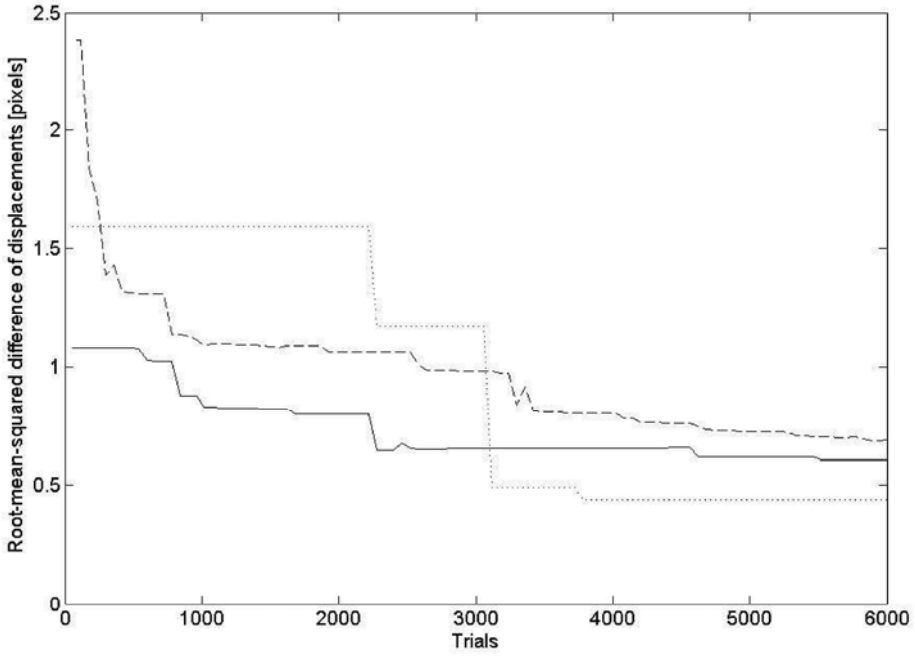



Figure 5. Root-mean-squared difference between displacements computed by genetic algorithms and the reference method. Three GA setups: random search ($n = 6000$) (·····), $n = 60$ (----), and $n = 180$ (—).

Table 1. Summary of best results of each GA setup: the best (lowest) fitness achieved in three runs of 6000 trials , the corresponding difference (in pixels) of the displacements obtained by GA and the reference method, and correlation between fitness and displacement differences.

| GA setup: | $n = 6000$ (random search) | $n = 60$ | $n = 180$ |
|---|---|---|---|
| Lowest fitness | 4.25 | 4.43 | 4.38 |
| Lowest rms difference (pixels) | 0.44 | 0.69 | 0.61 |
| Correlation between fitness values and rms differences of GA and reference method. | 1.000 | 0.994 | 0.994 |

If the root-mean-squared differences of the displacements between the two methods are compared to the corresponding fitness values, the quality of the fitness function can be estimated. Pearson's correlation coefficients for random search and GA runs with different GA setups are given in Table 1, together with some other statistics. The conclusion is that the fitness function gives good estimates of the estimation accuracy of the true displacements.

## 5 CONCLUSIONS AND FUTURE

A surface matching based strain measurement algorithm was suggested as an alternative to the template based one. Displacements of B-spline control points are optimized using genetic algorithms. A non-deformed image is artificially stretched and compared to a true deformed image. Mean absolute difference of intensities is used as the fitness value. B-splines are used to obtain a pixel-wise displacement field and after pixel displacements the artificially deformed image is constructed by intensity interpolation, re-sampling, and quantification.

Results show that the surface based method gives similar displacement fields than the template based method that was used as a reference. Smart initialization, control point crossover, and Gaussian mutation were found efficient and robust in exploration, but enhancement of the computational efficiency needs attention.

A potential idea might be to compute spatially local fitness function components, i.e. the difference image could be divided into blocks, whose fitness could be individually estimated. This *building block fitness function* could be subsequently used to control the crossover and mutation operators. In crossover, good building blocks could be combined from several chromosomes, and the displacement genotype coding and the properties of B-splines should guarantee that the resulting chromosome can efficiently utilize these inherited good building blocks without undesired negative interaction on the fitness. In mutation, bad building blocks could be subject to more variations to explore new areas of the search space. A hybrid genetic algorithm that utilized local hill climbing and curve fitting also looks very promising, because the local fitness landscapes are smooth.

## ACKNOWLEDGMENTS

Finnish Funding Agency for Technology and Innovation (TEKES) and the industrial partners of the research project *Process Development for Incremental Sheet Forming* have supported this research. J. Koljonen is grateful for the Finnish Cultural Foundation for the grant. Raimo Ruoppa is acknowledged for the reference material.

## REFERENCES

1. M. Sonka, V. Hlavac, R. Boyle, *Image Processing, Analysis, and Machine Vision*, second edition, Brooks/Cole Publishing, USA, 1999.
2. O. Faugeras, *Three-Dimensional Computer Vision – A Geometric Viewpoint*, The MIT Press, Cambridge, 1993.
3. R.Y. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses," *Robotics and Automation*, **3**(4), pp. 323-344, 1987.
4. J.V. Hajnal, D.L.G. Hill, D.J. Hawkes (eds.), *Medical Image Registration*, CRC Press, USA, 2001.
5. Z. Wang, P.M. Sharkey, "Toward accurate and automatic morphing," in *Intelligent Robots and Computer Vision XXIII: Algorithms, Techniques, and Active Vision*, Proc. of SPIE Vol. 6006, Boston, USA, 2005.
6. Vialux GmbH. URL: http://www.vialux.de/.

7. J. Koljonen, O. Kanniainen, J.T. Alander, "An implicit validation approach for digital image correlation based strain measurements," to appear in *The IEEE Region 8 Eurocon 2007 Int. Conf.*, Warsaw (Poland), Sept. 9-12, 2007.
8. S. Forrest, "Genetic algorithms: principles of natural selection applied to computation," *Science*, **261**(5123): 872-878, 1993.
9. M. Roberts, A.J. Naftel, "A Genetic algorithm approach to camera calibration in 3D machine vision," in *IEE Colloquium on Genetic Algorithms in Image Processing and Vision*, pp. 12/1-12/5, 1994.
10. G. Olague, R. Mohr, "Optimal camera placement for accurate reconstruction," *Patter Recognition*, **35**: 927-944, 2002.
11. T. Mantere, J.T. Alander, "Testing a structural light vision software by genetic algorithms – estimating the worst case behavior of volume measurement," in *Intelligent Robots and Computer Vision XX: Algorithms, Techniques, and Active Vision*, Proc. of SPIE Vol. 4572, Bellingham, USA, 2002.
12. T. Mantere, J.T. Alander, "Developing and testing a structural light vision software by co-evolutionary genetic algorithm," in *Proc. of the Second ASERC Workshop on Quantitative and Soft Computing based Software Engineering,* pp. 31-37, Banff, Canada, 2002.
13. S.A. Seyedin, H. Rezayi, "Finding the camera displacement in a 3D measurement system through matching the background texture of stereo images using hybrid GA," in *Proceedings of IEEE Int. Conf. on Information and Communication Technologies: From Theory to Applications*, pp. 361-362 2004.
14. J.T. Alander, *An Indexed Bibliography in Materials Science and Engineering*, University of Vaasa. URL: ftp://ftp.uwasa.fi/cs/report94-1/gaMSEbib.pdf.
15. W.-C. Lue, F. Hsu, "Use of B-spline curves and genetic algorithms to reduce the sidelobe level in array-patterns," *Microwave and Optical Technology Letters*, **38**(4): 308-311, 2003.
16. M. Sarfraz, S.A. Raza, "Capturing outline of fonts using genetic algorithm and splines," in *Proc. of IEEE Int. Conf. on Information Visualization*, pp. 738-743, London, UK, 2001.
17. S. Ying, G.W. Boon, "Camera self-calibration from video sequences with changing focal length," in *Proc. of IEEE Int. Conf. on Image Processing*, Vol. 2, pp. 176-180, Chicago, USA, 1998.

# PARAMETER OPTIMIZATION OF NUMERICAL METHODS USING ACCELERATED ESTIMATION OF COST FUNCTION: A CASE STUDY

*Janne Koljonen, Timo Mantere, Jarmo T. Alander*

Department of Electrical Engineering and Automation, University of Vaasa, P.O. Box 700, FIN-65101, Vaasa, Finland. E-mail: *Firstname.Lastname@uwasa.fi.*

## ABSTRACT

Problem solving algorithms and numerical methods typically include numerous control parameters. Their selection and tuning is usually done by experience or by trial-and-error, which is time consuming and hazardous. An automatic parameter search method introduced in this paper is easy to use, does not require experience and thus enables fast employment of the method. Furthermore, it guarantees that at least a suboptimal set of parameters is used regardless of the peculiarities of the task in hand. In order to accelerate the evaluation of the parameter set a simplified version of the algorithm is run and a cost function value computed. A correlation based optical strain measurement algorithm is used as a case study. Different evolutionary optimization algorithms are compared and correlations between accelerated and non-accelerated cost functions are studied.

**Keywords:** evolutionary algorithms, image-based measurements, parameter optimization.

## 1. INTRODUCTION

Problem solving algorithms and numerical methods typically include numerous or at least several control parameters – numerical coefficients, sub-method selections, etc. In addition to the high dimensionality of the parameter space, the effects and interacts of the parameters may be unknown and the execution of the algorithm may be time-intensive. Such methods are e.g. (a few to mention): genetic algorithms (GA) [1], Independent-Component-Analysis (ICA) [2], training of Artificial Neural Networks (ANN) [3, 4], computer vision algorithms, and classification algorithms.

The selection and tuning of these is usually done by experience or by trial-and-error. An experienced user knows immediately or by only a few tests a proper parameter combination. However, mistakes are possible. The trial-and-error approach is sometimes the source of experience for a novice, but on the other hand it may be time consuming, frustrating, and does not necessarily guarantee an optimal result.

If similar numerical tasks are solved repeatedly, the optimization of the algorithm parameters is beneficial, although the optimization may require numerous runs of the numerical algorithm itself. For example, studies on the internal parameters of genetic algorithms have been reported in many research papers [5–7]. Some general conclusions of the effects of different GA parameters on the algorithm performance can be drawn, but again at least moderate expertise is required to be able to apply them efficiently.

In general, we suppose that the optimal set of parameters of an algorithm is different when applied to different tasks. This implies that an optimal performance obtained only by re-optimization of the parameters, which is unfortunately time consuming. Nevertheless, in many cases an optimum found for one case can be used as an optimization seed for another one, thus accelerating parameter search. This paper introduces also another way to accelerate parameter optimization: a simplified version of the algorithm is used to evaluate a cost function. The idea to replace a complex cost function by a simpler substitute has been previously suggested at least in [8].

In this paper, a digital image correlation based optical strain measurement algorithm is used as a case study to test the accelerated parameter search. Different evolutionary optimization algorithms are compared and correlations between accelerated and non-accelerated cost functions are studied.

## 2. TEST CASE ALGORITHM

A strain measurement algorithm based on digital image correlation is used as a test case to develop and test the parameter tuning method proposed in this paper. The algorithm is introduced in [9]. An adaptive window size control algorithm is in turn introduced in [10]. Next the algorithm and its tunable parameters are summarized. Some

Reprinted, with permission of the University of Oulu, from [*Proceedings of the Finnish Signal Processing Symposium (Finsig 2007).* Parameter optimization of numerical methods using accelerated estimation of cost function: a case study. Koljonen, J., Mantere, T. & Alander, J. T.]

**Table 1.** Parameters, their explanations and ranges. (ROI=Region-Of-Interest.)

| Parameter | Explanation | Range |
|---|---|---|
| $Z_{xy}$ | Template size. | {9, 11, ..., 37} |
| $Z_{samp}$ | Template growth in sub-pixel correlation sampling. | {0, 2, ..., 28} |
| $Z_{inc}$ | Template growth in dynamic window size control. | {0, 2, ..., 16} |
| $R_x$ | Transversal ROI size. | 10 |
| $R_{y-}$ | Longitudinal ROI size opposite to the optical flow. | 30 |
| $R_{y+}$ | Longitudinal ROI size along the optical flow. | 40 |
| $R_{inc}$ | ROI growth in dynamic window size control. | {0, 1, ..., 4} |
| $R_{samp}$ | Size of sub-pixel correlation sampling neighborhood. | {3, 5, ..., 11} |
| *max_repeats* | Maximum number of repeats in dynamic window size control. | {0, 1, ..., 6} |
| $STD_{ZOI}Thres$ | Minimum requirement for standard deviation of template intensities. | [0, 20] |
| $DG_{ZOI}Thres$ | Minimum requirement for discrimination gap. | [0, 1] |
| $\Delta k$ | Number of images between template re-sampling. | {1, 2, ..., 40} |

new improvements are also added. All tunable parameters and their ranges are summarized in Table 1.

### 2.1. Basic strain measurement algorithm

Digital image correlation is a full-field method to measure strains of materials. The principle is to track fiducial landmarks of a sequence of images taken during deformation. Similar tasks also appear in object tracking [11], stereo matching [12], and image registration [13]. The strains are calculated from the displacements of these points. A regular $(J+1)\times(J+1)$ grid of fiducial point is selected from the non-deformed image (denoted by $k_0$) and templates of $Z_{xy}\times Z_{xy}$ pixels are sampled around the grid points.

The grid points are localized on the images taken from a deformed object using digital image correlation with sub-pixel localization based on correlation sampling and least-squares fitting of a parabola. Sub-pixel interpolation is done using an $R_{samp}\times R_{samp}$ neighborhood around the correlation maximum. To achieve better accuracy without increasing the localization time significantly a larger template of $(Z_{xy}+Z_{samp})\times(Z_{xy}+Z_{samp})$ pixel is used in the correlation surface sampling.

The Region-Of-Interest (ROI) of each point is determined by linear optical flow prediction. The dimensions of ROI are denoted by $R_x$, $R_{y-}$, and $R_{y+}$, in positive and negative transversal directions, opposite the principal optical flow, and in the direction of the principal optical flow, respectively.

Four neighboring points are used to estimate the transversal (x) and longitudinal (y) strains at the same location. Now the relative change of thickness (z strain) can be computed by assuming constant volume during deformation. The z strain is:

$$\varepsilon_z = \frac{1}{(\varepsilon_x - 1)(\varepsilon_y + 1)} - 1 \quad (1)$$

Because the target object is deformed but the templates are not artificially deformed before correlation matching, the fiducial points become gradually undetectable; the localization errors and the probability of a total mismatch increases. Therefore, templates are re-sampled after every $\Delta k$ images at the beginning, and after every image, when the object is necking (from images $k_{neck}$ to $k_N$), in which case local deformations are great. The strain increment corresponding to the re-sampling interval depends on the imaging interval and the deformation speed.

### 2.2. Adaptive window size control

Template size is a key parameter in pattern matching. The selection of a proper window size is a matter of optimization of accuracy e.g. in stereo vision [14–15], and sometimes balancing between accuracy and computational complexity as in strain measurements [9]. Compromises can be avoided using dynamic template size control.

In ref. [10], different confidence measures to be utilized in adaptive template size control are compared using pareto optimization [16] of strain measurement accuracy and complexity. A confidence measure based on the standard deviation of the template intensity values ($STD_{ZOI}$) was found efficient. Another promising one is based on the distance between the two first maxima of the correlation surface ($DG$ = Discrimination Gap).

In the strain measurement algorithm used in this study, the following cascade control algorithm is used:

```
Procedure windowSizeControl
compute STD_ZOI
i←0;
while i<max_repeats and STD_ZOI < STD_ZOIThres
   Z_xy←Z_xy+Z_inc;
   compute STD_ZOI
   i←i+1;
end while;
do pattern matching and compute DG
while i<max_repeats and DG < DG_ZOIThres
   Z_xy←Z_xy+Z_inc;
   R_x←R_x+R_inc;
   R_y+←R_y+R_inc;
   R_y-← R_y-+R_inc;
   do pattern matching and compute DG
   i←i+1;
end while;
end procedure;
```

The first while loop increases the template size until the standard deviation of the template intensities exceeds the given threshold. As no pattern matching is required, the first control loop is relatively quick to proceed.

The second loop does the correlation localization and sub-pixel interpolation and computes the discrimination gap that is compared to a threshold value. Because a low *DG* value may be due to either a too small template or too small ROI – so that the pattern is outside ROI due to false prediction of optical flow –ROI is also increased together with the template.

## 3. ACCELERATED ALGORITHM AND COST FUNCTION

The objective of this study is to test a method that optimizes the parameters of the strain measurement algorithm described in the previous section. The optimization is done without actually running the full algorithm but an accelerated version of it. The accuracy and computation time of the simplified algorithm are measured and used as components in the cost function that is in turn used to estimate the corresponding performance of the full algorithm that uses the same parameter setup. The underlying presumption is that there is a perfect or at least strong positive rank order correlation between the cost function values computed from the simplified and full version of the algorithm, i.e. optimizing one optimizes the other, too.

### 3.1. Acceleration methods

The methods of strain measurement acceleration applied in this study are: reduction of grid nodes and usage of only $k_{acc}$ first images. The minimum number of grid nodes that still allows to calculate an implicit measurement error estimate is 3×2.

### 3.2. Cost function

Strains in images $k_0$ to $k_{acc}$ are measured using the reduced grid and the computation time $t$ is measured and the measurement error $\delta\varepsilon_z$ at $k_{acc}$ is estimated. The measurement error is estimated by the standard deviation of the z strains at a transversal line of grid [9]:

$$\delta\varepsilon_{z,\cdot,j} = \sqrt{\sum_{i=0}^{I-1}(\varepsilon_{y,i,j} - \sum_{i=0}^{I-1}\varepsilon_{y,i,j}/I)^2/(I-1)}\ . \qquad (2)$$

The objective of the optimization is to minimize both $t$ and $\delta\varepsilon_z$. One option is to use pareto optimization [17], where either one of the objectives, say $\delta\varepsilon_z$, is kept constant while the other one ($t$) is minimized. Repeating this with different values of $\delta\varepsilon_z$ generates a pareto front with optimal solutions, from which a proper one can be selected. Unfortunately, pareto optimization is computationally rather intensive.

Another approach is to transform the dual-objective optimization task into a single-objective task by combining the cost function components ($t$ and $\delta\varepsilon_z$) into a single cost function. The following general distance function is proposed:

$$f(\delta\varepsilon_z, t) = (\delta\varepsilon_z^p + at^p)^{1/p}, \qquad (3)$$

where $p>0$ determines the topography of the space, and $a$ the weight of the components. Coefficient $a$ can be determined by selecting a pair of 'corresponding', nominal values for $t$ and $\delta\varepsilon_z$, whence:

$$a = \delta\varepsilon_{z\,nom} / t_{nom}. \qquad (4)$$

The cost function measures the distance of the cost function component pair to the origin using $p$-norm. E.g: if $p$ = 2, Euclidian distance is used. Figure 1 shows arcs of equal costs with different values of $p$ and $a$.

Parameter $a$ is used to weigh of the cost function components and $p$, together with $a$, is used to select the preferred path of optimization. Namely, the higher value of $p$ the more probably a low cost value is achieved in the middle of the equal-cost arcs, whereas with a low value of $p$ either one of the components becomes minimized first, which one, depends on the value of $a$ and also on the initial guess, because the deepest descent of the cost function is in the perpendicular direction from the equi-cost arcs.

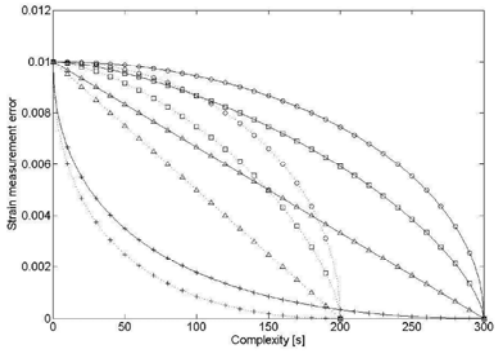



**Fig. 1.** Equi-cost arcs: a = 0.01/200s (dotted) and a = 0.01/300s (solid), and p = 0.5 (+), p = 1.0 (Δ), p = 1.5 (□), and p = 2.0 (o).

## 4. OPTIMIZATION METHODS AND EXPERIMENTS

Five different evolutionary optimization methods were tested and compared:

1 Random search,
2 mutation of a single trial,
3 population of five trials with:
  3.1 sole mutation: $prob_{mut\&\neg cross} = 1$,
  3.2 sole crossover: $prob_{\neg mut\&cross} = 1$,
  3.3 crossover and mutation: $prob_{mut\&\neg cross} = 0.33$, $prob_{\neg mut\&cross} = 0.33$, and $prob_{mut\&cross} = 0.33$.

Random trials are drawn from uniform distributions using the ranges given in Table 1. Mutations are applied to the phenotype presentations of the parameters. Mutation of the discrete variables follow a uniform distribution: $R_{samp}$, *max_repeats* from {-2, -1, 0, 1, 2}, $Z_{xy}$, $Z_{samp}$, $Z_{inc}$ from {-4, -2, 0, 2, 4}, $\Delta k$ from {-10, -9, …, 9, 10}, and $R_{inc}$ from {-4, -3, …, 3, 4}. Mutations of the real valued free parameters are done using normal distributions: $STD_{ZOI}Thres$ from $N(0, 3)$ and $DG_{ZOI}Thres$ from $N(0, 0.1)$. The results of mutations do not have to obey the ranges given in Table 1, but negative values are forced to zero.

Crossover also operates in the phenotype space. It crossbreeds two trials by selecting each parameter from either of the trials by equal probability.

Each of the five optimization method was tested by test runs of 40 trials. The test runs were repeated five times. The number of images used was $k_{acc} = 100$ out of 182 images. The distance function exponent $p = 1.5$ and the weight of cost function components $a = 0.01/300 \approx 33 \cdot 10^{-6}$.

In addition, the correlation between the accelerated and full cost functions was studied as follows: both the accelerated and the full cost functions were evaluated using same parameters of 20 random trials. This was repeated with $k_{acc} = \{50, 100, 150\}$, three different levels of acceleration.

## 5. RESULTS

This section first shows the correlations between accelerated and full cost functions. Then the optimization speed and reliability of the five different methods are compared. The optimization paths in the plane of cost function components are also studied.

### 5.2. Validity of the accelerated cost function

The full cost function values are plotted against the accelerated costs in Fig. 2. It seems that the variation of the most accelerated cost function values is clearly the smallest. Using only the first 50 images out of 182 does not probably expose the hazards concerning strain measurement accuracy. On the other hand, the dynamic template size control may cause unexpected variations to the execution time.

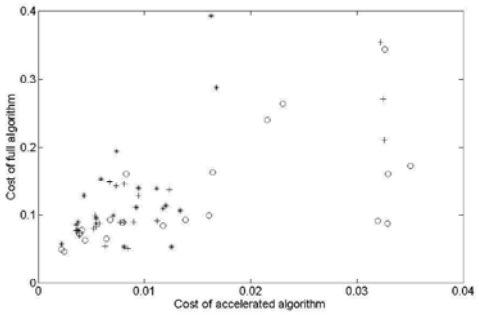



**Fig. 2.** Cost value pairs of 20 random trials with three levels of acceleration: $k_{acc} = 50$ (*), $k_{acc} = 150$ (+), and $k_{acc} = 150$ (o).

The Spearman's rank order correlation of the full and accelerated cost function values were calculated to obtain quantitative comparison. The rate of acceleration was computed by dividing the execution times of the full cost function by the corresponding execution times of the accelerated function and taking the mean of the ratios. The Spearman's correlation coefficients are plotted with respect to the acceleration ratios in Fig. 3. The results are consistent so that the greater the acceleration the lower the correlation.

### 5.3. Comparison of methods

The developments of the lowest cost during optimization are shown in Fig. 4. Each curve represents one of the optimization methods introduced in Section 4. Each curve shows the mean development of five test runs.

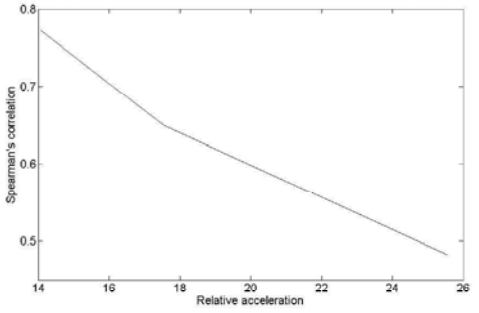



**Fig. 3.** Spearman's rank order correlation vs. acceleration ratio of the cost function.

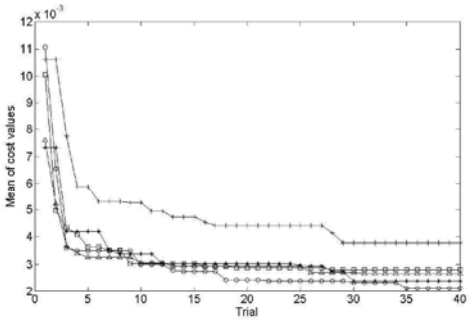



**Fig. 4.** Means of the best cost values against the number of trials evaluated. Five test runs with different operations (see Section 4): * = method 1, + = method 2, Δ = method 3.1, □ = method 3.2, and o = method 3.3.

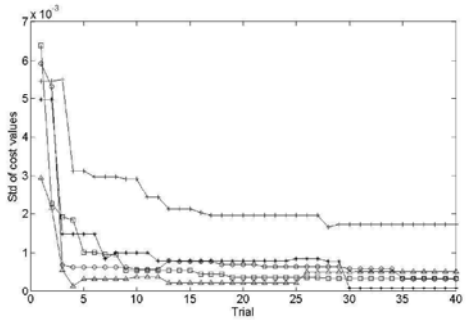



**Fig. 5.** Standard deviations of the best cost values vs. the number of trials evaluated. Same test runs and symbols as in Fig. 4.

Fig. 5 gives the standard deviations of the five test runs to each optimization method. The means indicate how fast each method converges in average, whereas standard deviation can be used as a measure of unreliability of the methods.

Figs. 4 and 5 indicate that random search finds (in average) quickly a somewhat good set of parameters, whereas sole mutation of a single trial converges slowly but nonetheless consistently. The efficiency of random search indicates that the cost function landscape is relatively flat (in contrast to the rugged or needle-in-a-haystack type landscapes).

In spite of the satisfying results of the random search, the more complicated evolutionary optimization strategies have some advantage over pure random search: Particularly population with mutation (method 3.1 introduced in Section 4) as well as with crossover and mutation (method 3.3) have as fast convergence as the random search and simultaneously significantly lower standard deviations than the random method.

As a conclusion, e.g. the use of crossover and mutation can guarantee a reliable suboptimal result with approximately 20 trials, when the acceleration ratio is about 17. Henceforth, the optimization takes totally only about 20/17 = 1.2 times a single execution of the full cost function. Moreover, the Spearman's correlation with the acceleration level in question is about 0.65.

The evolvements of the two cost function components are studied in Fig. 6. The results show that the distance function exponent $p = 1.5$ allows some flexibility to compromise between error and complexity in the search of optimum, but nonetheless focuses the trials to the same sink.

It is supposed that the higher the $p$, the more the evolvement path is focused. An analogy to focusing ($p > 1$) and diverging ($p < 1$) lenses can be considered.

## 6. CONCLUSION

A method to optimize the set of internal parameters of a digital image correlation based stain measurement algorithm was introduced. Due to high complexity of the algorithm an accelerated version of the algorithm was used to compute the cost function used in evolutionary optimization. The cost function consisted of two components: strain measurement error and execution time. The cost function components were combined into a scalar cost value by a general distance function with two tunable parameters: one for the mutual weight of the components, the other for the topology of the space.

The accelerated cost function values were compared to the costs computed with the full algorithm. The less acceleration was used the higher the Spearman's rank order correlation was. Five different setups of evolutionary optimization were compared. Even though pure random search converged quite fast to a satisfying result, an evolutionary algorithm with phenotype mutation and crossover

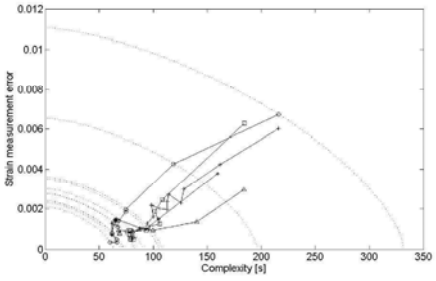



**Fig. 6.** Evolvement paths of the means of the best trials plotted in a plane spanned by the cost function components. Same test runs and symbols as in Fig. 4.

showed both fast and robust convergence.

In the future, the proposed accelerated optimization method of internal parameters should be applied to other numerical problems. Moreover, the effects of the parameters of the general distance function based cost function should be studied.

## 7. ACKNOWLEDGMENTS

Finnish Funding Agency for Technology and Innovation (TEKES) and the industrial partners of the research project *Process Development for Incremental Sheet Forming* have supported this research. J. Koljonen gratefully acknowledges the Finnish Cultural Foundation for the grant.

## 8. REFERENCES

[1] J. Holland, *Adaptation in Natural and Artificial Systems*, The MIT Press, Cambridge, MA, USA, 1992.

[2] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, John Wiley & Sons, 2001.

[3] W. McCulloch and W. Pitts, "A Logical Calculus of the Ideas Immanent in Nervous Activity," in *Bulletin of Mathematical Biophysics*, vol. 5, pp. 115–133, 1943.

[4] S. Haykin, *Neural Networks – A Comprehensive Foundation*. Second edition. Prentice-Hall, Inc., New Jersey, 1999.

[5] J.T. Alander, "On optimal population size of genetic algorithms", in *Proceedings of CompEuro 1992*, Haag, Holland, May 1992, pp. 65–70.

[6] J.T. Alander and T. Mantere, "Genetic algorithms in software testing - experiments with temporal target functions," In *Proceeding of the 6th International Conference on Soft Computing*, Brno, Czech Republic, Jun. 2000, pp. 9–14.

[7] J. Koljonen and J.T. Alander, "Effects of population size and relative elitism on optimization speed and reliability of genetic algorithms", in *Proceedings of the Ninth Scandinavian Conference on Artificial Intelligence* (SCAI 2006), Espoo, Finland, Oct. 2006, pp. 54–60.

[8] J.T. Alander, T. Mantere, and T. Pyylampi, "Digital halftoning optimization via genetic algorithms for ink jet machine," in *Developments in Computational mechanics with high performance computing*, CIVIL-COMP Press, Edinburgh, UK, 1999, pp. 211–216.

[9] J. Koljonen, O. Kanniainen, and J.T. Alander, "An implicit validation approach for digital image correlation based strain measurements," to appear in *Proceedings of the IEEE Region 8 Eurocon 2007 Conference*, Warsaw, Poland, Sept. 2007.

[10] J. Koljonen, O. Kanniainen, and J.T. Alander, "Dynamic template size control in digital image correlation based strain measurements," to appear in *Intelligent Robots and Computer Vision XXV: Algorithms, Techniques, and Active Vision*, Proceedings of SPIE, Optics East 2007, Boston, MA, USA, Sept. 2007.

[11] Y. Zhong, A.K. Jain, and M.-P. Dubuisson-Jolly, "Object tracking using deformable templates," *Pattern Analysis and Machine Intelligence*, vol. 22, no. 5, p.p. 544–549, May 2000.

[12] O. Faugeras, *Three-Dimensional Computer Vision – A Geometric Viewpoint*, The MIT Press, Cambridge, MA, USA, 1993.

[13] J.V. Hajnal, D.L.G. Hill, D.J. Hawkes (eds.), *Medical Image Registration*, CRC Press, USA, 2001.

[14] S.B. Steinman, B.A. Steinman, and R.P. Garzia, *Foundations of Binocular Vision: A Clinical perspective*, McGraw-Hill Medical, 2000.

[15] T. Kanade and M. Okutomi, "A stereo matching algorithm with an adaptive window: theory and experiment," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 16, no. 9, pp. 920-932, 1994.

[16] K.J. Arrow and G. Debreu (1954), "Existence of a Competitive Equilibrium for a Competitive Economy," *Econometrica*, vol. 22, no. 3, pp. 265–90, 1954.

[17] J. Horn, N. Nafpliotis, and D.E. Goldberg, "A Niched Pareto Genetic Algorithm for Multiobjective Optimization," in *Proceedings of the First IEEE Conference on Evolutionary Computation*, IEEE World Congress on Computational Intelligence, ICEC´94, Piscataway, NJ, USA, 1994, pp. 82–87.

International Deep Drawing Research Group
IDDRG 2008 International Conference
16-18 June 2008, Olofström, Sweden

# VALIDATION OF DIGITAL SPECKLE CORRELATION STRAIN MEASUREMENTS WITH EXTENSOMETER

**Janne Koljonen**[①], **Tuomas Katajarinne**[②], **Annette Lönnqvist**[②] **and Jarmo T. Alander**[①]

[①] Department of Electrical Engineering and Automation
University of Vaasa
P.O. Box 700, FI-65101, Vaasa, Finland
e-mail: Firstname.Lastname@uwasa.fi, web page: http://www.uwasa.fi

[②] Laboratory of Processing and Heat Treatment of Materials
Helsinki University of Technology
P.O. Box 6200, FI-405 31, TKK, Finland
e-mail: Firstname.Lastname@hut.fi, Web page: http://www.hut.fi

**Keywords:** Digital image correlation, Machine vision, Optical extensometer, Radial distortion, Strain, Tensile test, Validation

**ABSTRACT.** *Validation of a digital speckle correlation based strain measurement algorithm (optical extensometer) is studied. In addition, the effect of optical radial distortion is studied and a correction formula is derived. The validation set is obtained by making a series of uni-axial tensile tests, whose gold standards are obtained by a conventional mechanical extensometer. The radial distortion model is calibrated using this gold standard. In some test cases, the relative strain measurement error of the optical method, with respect to the mechanical one, was as low as 0.4 %. However, probably due to some nonlinearities with unclear origins the overall estimated relative strain measurement error was of order 10 %.*

## 1. INTRODUCTION

Optical strain measurements based on digital image correlation (DIG) have been studied and reported in several papers since the early 80's; see e.g. [1–8]. In this paper, a DIG based optical extensometer (or video extensometer) is tested using measurements of a conventional mechanical extensometer as the gold standard.

Benefits of DIG based strain measurement systems are at least the following: The measurement principle is non-contact without moving parts, which implies reliability and applicability to harsh environments. Strain measurements with good spatial and temporal resolution are obtained. The hardware can be constructed from standard components (cameras, PC, lights, etc.), which keeps the costs low. Moreover, the measurement procedure is simple: the sample can be prepared with a spray-painted random speckle, imaged during the load, and strains computed from the images, either offline or online.

The general challenges of machine vision systems apply also to DIG based strain measurements. Common error sources are related to lighting and contrast; e.g. shadows and other spatial lighting heterogeneities, ambient light, temporal modulation of light, reflections, and texture of the object are all harmful but difficult to control.

The sample preparation can also be regarded as a drawback, because no preparation would enable a more easily automated testing. Measurement systems based on images usually require calibration, which may be a demanding task if absolute accuracy is a crucial factor or if the measurement setups vary. In tensile testing, for instance, imaging geometrics (working distance, magnification) are sometimes changed according to the sample size, which may require recalibration of the camera. Furthermore, vibrations may change camera parameters and blur the images, too. However, validation is probably the most demanding task.

Validation of optical strain measurements is difficult, because a reliable reference method lacks. Error estimation in the image coordinate system can be done using, for example, figures of deviation [9] or artificial test images [4, 8]. These approaches are inadequate as they ignore the image forming process. Nonetheless, the implicit validation presented in [9] is usual to obtain online error margins for the strain measurements in the image coordinate system.

A common validation approach that takes account of the image forming process is to use mechanical stages [2, 7]. An x-y-stage enables to obtain images with an accurately known translation (displacements). Since displacement and strain fields can be converted to one another, testing of displacements have been regarded adequate. However, the method ignores (local) deformations that complicate the measurement of displacements. Moreover, lighting conditions, vibrations, and other ambient conditions are not equivalent to e.g. those during a tensile test. It is unclear how much these shortcomings affect the validation accuracy.

Mechanical extensometer has been tried to fix the problems related to strain validation by displacements. An obvious drawback in this approach is that the optical and mechanical extensometers cannot use the same points of the object as the mechanical one shades the specimen on the measurement site. Thus, if the deformation of the specimen is spatially heterogeneous, then the validation accuracy is reduced. This could be partly compensated by optically measuring strains both between points inside and outside the clamps of the extensometer.

Kanchanomai et al. [10] used an optical extensometer with a 512 × 480 pixel camera to measure strain during a fatigue test. Two pieces of black tape were used as artificial landmarks. Intensity interpolation was used to raise the spatial image resolution by a factor of ten. No lens distortion compensation was mentioned. The displacement measurement accuracy of a digital image measurement system was tested by comparing the displacement results with that of an extensometer. When the gauge length was 8 mm, the maximum difference was 2 μm at a displacement of 150 μm. Thus strain was 18,750 μstrain, strain error

250 µstrain, and relative strain error 1.3 %.

Dumoulin et al. [11] painted the sample with black and white random pattern. Stress-strain curves were obtained by a mechanical and optical extensometer. The equivalent stress–equivalent strain curve was used to simulate the tensile test with finite element (FEM) analysis. They concluded that the optical method and the FEM simulation gave identical results until the onset of transversal necking.

## 2. MATERIALS

### 2.1. Imaging hardware

The imaging hardware (Figure 1) consists of a 'dog-bone' specimen having a stochastic speckle coating, a remote-controlled digital SLR camera (Nikon D80), spotlights, and a personal computer. The imaging software consists of a camera remote-control program that enables scheduled image capture and transfer without any interfering contact to the imaging hardware.

Costs of the imaging hardware are low, because no special machine vision cameras are needed, unless high speed tensile tests are done. Instead, a digital SLR camera with good-quality optics is adequate to acquire high resolution images with a sampling interval of order 1–2 seconds, provided that a remote-control software is available. On the other hand, the strain analysis software sets no requirements for the imaging system as the image transfer is carried out by lossless compressed image files. Digital SLR cameras are much more inexpensive than machine vision cameras of the same resolution.

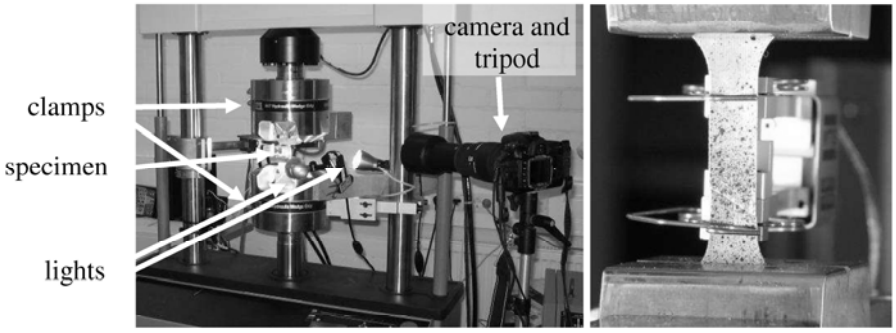



***Figure 1.*** *Experimental setup and hardware (left). Test object with a random spray-painted pattern and the mechanical extensometer (right).*

### 2.2. Strain measurement software

The images are analyzed by a Matlab application in order to estimate the whole-field strain distributions as a function of time. The algorithm has been previously presented in publications [9, 12, 13]. It is based on the principle of tracking landmarks of an originally rectangular grid. The landmarks are localized with sub-pixel accuracy using template matching and correlation interpolation. The strains are computed from the relative displacements of these landmarks.

The algorithm includes implicit error estimation [9], adaptive online template size control [12], and offline parameter optimization by a genetic algorithm [13]. It also has several other novel approaches, e.g. a dual template size coarse-to-fine search strategy, to increase the accuracy-complexity ratio.

The implicit error estimation is based on the deviation of several strain estimates along a transversal line. In adaptive template size control, the correlation window size is increased online if the positioning reliability is too low. Positioning reliability estimates are derived

from the statistics of the intensities of the template image and the correlation surface. The adaptive template size control has proved significant improvement in the accuracy-complexity ratio.

The coarse-to-fine search strategy has been implemented as follows:

1. Correlations are sampled with a rather small template to obtain a preliminary correlation peak. The spatial interval is as long as possible, typically 2–3 pixels, without significantly increasing the probability of mismatches.
2. Correlations are sampled around the preliminary peak, which is updated if necessary.
3. A larger template is used to update the correlations around the preliminary peak.
4. The updated samples from step 3 are used to interpolate the sub-pixel position of the landmark.

### 2.3. Radial distortion in strain measurements

Metric measurements based on image require accurate calibration of the internal and external parameters of the imaging hardware, as well as compensation of the optical distortions. Because engineering strain is measured as the change of length relative to the original length, units of length can be arbitrary. A natural approach is to use pixels and the image pixel coordinate system. Thus the determination of the transformation coefficients from pixels to meters is not needed.

Optical distortions in turn affect strain measurements. Properties of the optical materials, geometry of the lens (system), and assembly defects determine how the image is distorted. Distortions can be geometrical or they can be related to intensity. Geometrical distortions include: radial distortion, tangential distortion, linear distortion, chromatic aberration, spherical aberration, coma, and astigmatism. Intensity related distortions are in turn shading, focal aberration, and diffraction.

Suppose that the distortion is known (measured) as a function of image pixel coordinates $(u, v)$ both horizontally $\Delta u(u, v)$ and vertically $\Delta v(u, v)$. If the measured coordinates of a point are $(\tilde{u}, \tilde{v})$, the distortion corrected coordinates are obtained by:

$$(u,v) = (\tilde{u} + \Delta u(\tilde{u},\tilde{v}), \tilde{v} + \Delta v(\tilde{u},\tilde{v})). \tag{1}$$

To simplify the subsequent analysis the following presumptions are done (Figure 2): the test piece is parallel to the $v$-axis, the origin of the image coordinate system is on the optical axis (principal point), and distortion in $u$-direction is negligible (or assume $u = 0$ for all points).

Next suppose that two fiducial points are tracked during a tensile test. Before deformation their measured coordinates are $\tilde{v}_0$ and $\tilde{v}_1$, while after deformation they are $\tilde{v}_2$ and $\tilde{v}_3$, respectively. The longitudinal engineering strain of the corresponding distortion corrected points would be:

$$\varepsilon_y = \frac{\Delta l}{l_0} = \frac{(v_3 - v_2) - (v_1 - v_0)}{v_1 - v_0}. \tag{2}$$

Use eq. (1) to substitute $v$ with $\tilde{v}$. For simplicity, suppose that the points are selected symmetrically with respect to the principle point and only the latter points ($\tilde{v}_1$ and $\tilde{v}_3$) have moved, i.e. set $\tilde{v}_0 = -\tilde{v}_1$ and $\tilde{v}_2 = \tilde{v}_0 = -\tilde{v}_1$. Eq. (2) can now be rewritten as:

$$\varepsilon_y = \frac{\tilde{v}_3 - \tilde{v}_1 + \Delta v(0,\tilde{v}_3) - \Delta v(0,\tilde{v}_1)}{2\tilde{v}_1 + \Delta v(0,\tilde{v}_1) - \Delta v(0,-\tilde{v}_1)} \tag{3}$$

Furthermore, eq. (3) can be written as:

$$\varepsilon_y = \frac{\tilde{\varepsilon}_y}{1+(\Delta v(0,\tilde{v}_1)-\Delta v(0,-\tilde{v}_1))/2\tilde{v}_1} + \frac{\Delta v(0,\tilde{v}_3)-\Delta v(0,\tilde{v}_1)}{2\tilde{v}_1+\Delta v(0,\tilde{v}_1)-\Delta v(0,-\tilde{v}_1)}, \tag{4}$$

where $\tilde{\varepsilon}$ is *apparent strain*:

$$\tilde{\varepsilon}_y = (\tilde{v}_3 - \tilde{v}_1)/2\tilde{v}_1. \tag{5}$$

Suppose that the distortion is anti-symmetric i.e. $\Delta v(u, v) = -\Delta v(u, -v)$. Then eq. (4) becomes:

$$\varepsilon_y = \frac{\tilde{\varepsilon}_y}{1+\Delta v(0,\tilde{v}_1)/\tilde{v}_1} + \frac{\Delta v(0,\tilde{v}_3)-\Delta v(0,\tilde{v}_1)}{2\tilde{v}_1+2\Delta v(0,\tilde{v}_1)}. \tag{6}$$

A typical anti-symmetric distortion is radial distortion that is probably the most significant distortion. Radial distortion is usually modeled by even-order polynomials, such as:

$$\begin{aligned}\Delta u(\tilde{u},\tilde{v}) &= (\tilde{u}-u_0)(\kappa_1 r^2+\kappa_2 r^4+\kappa_3 r^6)\\ \Delta v(\tilde{u},\tilde{v}) &= (\tilde{v}-v_0)(\kappa_1 r^2+\kappa_2 r^4+\kappa_3 r^6)\end{aligned}, \tag{7}$$

where $(u_0, v_0)$ is the principal point of the image (intersection point of the optical axis), $\kappa_i$ are free model parameters that are obtained by calibration, and $r^2 = (\tilde{u}-u_0)^2 + (\tilde{v}-v_0)^2$ [14].

Now coordinate origin was set to the principal point, i.e. $(u_0, v_0) = (0, 0)$, and $\tilde{u} = 0$ implying $\Delta u = 0$. Let us also suppose, for simplicity, that $\kappa_3 = 0$. Then $\Delta v$ becomes:

$$\Delta v(\tilde{v}) = \tilde{v}(\kappa_1\tilde{v}^2+\kappa_2\tilde{v}^4) = \kappa_1\tilde{v}^3+\kappa_2\tilde{v}^5, \tag{8}$$

Substituting $\Delta v$ in eq. (6) by $\Delta v$ in eq. (8) gives:

$$\varepsilon_y(\tilde{v}_1,\tilde{v}_3) = \frac{\tilde{\varepsilon}_y}{1+\kappa_1\tilde{v}_1^2+\kappa_2\tilde{v}_1^4} + \frac{\kappa_1(\tilde{v}_3^3-\tilde{v}_1^3)+\kappa_2(\tilde{v}_3^5-\tilde{v}_1^5)}{2+2(\kappa_1\tilde{v}_1^2+\kappa_2\tilde{v}_1^4)}. \tag{9}$$

Eq. (9) shows how the apparent strain has to be corrected to obtain real strain in the presence of radial distortion and when the above mentioned presumptions apply. Correction includes a multiplicative part and an additive term. It can be seen that the correction depends on the original, apparent gauge length ($2\tilde{v}_1$) and the apparent change of length.

Figures 3 and 4 show the total additive strain corrections as a function of apparent strain and the original gauge length, respectively. The additive contributions of the multiplicative

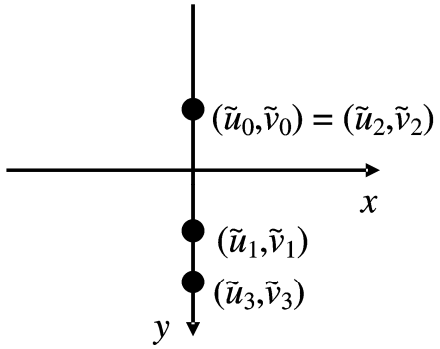



***Figure 2.*** *Positions of the fiducial points in the distortion analysis.*

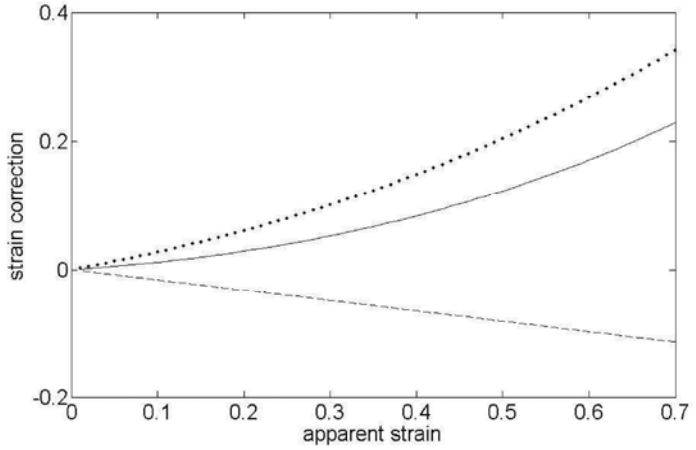



***Figure 3.*** *Total strain correction as a function of apparent strain according to* eq. (9) *(solid line), and its composites: multiplicative (dashed) and additive (dotted).* $\tilde{v}_1 = 877.5$ *pixels.*

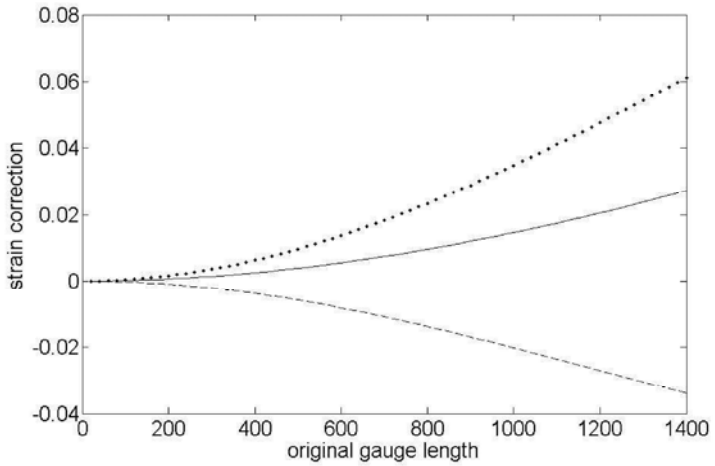



***Figure 4.*** *Total strain correction as a function of original apparent gauge length according to* eq. (9) *(solid line), and its composites: multiplicative (dashed) and additive (dotted).* $\tilde{\varepsilon} = 0.1$.

and the additive terms in eq. (9) are given, too. The distortion parameters in these examples are: $\kappa_1 = 2.3 \cdot 10^{-7}$ and $\kappa_2 = 1 \cdot 10^{-14}$.

### 2.4. Distortion calibration using extensometer

Usually the distortion parameters are obtained by camera calibration using a known scene. In this study, the parameter values $\kappa_1$ and $\kappa_2$ were obtained by fitting the measured apparent strains to the measured strain given by the mechanical extensometer. Hence, the measurements of the mechanical extensometer were regarded as the gold standard.

This approach to camera distortion calibration is practical, because the imaging setup is kept constant; camera is not moved and focal length is guaranteed to be constant. On the other hand, accuracy is probably not the best, because the gauge lengths of the mechanical and optical extensometers may vary. To compensate this shortcoming different apparent optical

gauge lengths were used. In general, this calibration procedure was used in order to introduce an alternative method, but the results should be validated in future.

Parameters $\kappa_1$ and $\kappa_2$ were obtained by multi-criteria optimization. First strain measurement errors (difference between results of mechanical and optical extensometers) as a function of time were computed for every tensile test. For some test cases (tensile tests) strains were measured using both inside-clamp and outside-clamp approach as described in Introduction. Means and standard deviations were calculated from these error time series.

The first criterion was that the grand mean of all error series was zero, i.e. the expectation value of error at a random time instant was zero. For each $\kappa_2$ this criterion can be fulfilled by one dimensional search in $\kappa_1$.

Another criterion was to minimize the mean of the standard deviations of the errors. This criterion was checked separately for the inside-clamp and outside-clamp measurements. Values for $\kappa_1$ and $\kappa_2$ were selected so that the outside-clamp mean standard deviation of errors was minimized, although the corresponding figure of merit for the inside-clamp errors was not minimized.

## 3. EXPERIMENTAL PROCEDURE

Square cones with different wall angles were produced by incremental sheet forming (ISF) [15, 16]. The aim was to investigate the formation of strain induced martensite and its effect on tensile behavior. When tensile test specimens were prepared from the square cones, the residual stresses and high deformation degree caused distortion to the flatness of the samples.

A 100 kN MTS 810.22 tensile testing machine was used for the testing. An extensometer with a gauge length of 25 mm, capable of 25 % elongation was used to obtain the gold standard measurements.

Cross head speed of 0.1 mm/min was used. A Nikon D80 digital camera was used for the photographing with a remote-controlled camera software for imaging. Working distance was 45 cm. The camera lens was in manual mode to prevent the camera from changing its setups.

The hydraulic jaws of the tensile testing machine caused some bending to the specimens upon attachment. This could clearly be seen from the strain-load curves as no load was detected before the sample had straightened.

Totally 13 tensile tests were imaged (test cases). The DIG based software in introduced in section 2.2 was used to measure the apparent strains of the test cases. The parameters of the strain measurement algorithm were optimized using the genetic algorithm described in [13]. The same optimized parameters were used for every test case.

The measurement sites were selected manually so that they are as close as possible to the measurement sites of the mechanical extensometer but so that the correlation windows do not overlap with the clamps of the extensometers.

## 4. RESULTS

Two samples of strain measurements as a function of time are given in Figure 5. Results show that the apparent strains are smaller than the real ones obtained by a mechanical extensometer. The samples are representative as for other results.

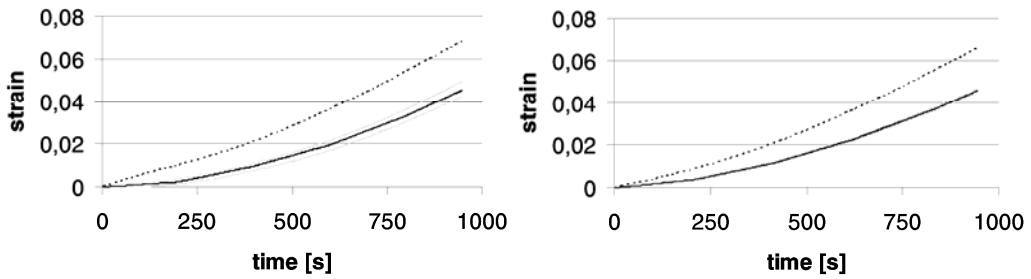



***Figure 5.*** *Strain vs. time from two tensile tests. Dotted line: strains measured by the mechanical extensometer. Black line: apparent strains. Gray lines: implicit confidence interval estimates.*

The calibration procedure described in Section 2.4 was followed and the following values for the distortions parameters were obtained: $\kappa_1 = 2.2975 \cdot 10^{-7}$ and $\kappa_2 = 1.0 \cdot 10^{-14}$.

The apparent strains and limits of the confidence intervals were converted to real strain using eq. (9) with the above values. Figure 6 a–b shows the converted versions of the strains in Figure 4 a–b, respectively. The respective errors are given in Figure 8. Figure 7 shows a problematic case, where the mechanical extensometer gives clearly non-zero strains only after approximately 100 seconds.

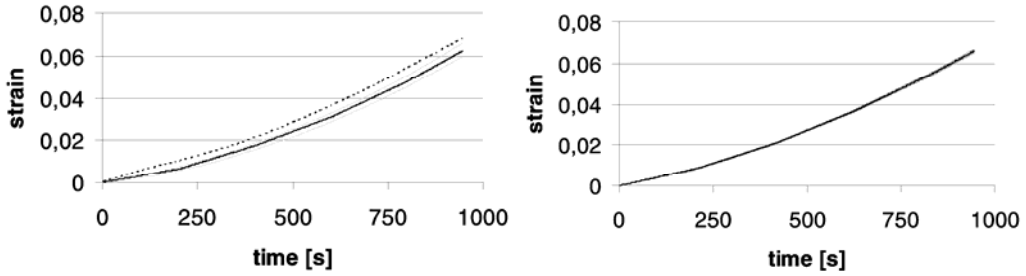



***Figure 6.*** *Strain vs. time from two tensile tests. Dotted line: strains measured by the mechanical extensometer. Black line: converted strains. Gray lines: implicit confidence interval estimates.*

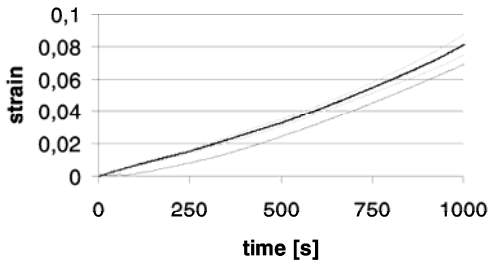



***Figure 7.*** *Strain vs. time from a tensile test where the strain value of the mechanical extensometer lies near zero for 100 s. Dotted line: strains measured by the mechanical extensometer. Black line: converted strains. Gray lines: implicit confidence interval estimates.*

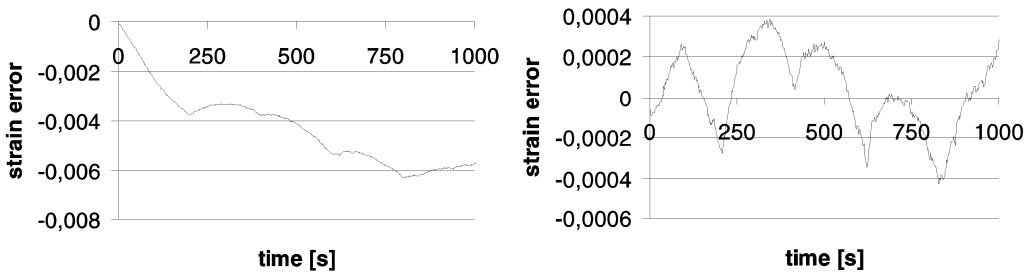



***Figure 8.*** *Measurement errors vs. time from two tensile tests.*

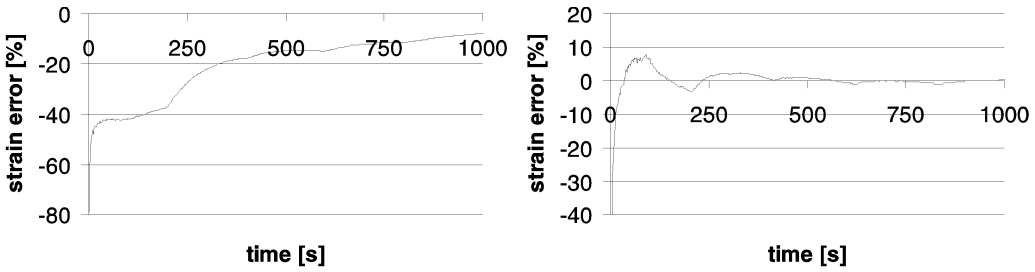



***Figure 9.*** *Relative measurement error (%) vs. time from two tensile tests.*

Relative errors, i.e. errors divided by the real strain given by the mechanical extensometer, are shown in Figure 9.

Figure 10 summarizes the errors from all tensile tests. The left subfigure shows the mean errors of all tests as a function of time. Because the grand mean error was forced to zero in calibration, the integral of the error curve is also zero. The means of the confidence intervals are also shown. The right subfigure shows the standard deviations of errors as a function of time, i.e. for each time instant, standard deviation is calculated from the strain errors of all tensile tests at that time instant.

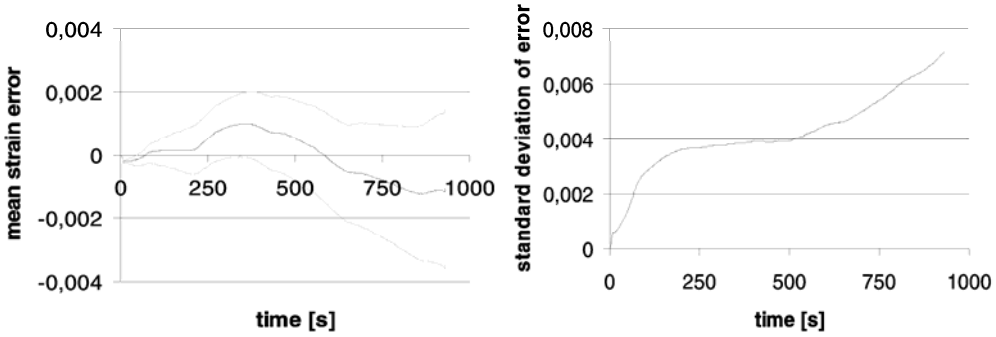



***Figure 10.*** *Left panel: mean strain errors of all tensile tests (black line) and means of the upper and lower bounds of the confidence intervals (gray lines). Right panel: standard deviation of errors of all tensile tests.*

## 5. DISCUSSION

Accurate strain measurements using digital image correlation require calibration. This can be seen by comparing Figures 5 and 6. However, significant residual errors still exist in approximately half of the test cases, as seen in Figure 5 (a).

Figure 8 (a) shows an example of residual errors, where clear trend is present. In some other cases, low frequency fluctuations occur. Figure 8 (b) in turn shows case with little errors, whence the error has a high frequency random-like fluctuation.

Figure 8 validates the previously observed [9] fact that the relative strain measurement error is reduced as the strain is increased. At the end of the tensile tests, the absolute values of the relative error are 8.6 % and 0.4 % in Figures 9 (a) and (b), respectively. These figures of merit can now be compared with the relative strain error of 1.3 % reported in [10]. Comparison reveals that both smaller and greater residual errors were obtained in our validation study.

Sources of error may be many. The first observation was the lighting intensity had changed temporally probably due to the ambient light from the fluorescent lamps that produce a flux with $f = 100$ Hz. Secondly, displacements do not seem to increase linearly with respect to time (see e.g. Figure 5), although the tensile testing machine works at a constant speed. Heterogeneous deformation explains the positive second derivative after the yield limit, because strains in the measurement area are greater than the overall strain.

There exist nonlinearities at the beginning of the tests, too. In four test cases cases, the measured strains of the mechanical extensometer were clearly non-zero only after appr. 100 seconds (Figure 7), although the optical measurements indicate an almost linear increase of strain. After 100 seconds, the strain measurements correlate but a gap remains. This phenomenon explains approximately one fourth of the errors in the summarizing statistics.

An explanation for the phenomenon may mismatch in time synchronization i.e. image capture and the tensile test may have been started asynchronously. However, a mismatch of almost two minutes sounds improbable, particularly when repeated four times. Another explanation may be related to the fact that the specimens were a little nonplanar. Consequently, the specimen is first straightened but not stretched in the beginning of the test. Intuitively, the straightening should induce measurable displacements, too. It remains unclear, how it could explain the difference between the mechanical and optical measurements. Changes in camera calibration parameters may also explain some errors.

The deviation of errors at a certain time instant can be regarded as an overall accuracy estimate of the optical extensometer, because the grand mean error was forced to zero. The zero mean error criterion does not apply to each time instant but over the whole time range. On the other hand, the confidence intervals in Figure 10 (a) indicate that the presumption of zero mean error should not be discarded.

Figure 10 (b) shows that the standard deviation of strain errors was appr. 0.7 % strain at the end of the tensile tests. As the corresponding strain was appr. 7 %, relative error was 10 %. Without any improvements to the validation procedure and further tests this figure (10 %) should be considered the best estimate of the relative accuracy of the optical extensometer.

## 6. CONCLUSIONS

Validation of the accuracy of an optical extensometer using mechanical extensometer was studied in this paper. In addition, a formula for strain measurement errors in the presence of optical radial distortion was derived. Values for the distortion parameters were obtained using this formula and the strains of the mechanical extensometer as a gold standard.

After calibration, the relative strain measurement errors of the best cases were at order 0.4 %, which is less than e.g. 1.3 % reported in [10]. However, relative errors were of order 10 % in several cases. An overall error estimate based on the standard deviation of the errors of all

test runs give a relative error estimate of 10 %, too.

Unexpected nonlinearities were detected in the strains measured by the mechanical extensometer in some test cases. Their sources remain unknown. Test cases with the nonlinearities largely caused to rise in the overall error estimate. Consequently, additional research is required to determine the accuracy of the optical extensometer and to find out, where the nonlinearities were from. The distortion calibration should also be validated using more conventional calibration approaches. More attention has to be paid also to lighting.

## 7. ACKNOWLEDGEMENTS

Finnish Funding Agency for Technology and Innovation (TEKES) and the industrial partners of the research project Process Development for Incremental Sheet Forming have supported this research. Jukka Tuomi, Seppo Kivivuori, Lotta Vihtonen from TKK, and Juha Tulonen from Rautaruukki are acknowledged for the valuable discussions.

## 8. REFERENCES

1. S. R. McNeill, W. H. Peters, W. F. Ranson, and M. A. Sutton: "A study of fracture parameters by digital image processing", in *Proc. of 18th Midwest Mech. Conf.* 12 (1983) 267–271.
2. T. C. Chu, W. F. Ranson, M. A. Sutton, and W. H. Peters: "Applications of digital-image-correlation techniques to experimental mechanics", *Experimental Mechanics* 25 (1985) 232–244.
3. G. Vendroux, and W. G. Knauss: "Submicron deformation field measurements: Part 2. Improved digital image correlation", *Experimental Mechanics* 38 (1998) 86–92.
4. H. Lu and P D. Cary: "Deformation measurements by digital image correlation implementation of a second-order displacement gradient", *Experimental Mechanics* 40 (2000) 393–399.
5. D. Amodio, G.B. Broggiato, F. Campana, and G. M. Newaz: "Digital speckle correlation for strain measurement by image analysis", *Experimental Mechanics* 43 (2003) 396–402.
6. T. Schmidt, J. Tyson, and K. Galanulis: "Full-field dynamic displacement and strain measurement using advanced 3D image correlation photogrammetry", *Experimental Technology* 27 (2003) 47–50.
7. P-C Hung and A. S. Voloshin: "In-plane strain measurement by digital image correlation", J. of the Braz. Soc. of Mech. Sci. & Eng. 25 (2003) 215–221.
8. C. Su and L. Anand: "A new digital image correlation algorithm for whole-field displacement measurement", in *Innovation in Manufacturing Systems and Tech., Proc. of Singapore-MIT Alliance Symp.* (2003).
9. J. Koljonen, O. Kanniainen, J.T. Alander: "An implicit validation approach for digital image correlation based strain measurements", in *Proc. of the IEEE International Conference on Computer as a Tool* (2007) 250–257.
10. C. Kanchanomai, S. Yamamoto, Y. Miyashita, Y. Mutoh, A. J. McEvily: "Low cycle fatigue test for solders using non-contact digital image measurement system", *International Journal of Fatigue* 24 (2002), 57–67.
11. S. Dumoulin, L. Tabourot, C. Chappuis, P. Vacher, R. Arrieux: "Determination of the equivalent stress-equivalent strain relationship of a copper sample under tensile loading", *Journal of Materials Processing Technology* 133 (2003) 79–83.
12. J. Koljonen, O. Kanniainen, and J. T. Alander, "Dynamic template size control in digital image correlation based strain measurements", in *Intelligent Robots and Computer Vision XXV: Algorithms, Techniques, and Active Vision* (2007) 67640L-1–12.

13. J. Koljonen, T. Mantere, and J. T. Alander: "Parameter optimization of numerical methods using accelerated estimation of cost function: a case study", in *Proc. of the Finnish Signal Processing Symposium* (2007).
14. M. Sonka, V. Hlavac, R. Boyle: *Image Processing, Analysis, and Machine Vision*, third edition, Thomson Learning, USA (2008).
15. S. Matsubara: "A study of a numerical control forming system. II. Incremental backward bulge forming of a sheet metal with a hemispherical head tool", *J Jpn. Soc. Technol. Plast.* 35 (1994) 1311–1316.
16. L. Lamminen, T. Tuominen, S. Kivivuori: "Incremental sheet forming with an industrial robot", *Materials Forum* 29 (2005) 331–335.

# Partially separable fitness function and smart genetic operators for area-based image registration

Janne Koljonen
University of Vaasa
P.O. Box 700, FIN-65101, Vaasa, Finland
Janne.Koljonen@uwasa.fi

**Abstract**

The displacement field for 2D image registration is searched by a genetic algorithm (GA). The displacement field is constructed with control points and an interpolation kernel. The common global fitness functions based on image intensities are partially separable, i.e. they can be decomposed into local fitness components that are contributed only by subsets of the control points. These local fitness components can be utilized in smart genetic operators. Partial separability and smart crossover and mutation operators are introduced in this paper. The optimization efficiency with respect to different GA parameters is studied. The results show that partial separability gives a great advantage over a regular GA when searching the optimal image registration parameters in nonrigid image registration.

**Keywords:** computer vision, genetic algorithm, genetic operators, image registration, partially separable fitness function.

## 1 Introduction

Image registration methods consist of a few basic tasks: selection of the image transformation model, selection of features, extraction of the features, selection of the matching criterion (objective function), and search for the optimal parameters of the transformation model (Zitove and Flusser, 2003). Hence, registration can be regarded as an optimization problem:

$$\mathbf{T}^* = \arg\min_{\mathbf{T}_{\text{registration}} \in S} h\left(\mathbf{T}_{\text{registration}}(\mathbf{F}_1), \mathbf{F}_2\right), \quad (1)$$

where $h$ is a *homology function* between two images, $\mathbf{T}_{\text{registration}}$ is an image transformation (the result is an image) to register images $\mathbf{F}_1$ and $\mathbf{F}_2$, and $S$ is the search space.

The homology function measures the correspondence of the homologous points of two images. In practice, the homology function is replaced by an objective (similarity, fitness, cost) function that is expected to correlate with $h$, because the homology function cannot be directly measured. There are two main categories of similarity functions: feature-based and area-based. In feature-based approaches, salient structures, e.g. corners, are extracted from the images. The positions of corresponding structures in the images are used estimate the homology function.

Area-based similarity functions consider the tonal properties (intensities) of each pixel as features. Thus the feature extraction step is trivial. In order to evaluate the objective function the reference image is transformed with a given registration transformation and the intensities of the transformed images are compared using a similarity metric. Typical metrics include cross-correlation and root-mean-square difference. Image registration may also include correction of optical distortions.

Area-based similarities can be computed using either small windows (templates) or entire images. The approach based on templates evaluates an area-based similarity function locally. On the other hand, the positions of the localized templates can be utilized in the calculation of the feature-based objective function. Usually templates are used to estimate local translations. If the image is subject to local deformations, the accuracy of the template based method deteriorates.

The image transformation $\mathbf{T}_{\text{registration}}$ and its search space $S$ should be such that the correspondence between the transformed images, according to eq. (1), can be as close as possible. On the other hand, the complexity of the image transformation model should be as low as possible so that the parameter search can be done efficiently and overfit-



Reprinted, with permission of FAIS, from [*AI and Machine Consciousness. Proceedings of the 13th Finnish Artificial Intelligence Conference (STeP2008)*. Partially separable fitness function and smart genetic operators for area-based image registration. Koljonen, J.].

ting to noise is avoided. The type of the transformation should take into account the premises of the registration task. For instance in multiview analysis (e.g. in stereo vision), a perspective transformation is applicable.

In nonridig medical registration, typical transformation models use e.g. basis functions, splines, finite-element methods (FEM) with mechanical models, and elastic models (Hajnal, Hill, and Hawkes, 2001). Basis functions are e.g. polynomials. In in-plane strain analysis, Lu and Cary (2000) have used a second-order Taylor series approximation to describe local displacements, whereas Koljonen et al. (2007) have used cubic B-splines, whose control points were optimized by a genetic algorithm. Veress at al. (2002) have used FEM to measure strains from pairs of cross-sectional ultrasound images of blood-vessels.

Usually registration requires iterative optimization starting from initial candidate(s) of transformation. The candidates are evaluated by an objective function. Optimization algorithms are used to create new candidate transformations using the evaluated ones, except in exhaustive search and Monte Carlo (random walk) optimization. The new candidates hopefully introduce fitness improvements, but it cannot be guaranteed in numerical optimization.

Optimization methods can be local or global. Local methods, such as hill-climbers, usually deal with only one candidate at a time and they utilize the local information, for instance, gradient, of the fitness landscape. Thus local methods are prone to get stuck to local optima.

Global methods are used to avoid the curse of local optima. They usually utilize parallel search with several concurrent candidates. Furthermore, information between the candidates can be exchanged. One group of such algorithms is genetic algorithms (GA) that are also utilized in this study (Forrest, 1993).

# 2 Genetic algorithm

A genetic algorithm with a partially separable fitness function is defined. It consists of encoding of nonrigid registration (deformation field), artificial image deformation, a scalar global fitness function, partially separable sub fitness functions, and genetic operators, some of which utilize the separability properties of the sub fitness functions.

## 2.1 Deformation encoding

The deformation field is encoded as displacements of control points (see Figure 1 a). Control points $\mathbf{O} = [\mathbf{o}_{m,n}] = [(o_x(m,n), o_y(m,n))]$ form a regular $M \times N$ (now $13 \times 20$) grid on the undeformed reference image $\mathbf{R}$. For a deformed (sensed) image $\mathbf{S}$, displacements $\mathbf{D} = [\mathbf{d}_{m,n}]$ of the control points are searched for to maximize the image similarity.

Both control points and displacements are encoded using floating-point numbers. Displacements are given in Cartesian coordinates $\mathbf{d} = (d_x, d_y)$. Thus there are $2MN$ (now 520) free floating-point parameters to be optimized.

## 2.2 Image deformation

Displacements are used to geometrically transform the reference image into an artificially deformed image $\mathbf{A}$. Thus the geometrical transformation $\mathbf{T}_{\text{registration}}(\mathbf{R}; \mathbf{O}, \mathbf{D})$ to register the image is defined as the following algorithm:

1. Displacements $\mathbf{D}$ at pixels $\mathbf{O}$ are interpolated to obtain a displacement vector for every pixel. A bi-cubic interpolation kernel is used (Sonka, Hlavac, Boyle, 2008).
2. Pixels of the reference image $\mathbf{R}$ are translated using the interpolated displacements.
3. The translated pixels are interpolated using bicubic interpolation and a regular grid, whose resolution is equal to that of the reference image. The resulting image $\mathbf{A_D}'$ has floating-point pixel values due to interpolation.
4. The pixel values of $\mathbf{A_D}'$ are truncated to 8 bits resulting in the artificially deformed image $\mathbf{A_D}$.

A similar algorithm is used to create the test images, too. However, in test image generation the effect of deformation on the image saturation and brightness as well as the influence of heterogeneous illumination is taken into account. Moreover, noise could be added, but in this study noise is neglected. More details on the test image generation are given in (Koljonen, 2008).

## 2.3 Scalar fitness functions

The global scalar fitness function is based on the tonal properties of the target image $\mathbf{S}$ and the artificially deformed image $\mathbf{A_D}$. With noiseless images and an optimal solution $\mathbf{D}_{\text{opt}}$ the deformed image and the sensed image $\mathbf{S}$ would be (almost) identical:

$$\mathbf{A}_{\mathbf{D}_{\text{opt}}} = \mathbf{T}_{\text{registration}}\left(\mathbf{R}; \mathbf{O}, \mathbf{D}_{\text{opt}}\right) \equiv \mathbf{S} \ , \tag{2}$$

In practice, images include noise. Consequently, there is a residual error at each pixel $(x, y)$:

$$\mathbf{A}_{\mathbf{D}_{\text{opt}}}(x, y) - \mathbf{S}(x, y) = \varepsilon \ , \tag{3}$$

Assuming that noise is independent and normally distributed, i.e. $\varepsilon \sim \text{NID}(0, \sigma)$, a common approach is to minimize the sum of squared difference (SSD) of the images:

$$\arg\min_{\mathbf{D}} \sum_{(x,y)\in\mathbf{A_D}} \left(\mathbf{A_D}(x, y) - \mathbf{S}(x, y)\right)^2 \ , \tag{4}$$

The corresponding global fitness function is:

$$f(\mathbf{D}) = \sum_{x=1}^{X}\sum_{y=1}^{Y}\left(\mathbf{A_D}(x,y) - \mathbf{S}(x,y)\right)^2 \ , \qquad (5)$$

In order to have a clearer interpretation of the fitness values the root-mean-square (RMS) value of the difference is used to present the values of the global fitness function in the experimental part of this study:

$$g(\mathbf{D}) = \sqrt{\frac{1}{XY}\sum_{x=1}^{X}\sum_{y=1}^{Y}\left(\mathbf{A_D}(x,y) - \mathbf{S}(x,y)\right)^2} \ , \qquad (6)$$

Obviously, minimizing eq. (6) minimizes eq. (5), too. Global fitness is used in trial evaluation.

## 2.4 Partially separable fitness function

The fitness function (eq. 5 or 6) has $2 \times M \times N$ free input parameters. Due to bi-cubic interpolation, each pixel in $\mathbf{A_D}$ is affected only by the 16 neighboring points of $\mathbf{D}$. This partial separability gives an opportunity to measure local fitness related to certain input parameters and use it to favor good building blocks in the reproduction phase of the genetic algorithm.

In strict terms, fitness function $f(\mathbf{x})$ is partially separable if it is the sum of $P$ positive functions $f_i$. Moreover, the sub-functions $f_i$ should be affected only by a subset of $\mathbf{x}$, the input variables (Durand and Alliot, 1998). In theory, each pixel could be used as a sub-function. Alternatively, small regions of contiguous pixels that have common control points could be searched and used as the region of the sub-functions. However, neither of these would be practical. Instead, local fitness functions related directly to each control point are used.

Each control point $\mathbf{d}_{m,n}$ has a local region of influence on the pixels of $\mathbf{A_D}$. Each pixel, to which $\mathbf{d}_{m,n}$ is one of the 16 closest control points, belongs to the local region of influence. However, solving the region is impractical.

Therefore, the ideal region is replaced by a square positioned around $\mathbf{d}_{m,n}$ (see Figure 1 b.). The horizontal and vertical dimensions ($W$, $H$) of the squares equal to the mean horizontal and vertical distances of the translated control points $\mathbf{O} + \mathbf{D}$, respectively. Thus the squares occupy each pixel, on average, approximately once.

The sub fitness function $f_{m,n}$ related to control point $\mathbf{d}_{m,n}$ is computed as follows:

$$f_{m,n}(\mathbf{D}) = \sum_{x=o_x(m,n)+d_x(m,n)-W/2}^{o_x(m,n)+d_x(m,n)+W/2}\ \sum_{y=o_y(m,n)+d_y(m,n)-H/2}^{o_y(m,n)+d_y(m,n)+H/2}\left(\mathbf{A_D}(x,y) - \mathbf{S}(x,y)\right)^2 , \qquad (7)$$

Sub-function $f_{m,n}$ is primarily affected by $\mathbf{d}_{m,n}$, but several other control points also affect it. These interactions are also a motivation to use global optimization in this study.

The global fitness function $f$ and the sub-functions $f_{m,n}$ do not exactly meet the definition of partially separable functions. Nevertheless, the sum of the sub-functions approximates the global fitness function:

$$\sum_{m=1}^{M}\sum_{n=1}^{N} f_{m,n}(\mathbf{D}) \approx f(\mathbf{D}) \, , \qquad (8)$$

## 2.5 Genetic operators

Displacement vectors $\mathbf{D}$ are modified using genetic operators. Smart initialization sets the original states of $\mathbf{D}$ in the population of trials, after which new trials are generated using reproduction.

In each iteration, two parents are randomly drawn from the population. Crossover operators recombine the displacement vectors of the parents to come up with a new trial, while mutation operators modify a parent trial to create an offspring.

Two crossover operators are used. *Uniform crossover* (Syswerda, 1989) recombines two trials

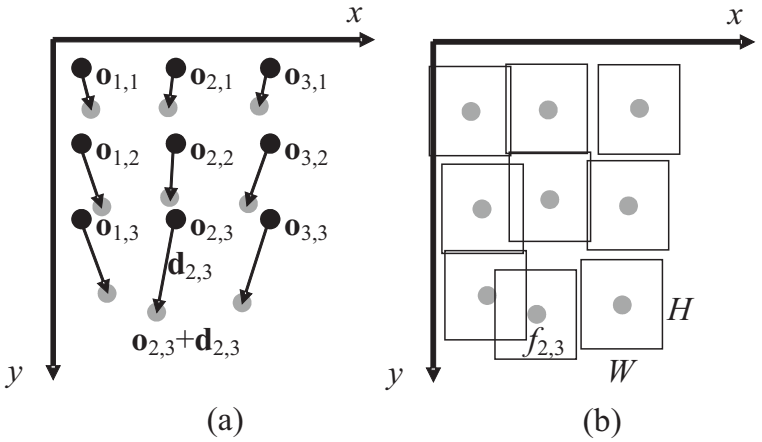



Figure 1. a: The principle of deformation encoding. The grid of control points **o**, displacement vectors **d**, and the translated control points (gray dots). b: The principle of local fitness evaluation.

totally randomly. A *smart crossover* operator utilizes the local fitness estimates to select the best building-blocks from each parent.

Two mutation operators are used. *Uniform mutation* treats each control point statistically equally, while in *smart mutation* larger mutations are applied to control points with poorer fitness.

#### 2.5.1 Smart initialization

A 'seed' trial is obtained by a template based registration algorithm described in (Koljonen, 2008). The displacements of the control points obtained for the sensed image **S** are interpolated by a bi-cubic kernel to obtain the seed trial, which should be relatively close to the optimum, for the genetic algorithm.

A population of $p$ trials is initialized using the seed trial. $p$–1 new trials are created by uniform mutation from the seed trial. Mutation is used to obtain enough diversity in the initial population, i.e. to span the search space adequately. In the subsequent optimization, only mutation can explore new search directions. Hence, an adequate spanning of the initial search space is required so that crossover can exploit the good building blocks of the trials.

#### 2.5.2 Smart crossover

Durand and Alliot (1998) introduced a genetic crossover operator for partially separable functions. In a similar way, a crossover operator based on local fitness (when minimizing $f$) is used in this study. On the basis of the local fitness, each displacement vector is selected from either of two parents as follows:

1. If $f_{m,n}(\text{parent}_1) - f_{m,n}(\text{parent}_2) < -\Delta$, then
   $\mathbf{D}_{\text{offspring}}(m,n) = \mathbf{D}_{\text{parent1}}(m,n)$
2. If $f_{m,n}(\text{parent}_1) - f_{m,n}(\text{parent}_2) > \Delta$, then
   $\mathbf{D}_{\text{offspring}}(m,n) = \mathbf{D}_{\text{parent2}}(m,n)$
3. If $|f_{m,n}(\text{parent}_1) - f_{m,n}(\text{parent}_2)| \leq \Delta$, then
   $\mathbf{D}_{\text{offspring}}(m,n) = \{\mathbf{D}_{\text{parent1}}(m,n) \text{ or } \mathbf{D}_{\text{parent2}}(m,n)\}$, (9)

where $\Delta$ is the (non-normalized) level of indeterminism. In step 3, the selection of the displacement is either totally random, like in this study, or it may still depend on the local fitness.

#### 2.5.2 Smart mutation

Local fitness can be utilized in mutation, too. It is presumed that the local fitness is proportional to the local alignment error. Therefore, a good local fitness implies that the corresponding control point should be translated only little, and subsequently the mutation energy should be small. For simplicity, the standard deviation $\sigma$ of the mutation operator is referred as mutation energy, because its units can be given in pixels.

Provided that the fitness function is subject to minimization and the optimum fitness is 0, the mutation energy can be e.g. directly proportional to the local fitness. The following smart mutation based on local fitness is used in this study:

$$\mathbf{D}_{\text{offspring}}(m,n) = \mathbf{D}_{\text{parent}}(m,n) + f_{m,n}(\text{parent}) \cdot \varepsilon, \qquad (10)$$

where $\varepsilon \sim \text{NID}(0, \sigma)$.

### 2.6 Pseudo-code of the GA

The following pseudo-code describes the essential parts of the genetic algorithm used in this study:

```
population[1]<-seedTrial(images);
for i from 2 to p do
population[i]<-
uniformMutation(population[1], 1*maxsigma);
end for;
for i from p+1 to n do
evaluateAndSort(population);
parent1<-population[ceil(rand*rand*rand*p)];
parent2<-population[ceil(rand*rand*p)];
if rand<crossoverprob // Only crossover
   if rand<smartcrossoverprob
   population[p+1]<-smartCrossover(parent1,
   parent2, delta);
   else
   population[p+1]<-
   uniformCrossover(parent1, parent2);
   end if;
else // Only mutation
   if rand<smartmutationprob
   population[p+1]<-smartMutation(parent1,
   rand*maxsigma, mdensity);
   else
   population[p+1]<-uniformMutation(parent1,
   rand*maxsigma, mdensity);
   end if;
end if;
end for;
```

Function `rand` returns a random number from [0, 1), whereas `ceil(arg)` rounds the argument to the nearest integer greater than the argument. The algorithm includes several parameters, whose explanations are given in Table 1.

## 3 Experiments and results

The objectives of the experiments were to test the feasibility and the efficiency of the proposed registration method, to study the effect of different GA parameters, to find an optimal set of GA parameters, and to understand the optimization mechanism of the proposed algorithm to come up with improvements. The test setups, a meta-optimization scheme and the results are given and discussed in this section.

Table 1. Explanations of the algorithm parameters.

| Parameter | Explanation |
|---|---|
| p | Population size |
| maxsigma | The maximum value of σ in mutation. |
| n | Number of iterations. |
| crossoverprob | Probability that solely crossover is applied. |
| smartcrossoverprob | Probability that the crossover operator is the smart one. |
| delta | Δ in smart crossover. |
| smartmutationprob | Probability that the mutation operator is the smart one. |
| mdensity | Mutation point density (mutation frequency). E.g. if mdensity = 1, then every control point is mutated, if mdensity = 0.5, on average half of the points are mutated. |

## 3.1 Test images

A series of 160 images was created using a seed image and the algorithm proposed in (Koljonen 2008). In the deformation process, saturation decrease and brightness increase are directly proportional to the local engineering strain. Moreover, the effect of nonuniform illumination is taken into account.

A significant benefit comes with the use of artificial test images; the homology function is known. Hence, the accuracy of the fitness function, which is used to estimate the homology function, can be computed.

The objective of the registration is to determine the correspondence between the seed image **R** and the last artificially deformed image **S** (Figure 2). The template based registration algorithm uses the intermediate images to determine the seed trial for the genetic algorithm, while the GA uses only the seed image and the last image. The seed image has been taken from a tensile test specimen with a random speckle pattern obtained by spray-paint.

## 3.2 Meta-optimization

Table 1 shows that there are several GA parameters that may have significant effects on the optimization performance. GA parameters have been optimized by another genetic algorithm, called meta-GA, in several studies (see e.g. Alander 1992; Koljonen and Alander 2006). In this study, the meta-GA approach would have been computationally expensive, and

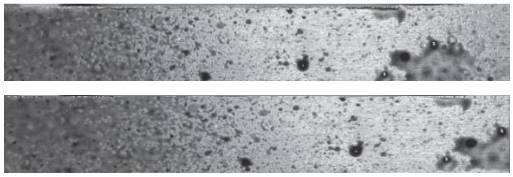


Figure 2. Seed image **R** (top) and the last artificially deform image **S** (bottom).

therefore a one-dimensional line search approach was adopted.

If it was assumed that the GA parameters have no interaction on the GA performance, an assumption which is undoubtedly too simplifying, each parameter could be optimized separately.

In order to have more reliable optimization results, a sequential optimization method is used: After optimizing one parameter (dimension), that dimension is fixed to the local (one-dimensional) optimum. This method carries evidently an implicit assumption that any dimension optimized after dimension *k* has no effect on the position of the one-dimensional optimum of that dimension.

In order to maintain good comparability, the number of iterations *n* was fixed to 1000. The initial values of the other dimensions were: popsize = 100, crossoverprob = 0.6, smartcrossoverprob = 0.8, delta = 0, smartmutationprob = 0.8, and mprob = 1.

In meta-optimization, the GA parameters were varied as follows, respectively:

1. maxsigma = {0.02, 0.04, …, 0.1} pixels,
2. popsize = {50, 75, …, 150},
3. crossoverprob = {0.2, 0.4, …, 1.0}
4. smartcrossoverprob = {0.2, 0.4, …, 1.0}
5. smartmutationprob = {0.2, 0.4, …, 1.0}
6. mdensity = {0.4, 0.6, 0.8, 1.0}
7. delta = {0, 0.2, …, 1.0}

## 3.3 Effect of GA parameters

Figure 3 shows how mutation energy (maxsigma in Table 1, corresponding to σ in eq. 10) affects optimization speed. The solid line represents the fitness after 1000 trials whereas the dashed line is the homology function that gives the mean registration (alignment) error in pixels.

Two notions from Figure 3: the fitness and homology functions have a strong correlation, and the optimum of σ lies appr. at 0.06 pixels, a value to which σ was fixed in the subsequent tests.

The effect of population size is given in Figure 4. It shows a weaker correlation between fitness and homology distance. Population size was fixed to 150, because fitness value was used as the optimization criterion.

Fitness and homology distance against crossover probability is shown in Figure 5. 0.4 was found to

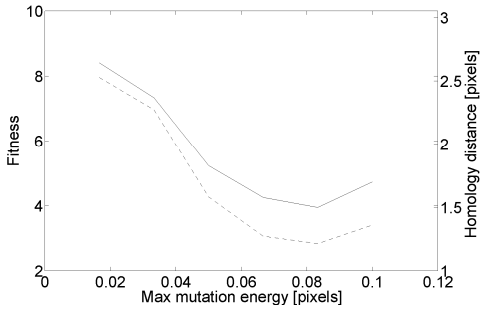



Figure 3. Effect of mutation energy. Solid line: fitness, dashed line: homology distance.

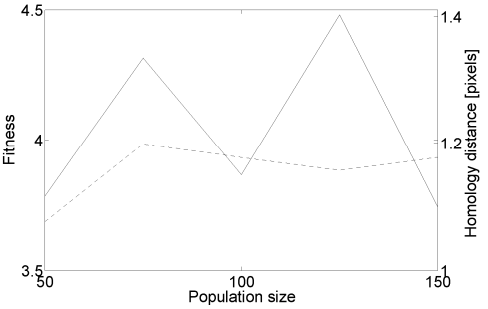



Figure 4. Effect of population size. Solid line: fitness, dashed line: homology distance.

be an optimal selection. However, variation with respect to crossover probability seems to be small. Moreover, the sampling in the optimization is rather sparse. Consequently, the optimization does not give reliable results, at least as for crossover probability.

Figures 6 and 7 validate the efficiency of the smart crossover and mutation operators, respectively. Figure 6 suggests that using solely smart crossover gives both superior fitness and homology distance after 1000 iterations, when comparing to parameter setups, in which also uniform crossover is occasionally applied.

Figure 7 shows that the homology distance attains its minimum when smartmutationprob = 0.8. This gives some indication, may it be rather weak, that it might be beneficial to include uniform mutation to the genetic operators, too.

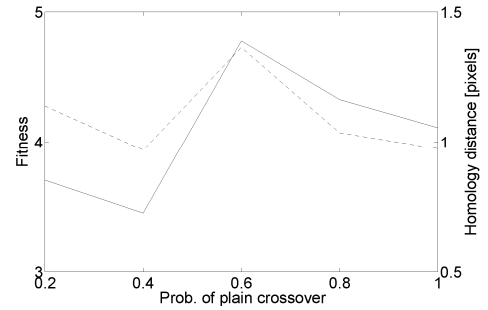



Figure 5. Effect of crossover domination. Solid line: fitness, dashed line: homology distance.

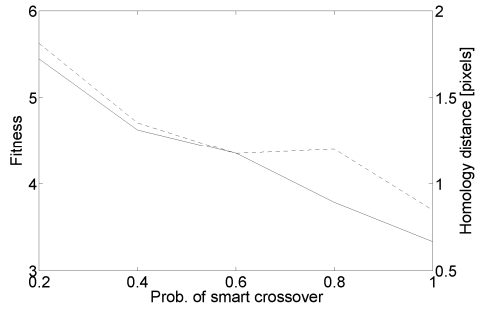



Figure 6. Effect of smart crossover domination. Solid line: fitness, dashed line: homology distance.

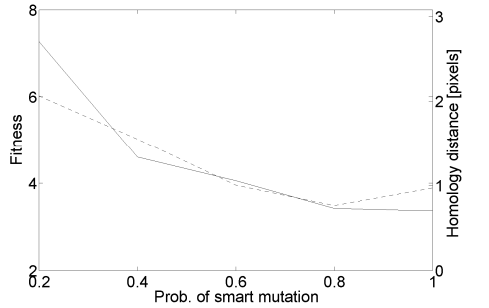



Figure 7. Effect of smart mutation domination. Solid line: fitness, dashed line: homology distance.

Figure 8 shows that mutation frequency should to 1, i.e. each time mutation is applied, it should be applied to each control point. Nevertheless, other more efficient mutation strategies may exist.

Figure 9 gives more detailed information concerning the determinism of the smart crossover. In smart crossover, each control point of the offspring trial is selected from either of the parents. If $\Delta$ is enlarged, smart crossover resembles more and more uniform crossover.

If $\Delta = 0$, control points that are estimated to be nearer to the solution are selected. This corresponds to an attempt to construct an optimal combination of the parents. Such a strategy might be too greedy, but Figure 9 shows that it is optimal in this case. The

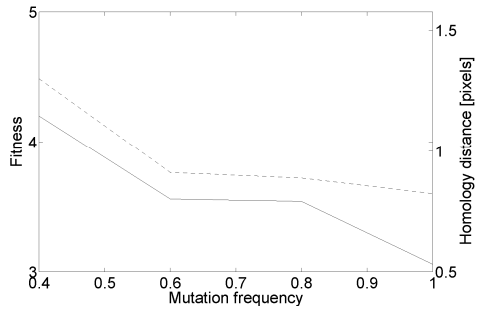



Figure 8. Effect of mutation frequency. Solid line: fitness, dashed line: homology distance.

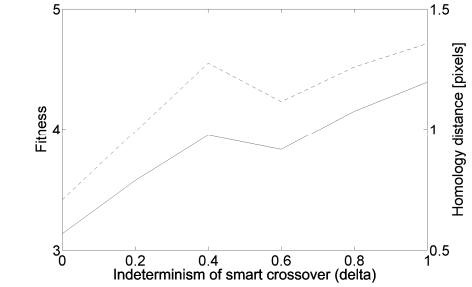



Figure 9. Effect of indeterminism of smart crossover (nonnormalized Δ). Solid line: fitness, dashed line: homology distance.

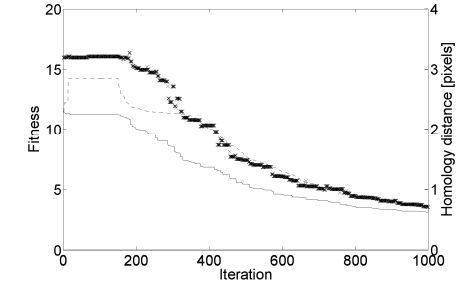



Figure 10. Development of the best (solid) and worst (dashed line) fitness of the population. × = homology distance of the best trial.

results are in line with the results in Figure 6, where smart crossover dominated uniform crossover. As a conclusion, smart crossover outperforms uniform crossover.

### 3.4 GA performance

The development of fitness in a single GA run is given in Figure 10. It shows that fitness is improved significantly during optimization despite the smart initialization.

The difference between the worst and best fitness of the population is used to estimate the diversity of the population. Now the diversity decreases almost consistently, but it never vanishes. This observation indicates that the decrease of fitness could continue slightly after the 1000 iterations, even though the rate of improvement was rather slow at the end of the GA run.

On the other hand, the diversity is rather low at the end of the GA run, which indicates that the population size was probably selected quite optimally. Population size and diversity should namely have a positive correlation.

In order to determine the feasibility of the fitness function (eq. 6) fitness and homology distance are compared. Figure 10 shows that fitness and homology distance have a strong correlation. Computing linear correlation gives: $r = 0.995$ ($p < 0.001$). Consequently, eq. (6) proves to be an efficient fitness function to minimize the homology distance.

However, the high correlation does not guarantee that an arbitrarily low (sub-pixel) alignment error could be achieved using eq. (6). In fact, when using only the 50 last iteration, $r = 0.935$. Figure 10 shows that the residual alignment error is still 0.7 pixels at the end of the best optimization run.

As for GA efficiency, it seems that the smart operators make the GA faster and more robust. However, no deviation figures were estimated due to computational complexity.

Figure 11 shows the evolvement of meta-optimization. The results indicate that the GA parameters have a significant influence on the GA efficiency, but the meta-optimization gave some clear guides to the selection of them, particularly as for the selection of genetic operators. It is yet unclear, how optimal the GA parameters, found by the one-dimensional optimization scheme, are.

The evolvement of two control points during an optimization run is studied in Figure 12. In the left panel, the control point position is initially (obtained by smart initialization) appr. one pixel away from the correct position (target). During optimization, the control points almost resides the correct position, but finally it drifts appr. 0.3 pixels away from the target.

In the right panel, the control point is initially appr. 3 pixels from the target. In the beginning, the homology distance increases, after which the control point starts to approach the target. It seems that the optimization was stopped too early.

## 4 Conclusions and future

It was proposed how the nonrigid registration problem can be solved using control points of displacements, bi-cubic interpolation of both displacements and intensities, intensity based global fitness function, and search of optimal control point positions by a genetic algorithm. It was also proposed how the global fitness function can be decomposed into local

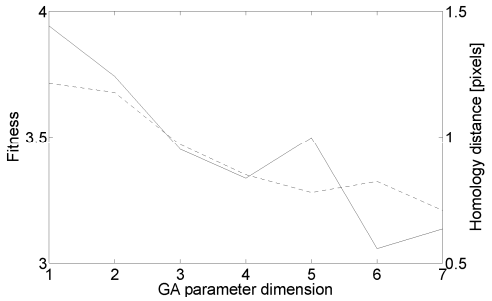



Figure 11. Effect of the meta-optimization of the GA parameters. Solid line: fitness, dashed line: homology distance.

sub fitness functions using the principle of partial separability. The sub fitness functions were utilized in smart crossover and mutation operators.

The results show that the smart genetic operators improve the optimization speed significantly. The displacement error of registration was 0.7 pixels at the end of the best GA run. Improvements to optimization speed are needed to make the method practically more feasible.

One possibility to speed up optimization might be to use the momentum of the control points, i.e. the mutation operator could favor the direction, to which fitness was improved. Such algorithms that utilize experience are called cultural algorithms.

On the other hand, the second example in Figure 12 showed that although the global fitness improved, the homology distance of the individual control point increased temporarily. Hence, the relationships between local and global fitness and homology distance should be studied more closely.

## Acknowledgements

Finnish Funding Agency for Technology and Innovation (TEKES) and the industrial partners of the research project Process Development for Incremental Sheet Forming have financially supported this research.

## References

J.T. Alander. On optimal population size of genetic algorithms. In *Proceedings of the 6th Annual IEEE European Computer Conference on Computer Systems and Software Engineering*, 65–70, 1992.

N. Durand and J–M. Alliot. Genetic crossover operator for partially separable functions. In *Proceedings of the Third Annual Conference on Genetic Programming*, 487–494, Madison, Wisconsin, USA, 1998.

S. Forrest. Genetic algorithms: principles of natural selection applied to computation. *Science*, 261(5123): 872–878, 1993.

J. V. Hajnal, D. L. G. Hill, and D. J. Hawkes (eds.). *Medical Image Registration*, CRC Press, Boca Raton, 2001.

J. Koljonen and Jarmo T. Alander. Effects of population size and relative elitism on optimization speed and reliability of genetic algorithms. In *Proceedings of the Ninth Scandinavian Conference on Artificial Intelligence*, 54–60, 2006.

J. Koljonen, T. Mantere, O. Kanniainen, and J. T. Alander. Searching strain field parameters by genetic algorithms. In *Intelligent Robots and Computer Vision XXV: Algorithms, Techniques, and Active Vision*, Proc. of SPIE, 67640O-1–9, 2007.

J. Koljonen and Jarmo T. Alander. Deformation image generation for testing a strain measurement algorithm. Submitted to: *Optical Engineering*, 2008.

H. Lu and P. D. Cary. Deformation measurements by digital image correlation implementation of a second-order displacement gradient. *Experimental Mechanics*, 40(4): 393–399, 2000.

M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis, and Machine Vision*. Third edition, Thomson Learning, USA, 2008.

G. Syswerda. Uniform crossover in genetic algorithms. In *Proceedings of the Third International Conference on Genetic Algorithms*, 2–9, 1989.

A. I. Veress, J. A. Weiss, G. T. Gullberg, D. G. Vince, and R. D. Rabbitt. Strain measurement in coronary arteries using intravascular ultrasound and deformable images. *J. of Biomechanical Engineering*, 124(6): 734–741, 2002.

B. Zitove and J. Flusser. Image registration methods: A survey. *Image and Vision computing*, 21: 977–1000, 2003.

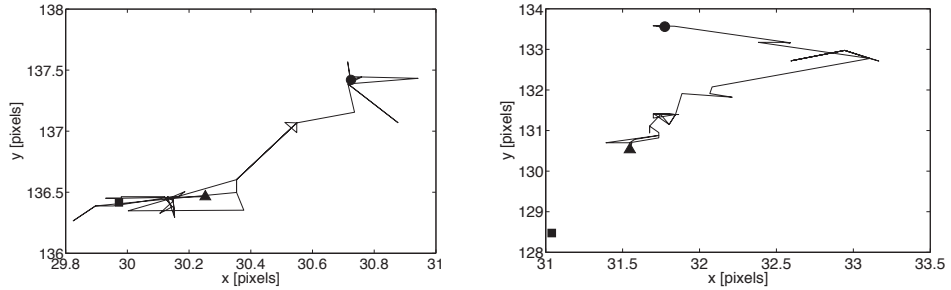



Figure 12. Two examples of control point evolvements. • = initial position, ▲ = final position, ■ = target position.

Optical Engineering 47(10), 107202 (October 2008)

# Deformation image generation for testing a strain measurement algorithm

**Janne Koljonen**
**Jarmo T. Alander**
University of Vaasa
Department of Electrical Engineering and Automation
P.O. Box 700
FIN-65101, Vaasa, Finland
E-mail: janne.koljonen@uwasa.fi

**Abstract.** An optical extensometer was tested using artificially deformed images with a known strain field. A real image series from a tensile test was used to obtain realistic deformation parameters, including spatial and temporal strain characteristics, changes in tonal pixel properties due to deformation, and the effect of nonuniform illumination. These parameters are used to artificially deform a real image taken from an object with a random speckle pattern. The signal-to-noise ratio of the resulting artificially deformed images is varied by applying a blurring pillbox filter and additive Gaussian noise to them. The optical extensometer uses digital image correlation to track homologous points of the object, and further to measure strains. The strain measurement algorithm includes a heuristic to dynamically control the template size in image correlation. Furthermore, several other methods to improve the accuracy-complexity ratio of the algorithm exist. The effects of different parameters and heuristics on the accuracy of the algorithm as well as its robustness against blur and noise are studied. Results show that the proposed test method is practical, and the heuristics improve the accuracy and robustness of the algorithm. © *2008 Society of Photo-Optical Instrumentation Engineers.* [DOI: 10.1117/1.2993319]

Subject terms: algorithm testing; computer vision; pattern recognition; strain analysis; test image generation.

Paper 080376R received May 13, 2008; revised manuscript received Aug. 13, 2008; accepted for publication Aug. 16, 2008; published online Oct. 9, 2008.

## 1 Introduction

Measuring strains in a uniaxial tensile test using digital speckle correlation basically incorporates non-rigid-body image registration[1] of the homologous points of the test object. In addition to material engineering, non-rigid-body registration is common, e.g., in medical imaging. Medical image registration[2] tasks typically handle 3-D volume element (voxel) images, which may furthermore be of different modality, e.g., x-ray and magnetic resonance images.

Algorithms for digital speckle correlation strain measurements (optical and video extensometers) have been developed particularly during the last decades,[3–10] since the rapidly increasing performance of microprocessors and the advances in digital imaging technology have made such applications feasible. Commercial equipment and software are nowadays also available for various tasks of optical deformation measurements.[11,12] However, it still seems to be rather an exception to use local strain measurements in experimental mechanics. Reductions in costs and computation time are probably awaited. Parallel image processing by embedded field-programmable gate arrays (FPGAs) should at the latest open the way for many new real-time computer vision tasks.[13]

It is important to measure the elastic properties of materials in numerous fields of engineering. Deformation measurements based on images have been applied at least to sheet metal,[14,15] metal foams,[16,17] solder joints,[18] composites and textiles,[19,20] thin films,[21] mineral wools,[22] wood,[23,24] biological mechanical systems (bones, ligaments, etc.),[25] and human skin.[26]

A serious limitation when developing algorithms to measure or register objects subject to deformation is the lack of reliable ground truth of the test images. It is relatively easy to create reference images with a known rigid-body transformation using, e.g., accurate mechanical stages.[4,9] Determination of local strains in an arbitrarily deformed object while imaging it is fairly difficult, or even impossible.

Amodio et al. used a mechanical extensometer to validate their strain measurements, but an extensometer enables measuring strain only in a single interval at a time.[7] Synthetic images are also common when testing computer vision algorithms and systems. Artificially created test images have been utilized, e.g., when testing video extensometers,[6,10] structural light vision software,[27] halftoning methods,[28] and methods for nonrigid medical registration.[29] Synthetic test images seem to be the most feasible method to obtain reference images with a known deformation field.

A regular or stochastic pattern on the object is often required in computer-vision-based strain measurements. Synthetic test images of a regular pattern have been created, e.g., by defining the spatial gray scale profiles of dots that are then aligned regularly in a mesh that is further warped by a prescribed displacement field.[10] Synthetic images have also been obtained using real images as a base. Lu and Cary generated deformation images from such a real sample image, using second-order displacement

functions.[6] The images were used to test a video extensometer that used second-order Taylor series in displacement field approximation.

Noise is easily controlled in synthetic images. However, surprisingly many studies seem to use noiseless synthetic images despite the great opportunity to test the noise tolerance of the algorithm. Robinson and Milanfar made a comprehensive study of the fundamental performance limits in image registration. They made test images from real sample images using discrete Fourier transforms. White noise was also added to the images. Theoretical considerations revealed that the performance bound of image registration depends only on the gradients (or texture) of the image. Thus the bandwidth of the image is a fundamental limitation, as expected. In their experimental part, seven registration methods were compared with varying signal-to-noise ratio (SNR) levels.[30]

In this paper, a method to create synthetic images by artificially deforming real images and to control the SNR is presented. Moreover, a strain measurement algorithm is tested using the synthetic test images. The measurement algorithm, presented in Sec. 2, includes several heuristics to increase accuracy, robustness, and computational efficiency. Section 3 introduces the procedure to generate test images that resemble real deformation images with respect to geometrical and tonal image transformations. Section 3 also includes the resulting deformation model that is used when testing the video extensometer. Section 4 includes results from test runs that study the error characteristics of the strain measurement algorithm with respect to the internal parameters of the algorithm and noise of the test images. In addition, a dynamic template-size control strategy and an evolutionary optimization strategy are tested and compared with the basic algorithm.

## 2 Strain Measurements Based on Digital Speckle Correlation

A strain measurement program called DeforMEERI is introduced in this section. Previously published[31–35] technical details are briefly revisited, and some new heuristics to improve the accuracy-complexity ratio are introduced.

### 2.1 *Previous Work*

The measurement setup and the basic principle of the algorithm were introduced in Ref. 31. In addition, an implicit error estimation approach was suggested, and the effects of different algorithm parameters on accuracy were studied. Given the basic algorithm and a method to estimate measurement accuracy, a dynamic template size control (DTSC) strategy was suggested in Ref. 32. DTSC was found to improve the accuracy-complexity ratio up to nine fold.

In a third paper,[33] an optimization scheme that uses a genetic algorithm[36] and an accelerated fitness-function estimation approach were introduced to optimize the program parameters offline. The approach enables users to optimize the internal parameters automatically. It was concluded that the accelerated optimization followed by a full strain measurement required only a little more than twice the time that the plain strain measurement had required. Thus the alternative method of trial and error would not be competitive for finding a suboptimal set of program parameters.

The principle of artificially deforming images was utilized already in Ref. 34, where artificially deformed images were compared with real deformed images in order to measure the deformation field (strains). In that study, a genetic algorithm was used to look for deformation parameters such that the similarity between the artificial and real deformed images was maximized.

The basic principle of digital speckle strain measurements is to trace homologous points (fiducial points, landmarks) of the object surface using images taken before and after deformation. Linear, or engineering, strain is the change of length of a line relative to its original length:

$$\varepsilon = \Delta l / l_0. \tag{1}$$

If the movements of the homologous points are given as a displacement field $\mathbf{u}(x,y,z)$, then the transverse $(x)$ and longitudinal $(y)$ strains are given, respectively, by

$$\varepsilon_x = \Delta u_x / \Delta x \quad \text{and} \quad \varepsilon_y = \Delta u_y / \Delta y, \tag{2}$$

where $\Delta u_x$ is the relative displacements of two points whose transverse distance is $\Delta x$. If $\Delta x \to 0$, the derivative of the displacement field is obtained.

Because strains are calculated from the relative movements of the homologous points, their units of measurements can be arbitrary (e.g., pixels can be used). However, optical distortions should be compensated by calibration to obtain accurate measurements. The effect of radial distortion was studied in Ref. 35.

### 2.2 *Coarse-to-Fine Search*

After predicting the position of the grid point as presented in Ref. 31, a search area (ROI) is determined around the predicted position. The correlation function between the template and the deformed image is estimated by translating the template in the ROI. Because cross-correlation has a strong autocorrelation in small translations, the following coarse-to-fine search strategy is feasible:

1. Cross-correlations are sampled with intervals of $\Delta r$ pixels to find a preliminary correlation peak. The template size is $Z_{xy}$ at this stage.
2. Cross-correlations are sampled around the preliminary correlation peak, at distance $\Delta r$, using a sampling interval of 1 pixel. The peak position is updated if necessary.
3. Cross-correlations are resampled around the updated peak, at distance $r_i$, using a larger template of size $Z_{xy}+Z_s$.
4. The samples obtained in step 3 are used to interpolate the subpixel position of the grid point.

The objective of this iterative approach is to accelerate the grid-point positioning. Time reduction is achieved in two ways: The search space is reduced by using lower sampling frequency, and the sampling time is reduced by using smaller templates. The search-space reduction is proportional to $1/\Delta r^2$; e.g., if $\Delta r$ is changed from 1 to 2, the search time is reduced to approximately $1/4$. On the other hand, too large $\Delta r$ increases the risk of mismatch. In prac-

tice, $\Delta r$ should be 2 or 3; larger values do not reduce the complexity significantly, because the other computation tasks independent of $\Delta r$ begin to dominate.

The strategy of using a larger template in the interpolation phase also helps increase accuracy without any significant increase in complexity. The increase of accuracy when the template is enlarged was shown in previous studies.[31,32] Although some cross-correlations are sampled twice (with different template size), the complexity is not significantly increased, because the ratio of resampled correlations is low. For typical values $\Delta r=2$ and $r_i=2$ and an ROI of $20 \times 80=1600$ pixels, the fraction of resampled correlations is $100\% \times (1+2r_i)^2/(1600/\Delta r^2)=6\%$.

## 3 Test Image Generation

The objective of this study is to measure the accuracy of the strain measurement algorithm with respect to image quality. By using artificial test images two major advantages are gained: the actual ground-truth strains are exactly known, and different image quality aspects are easily controlled. The objective of the test image generation is to obtain image series whose spatial strain fields and temporal strain paths are realistic, and whose signal-to-noise ratio can be controlled.

The image-series generation consists of the following fundamental steps: strain field modeling, modeling of tonal pixel value changes caused by deformation and nonuniform illumination, image morphing (i.e., artificial stretching), application of pixel value changes, application of motion blur, and application of Gaussian pixel noise. In the modeling stages, an image series taken from a uniaxial tensile test is used to obtain realistic parameters. Image morphing is applied to a real image (*seed*) with a random pattern. Pixel value changes and noise operators are in turn applied to the morphed images. In the following, all these steps are discussed in detail. In addition to the principles, the resulting model is given in this section.

### 3.1 *Strain Field Acquisition*

In order to have a realistic model of strains in image morphing, an image series was captured during a uniaxial tensile test. A test piece was painted with black spray paint to have a random speckle pattern on it. The resulting image series consisted of one undeformed image and 181 deformed images.

The same strain measurement algorithm that was tested with the artificial test images was used to obtain the strain paths of the image series. The numbers of transverse and longitudinal strain samples were 6 and 12, respectively. Moreover, strains are measured from every tenth image. Strains for the rest of the images were obtained by interpolating the strain samples with cubic B-splines. The $y$-strain samples of the last image (with the largest deformations) are shown in Fig. 1.

To obtain strain and displacement values ($x$ and $y$) for every pixel, strains are converted to displacements. Displacements and strains are subsequently interpolated using cubic B-splines. The centers of gravity of the four neighboring nodes that are used to compute the strains and the corresponding strains are used as the control points when interpolating strains.

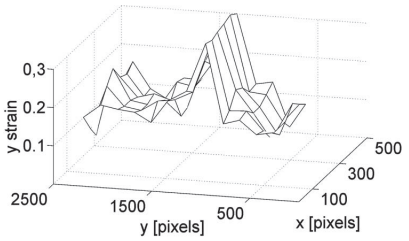



**Fig. 1** Samples of longitudinal strain that were used as a strain field model of the last image (largest deformation).

It may sound hazardous to use the same algorithm to make a strain model and to be the target of the test algorithm that uses the strain model. However, the strain path modeling can be done arbitrarily, because the model is used to make synthetic test images. Hence, the strain model is the ground truth for the strains of the artificial test images. An equally valid method would be to use, e.g., an analytic or stochastic function for strain modeling, but the strain paths would probably be less realistic.

### 3.2 *Saturation and Brightness Changes Due to Deformation*

When an object is stretched in one (say, the longitudinal) direction, distances between the points on its surface change. If the area of the plane spanned by the $x$ and $y$ axes (the specimen surface facing the camera) also changes, the reflective properties of the surface may change.

Suppose a one-dimensional case (Fig. 2): A white (grayscale color 255) rod is of length 10. The rod is painted with four small black (grayscale color=0) color particles of length 1 in a row at the center of the rod [Fig. 2(a)]. Suppose that the rod is imaged with one pixel of length 10. The image color of the rod is thus $[(10-4)\times 255+4\times 0]/10=153$. The rod is stretched by 100% ($\varepsilon=1$), so that its length is now 20. The small color particles do not stretch, but they move with the displacing points [Fig. 2(b)]. The resulting rod is imaged with two pixels. Each pixel includes two particles of length 1 and a white rod of length 8. Thus the resulting image color is $8\times 255/10=204$; the intensity of the object has increased.

Theoretically, the intensity change would be a nonlinear function of background and paint colors, pigment particle density, and strain. For simplicity, linear models are used in this study. Because the object under deformation is imaged with a color camera, the following hypotheses were made: the changes of saturation ($S$) and brightness ($B$) are directly proportional to the change of local surface area. The relative area change can be computed using $\varepsilon_x$ and $\varepsilon_y$:

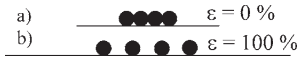



**Fig. 2** Model for change of brightness in deformation. ●=pigment particle.

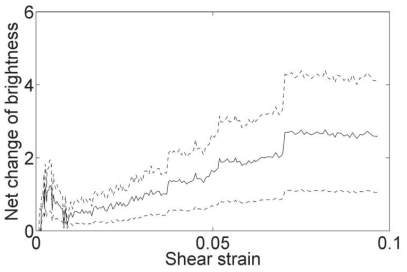



**Fig. 3** Change of brightness as a function of average shear strain. Solid line: mean; dashed lines: min and max.

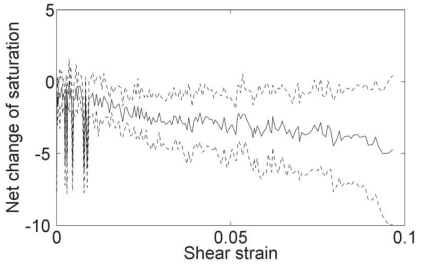



**Fig. 4** Change of saturation as a function of average shear strain. Solid line: mean; dashed lines: min and max.

$$\Delta A_{xy}/A_{xy0} = (\varepsilon_x + 1)(\varepsilon_y + 1) - 1 = \varepsilon_x\varepsilon_y + \varepsilon_x + \varepsilon_y \approx \varepsilon_x + \varepsilon_y = \gamma_{xy}, \quad (3)$$

where $\varepsilon_x\varepsilon_y \approx 0$ for small strains, and $\gamma_{xy}$ is the shear strain in the $x$-$y$ plane. Thus the linear models are

$$\Delta S(m,n) = c_S\gamma_{xy}(m,n) \quad (4)$$

and

$$\Delta B(m,n) = c_B\gamma_{xy}(m,n), \quad (5)$$

where $\Delta S$ and $\Delta B$ are the saturation and brightness changes, respectively, at pixel $(m,n)$.

The coefficients $c_S$ and $c_B$ can in principle be obtained by following physical points of the specimen and recording their saturation and brightness values, by measuring local strains, and by computing the shear strain or area chance. However, if single pixels were used in saturation and brightness comparison between different images, misalignment errors would become large and the result would be noisy. Therefore, the images were downsampled by binning twelve rectangular areas as follows: The 12 strain sampling centers of the first column of the sampling grid ($\mathbf{s}_{1,j}$) are used as reference points on each image. Now the rectangle for row $j$ is defined by the following top left ($\mathbf{s}_{\mathrm{TL},j}$) and bottom right ($\mathbf{s}_{\mathrm{BR},j}$) corners, respectively:

$$\mathbf{s}_{\mathrm{TL},j} = (s_{x,1,j} \quad s_{y,1,j} - 4)^{\mathrm{T}}, \qquad \mathbf{s}_{\mathrm{BR},j} = (s_{x,I-1,j} \quad s_{y,1,j} + 4)^{\mathrm{T}}. \quad (6)$$

Hence, nine rows of pixels are binned by averaging, and because the reference point should almost correspond to the same physical point of the object on different images, the binned area should closely correspond to the same physical area. Finally, the mean values of brightness and saturation of the binned areas are compared with the corresponding areas in the original image.

Twelve differences of brightness and saturation were obtained for each image. Means, minima, and maxima were computed from these figures. The results for brightness and saturation are plotted against average shear strains in Figs. 3 and 4, respectively.

Due to the almost monotonic behavior, it can be concluded that the hypothesis that brightness increases and saturation decreases with increasing shear strain seems to be valid. However, the variation seems to be large and the maximum change of brightness and saturation were only approximately 3 and −8 intensity units (with 8 bits), respectively. Consequently, the brightness and saturation changes should not be significant sources of measurement error.

In order to obtain the coefficients $c_B$ and $c_S$, a first-order (linear) least-mean-squares fitting was used, and the slopes of the fitted lines were used as the coefficients. The results were $c_B = 35.3$ and $c_S = -17.6$.

### 3.3 *Modeling Spatial Illumination Nonuniformity*

Spatial nonuniformities of illumination may be harmful in strain measurements, because the physical points of the object move during a tensile test and thus they are subject to different illumination in different images. Subsequently, the brightness values of corresponding pixels may differ. Temporal nonuniformities likewise affect brightness even when no movement occurs. Fortunately, the correlation coefficient is quite robust against overall brightness level changes. Nevertheless, if the illumination gradients are steep, the patterns may change significantly and cause significant measurement errors.

In uniaxial tensile tests, object points mostly move longitudinally. Therefore, only the longitudinal illumination (or brightness) profile $\Sigma B'(y)$ was determined. Let us assume that the spatial illumination distribution is temporally constant and that the imaging geometry is also constant. The resulting brightness profile $\Sigma B'(y)$ tells how the brightness in pixel $(x,0)$ would change if the corresponding physical point moved to pixel $(x,y)$. If the point in pixel $(x,y_1)$ in turn moved to pixel $(x,y_2)$, the brightness would change from $B(x,y_1)$ to

$$B(x,y_2) = B(x,y_1) + \Sigma B'(y_2) - \Sigma B'(y_1). \quad (7)$$

The brightness profile is determined by integrating the longitudinal brightness derivative. In theory, the derivative can be measured by tracking physical points during a tensile test. If point $\mathbf{s}$ is moved by $\Delta y$ and its brightness changes by $\Delta B$, the derivative at $\mathbf{s}$ is

$$B'(\mathbf{s}) = \Delta B/\Delta y. \quad (8)$$

In practice, misalignment errors would be large. Therefore, another procedure was followed. First, a series of artificially deformed images $\mathbf{A}_k$ was created using the procedure described in the Sec. 3.4, except that the brightness

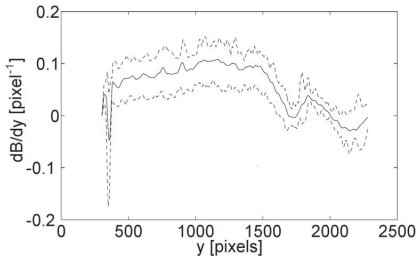



**Fig. 5** Vertical derivative estimate of brightness ($B$) as a function of vertical position ($y$). Solid line: mean; dashed lines: min and max.

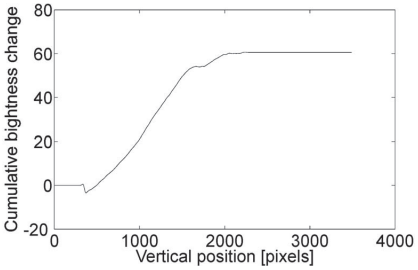



**Fig. 6** Cumulative brightness as a function of vertical position (brightness profile).

profile was set at $\Sigma B'(y)=0$ for every $y$ and brightness changes due to shear strain were omitted. Hence, each pixel in $\mathbf{A}_k$ should approximately correspond to the pixels in the corresponding real image $\mathrm{im}_k$. For each image $k$, brightness differences of each pixel were computed by subtraction in the hue-saturation-brightness color space: $\mathrm{diff}_k=\mathrm{im}_k-\mathbf{A}_k$. Subimages were cropped from a rectangle bounded by

$$\mathbf{s}_{\mathrm{TL}}=(s^{(k)}_{x,0,0}\quad s^{(0)}_{y,0,0})^{\mathrm{T}},\qquad \mathbf{s}_{\mathrm{BR}}=(s^{(k)}_{x,I-1,J-1}\quad s^{(0)}_{y,I-1,J-1})^{\mathrm{T}}. \tag{9}$$

The resulting brightness difference images were filtered by a $25\times25$ box filter in order to compensate for the alignment errors. Subsequently, all columns were binned together by averaging, resulting in vertical brightness difference vectors, which were filtered by a moving average of 15 pixels to further remove noise.

Because the pixels of the synthetic images were obtained by translating pixels of the undeformed image (and by interpolation), the brightness values of the pixels were almost unchanged. Thus the brightness difference image indicates how much the brightness values of the physical points have changed in the real deformation process. The brightness change is assumed to be due to shear strain, which was considered almost negligible, and to the nonuniformity of illumination.

The pixel displacements are exactly known for the artificially deformed images. Hence, the brightness derivatives can be computed using the longitudinal displacements and the brightness change images. Because the derivatives can be computed using each deformed image, altogether 181 estimates of the derivative are obtained for each row ($B'^{(k)}(y)$). In the first images, both displacements and brightness changes are very small, but on the other hand, alignments are accurate. After a preliminary analysis, derivative estimates based on images {50, 51,…, 100} were included in the determination of the brightness profile. The mean, minimum, and maximum of the selected derivative estimates are shown in Fig. 5.

The brightness profiles are obtained by summing the brightness derivatives:

$$\Sigma B'^{(k)}(y)=\sum_{n=0}^{y}B'^{(k)}(n). \tag{10}$$

In positions where no estimate is available, the derivative is assumed to be 0. Finally, the profile to be used in image generation is obtained by a minimum filter:

$$\Sigma B'(y)=\min_{k}\Sigma B'^{(k)}(y). \tag{11}$$

The resulting brightness profile is shown in Fig. 6. Due to successive filtering the profile is very smooth, but noise could easily be added to the profile to simulate more challenging illumination conditions.

### 3.4 *Geometrical and Tonal Image Transformations*

The strain fields acquired in Sec. 3.1 are transformed to displacements with respect to the image without deformation. The displacements of each image are given by $(I+1)\times(J+1)$ matrices $\mathbf{D}_x,\mathbf{D}_y$. To obtain an artificially deformed image the following procedure is applied to each RGB color channel:

1. $\mathbf{D}_x$ and $\mathbf{D}_y$ are interpolated by cubic B-splines to obtain $x$ and $y$ displacements for every pixel.
2. The pixels of the seed image $\mathrm{im}_0$ are translated according to the interpolated displacement matrices, i.e., the intensities of the pixels remain unchanged, but their locations change. The locations are represented as floating-point numbers, and they usually do not coincide with the original pixel grid. Thus the pixels locate on a nonuniform grid in the resulting intermediate image.
3. The intensities of the translated pixels are interpolated by cubic B-splines.
4. The interpolated intensity B-splines are sampled using the image grid of the seed image. The samples are floating-point numbers.
5. The sampled intensities are quantized to 8 bits.

The resulting image is an artificially deformed image (ADI) without explicit intensity changes. However, interpolation and quantization of intensities cause small changes separately to the RGB channels, which in theory can change the colors. Nevertheless, the effects are assumed to be so small that they may be omitted.

The explicit intensity changes are applied to the artificially deformed images. First, the RGB images are converted to the hue-saturation-brightness (HSB) color space. For each pixel the saturation is modified according to Eq. (4), and the brightness according to Eqs. (5) and (6). The next two subsections deal with two major noise types (motion blur and Gaussian pixel noise) that are assumed to be the dominant error sources.

### 3.5 *Motion Blur*

Vibration of the camera and defocus of optics are considered major sources of noise in tensile test imaging. A major source of vibration is the tensile test machine itself. Defocus is practically always present, but the object distance may also change a little during a tensile test, due to the deformation of the object.

The camera motion is assumed to be nondirectional sinusoidal translational vibration. Wang et al. have derived a mathematical model for the blurring effects of such a camera motion.[37] They used temporal integration to create synthetic motion blurred test images to test their algorithm that extracts the vibration parameters. Myles and da Vitoria Lobo have in turn modeled the effects of small defocus by Gaussian blur and higher levels of defocus by convolution of a pill-box function.[38] For simplicity, both defocus and motion blur are now modeled by convolution of the pillbox function defined as

$$\mathrm{Pillbox}(\mathbf{r},R)=\begin{cases}1/\pi R^2, & |\mathbf{r}|\leqslant R,\\ 0 & \text{otherwise}.\end{cases} \tag{12}$$

The amplitude of vibration (denoted by $R$) may vary during a tensile test. The vibration amplitude of each image is assumed to be uniformly and independently distributed with a maximum amplitude of $R_{\max}$:

$$R^{(k)} \sim \mathrm{uid}(0,R_{\max}), \tag{13}$$

where $k$ is the image index. Because the effect of defocus is similar to motion blur, it is not separately modeled. Directional vibrations may also occur in imaging, but that is out of the scope of this study. In future, it could be modeled using, e.g., the methodology presented by Wang et al.[37]

### 3.6 *Gaussian Pixel Noise*

Colored pixel noise is added after the application of motion blurring. Pixel noise is assumed to be normally and independently distributed:

$$\nu_{m,n,c}^{(k)} \sim \mathrm{nid}(0,\sigma^2), \tag{14}$$

where $m$ and $n$ are spatial pixel coordinates, $c$ is the color band, and $\sigma^2$ is the variance of the pixel noise.

Pixel noise is applied independently to the RGB bands, i.e., the noise is colored. Pixel noise is additive, which means that the resulting image $f=g+\nu$, where $g$ is the artificially deformed, motion-blurred image. With blur and pixel noise the resulting SNR of the image is easily controlled because the signal level is constant.

## 4 Experiments and Results

A spray-painted random-speckle uniaxial dogbone tensile test specimen was stretched and imaged during deformation. The acquired image series consisted of 182 images. The mean $x$ and $y$ strains of the last image were −0.041 and 0.16, respectively. The image series was used to obtain realistic strain paths, coefficients for the saturation and brightness changes with respect to local strain, and the longitudinal brightness profile.

The first image of the series was used as the seed image of the artificially deformed images. Hence the quality of artificially deformed images could be easily verified by comparing them with the real deformed images. Other seeds were also used, even one without any painting, and the results were similar. Therefore all results given here are from the seed image mentioned first.

In the test runs, true strain measurement errors can be computed, because the ground-truth strains are exactly known. The measured strains $\tilde{\varepsilon}_x(i,j)$ and $\tilde{\varepsilon}_y(i,j)$ of a 3 $\times 7$ strain sampling grid (i.e., four transverse and eight longitudinal nodes of templates) are compared with the true strains $\varepsilon_x$ and $\varepsilon_y$ at the corresponding points $\mathbf{s}(i,j)$. An overall value of the true $x$ strain error (and similarly $y$ and $z$) is obtained as an average of the absolute errors:

$$\delta\varepsilon_x=\frac{\sum_{i=0}^{I-1}\sum_{j=0}^{J-1}|\varepsilon_x(\mathbf{s}(i,j))-\tilde{\varepsilon}_x(i,j)|}{IJ}. \tag{15}$$

Assuming that the $x$ and $y$ strain errors are independent (orthogonal), they can be summed to an overall figure of strain error $\delta\varepsilon$ (total strain error), which is used in many analyses instead of using $x$, $y$, and $z$ errors separately:

$$\delta\varepsilon=(\delta\varepsilon_x^2+\delta\varepsilon_y^2)^{1/2}. \tag{16}$$

In order to estimate the deviation of the true strain measurement errors, each measurement was repeated five times. The strain sampling grid was translated vertically by 30 pixels between the measurements. Hence, both the underlying signal (i.e., the noise-free image) and the specific noise pattern were different in each measurement.

The following tests were done: The true errors were studied with respect to basic internal algorithm parameters, such as template size ($Z_{xy}$), update interval ($\Delta r$), coarse-search resolution ($r_i$), and fine-search template size ($Z_{xy}+Z_s$). Motion blur and pixel noise levels were varied in order to find out the sensitivity of the algorithm accuracy to the changes in SNR. The efficiency of the dynamic template size control method and parameter optimization strategy was tested with varying noise levels. Finally, the implicit strain error estimates were compared with the true strain errors. Unless otherwise mentioned, the most important internal parameters were as follows: $Z_{xy}=33$, $\Delta r=2$, $r_i=2$, $Z_s=0$, $\Delta k=21$ (image update interval[31]), number of repeats=3, $T_\sigma=8$, $T_{\mathrm{DG}}=0.1$, $Z_i=10$, and $R_i=3$. The last five parameters are related to the dynamic template size control.[32] Their explanations are, respectively: maximum number of template growths, threshold for standard deviation of intensities of the template, threshold for *discrimination gap* (DG; i.e., the difference of the highest two correlation peaks), template size growth (pixels), and ROI size growth (pixels).

### 4.1 *Artificially Deformed Images*

The first test was to generate images using the deformation model introduced in Sec. 3 and to compare the synthetic images with the true deformed ones. Figure 7 shows three true deformed images [(a) to (c)] and three artificially deformed images without added noise [(d) to (f)]. Images (g) to (i) are obtained from images (d) to (f) by adding motion

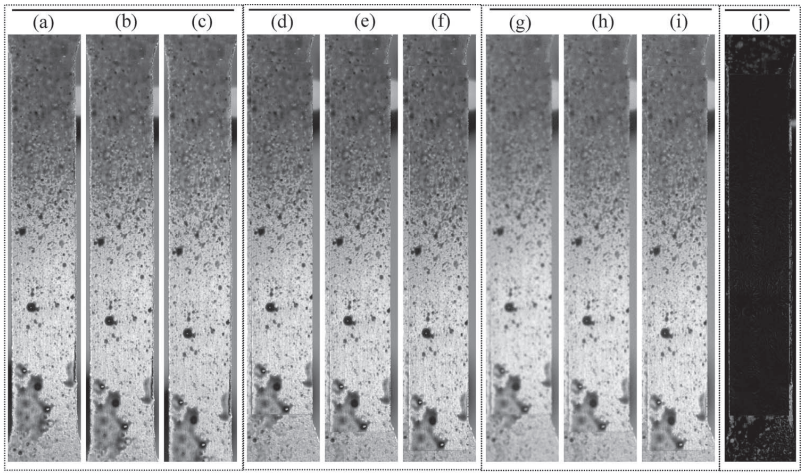



**Fig. 7** Three true deformed images [(a) to (c)], three artificially deformed images (ADIs) without added noise [(d) to (f)], and three ADIs with $R_{max}=30$ and $\sigma^2=30$ [(g) to (i)]. The rectangular area that is artificially deformed is visible in images (d) to (i). Note that the background is unchanged. The rightmost image (j) is the absolute pixelwise difference between images (a) and (d).

blur and pixel noise. The rightmost image (j) shows the absolute difference of grayscale intensities between images (a) and (d).

The sample images in Fig. 7 show that the artificially deformed images greatly resemble the real ones. The difference image verifies that the morphing and pixel value modifications were such that the resulting artificially deformed image closely matches the real one. It can be concluded that the noiseless ADIs are thus quite realistic. The added motion blur and pixel noise are used to change the signal-to-noise ratio. Images (d) to (i) show, as expected, that the recognition of small details becomes more difficult on decreasing SNR. However, it has not been studied how realistic the noise model is.

### 4.2 *Effects of Template Size, Update Interval, and the Coarse-to-Fine Search Strategy*

In this subsection, the effects of some algorithm parameters on the strain error are studied. The effect of template size and update interval have been previously studied in Ref. 31, but now it was possible to obtain true errors as well as the estimates. The SNR was varied when the effect of template size, perhaps the most important parameter, was studied. In addition, the effects of three other parameters, namely the correlation sampling resolution, subpixel sampling radius, and subpixel correlation sampling template size, were studied.

The objectives of the heuristics of the algorithm are to accelerate the algorithm without any significant increase in error. Now only errors are studied empirically, but Sec. 2 briefly discussed the complexity issues, too. The test runs in this section do not utilize dynamic template size control or optimization by genetic algorithms, i.e., the algorithm parameters are fixed during each run (excluding the parameters whose effect is studied). Parameters also have interactions, but multidimensional tests are mostly omitted in this study.

The effect of template size is shown in Fig. 8, where three different combinations of noise are used. The results indicate that for a certain noise level, there is a breakdown value of the template size, below which the errors increase rapidly. The breakdown value seems to increase monotonically on increasing the noise level. On the other hand, increasing the template size from the breakdown value does not always significantly decrease strain measurement errors. Because the SNR varies spatially in real images (e.g., due to the varying speckle pattern), local estimation of the SNR gives a good opportunity to dynamically control the template size.

Figure 9 shows the effect of the update interval $\Delta k$ on the strain error when no noise is added. The *update interval* is the number of images after which the intensity values of

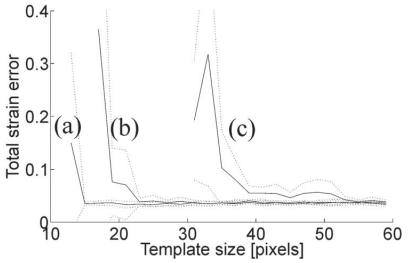



**Fig. 8** Total strain error versus template size for three different combinations of motion blur and noise: (a) no motion blur, no added noise, (b) motion blur=2.5 pixels, Gaussian noise=20, and (c) motion blur=5.5 pixels, Gaussian noise=30.

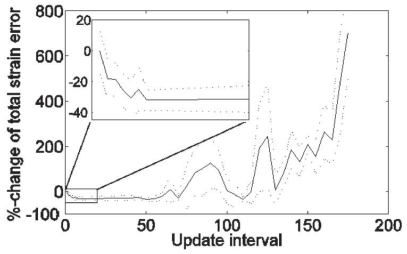



**Fig. 9** The relative effect of $\Delta k$ on the total strain error. The base level is the error when the update interval is 1. The inset shows a magnified snapshot from update intervals 0 to 20.

the templates are resampled. Each resampling causes error accumulation, but, on the other hand, without resampling the shape distortions may grow too large and inhibit pattern recognition.

Due to error accumulation, increasing $\Delta k$ from 1 to 10 decreases the strain error by approximately 30%. Any further increase of $\Delta k$ does not seem to decrease errors significantly. Large update intervals, in contrast, lead to very large errors. Another observation is that the error landscape is multimodal with many local minima. This implies that local optimization methods might be inefficient in such a landscape and the use of genetic algorithms is now well grounded.

The principle of the coarse-to-fine search strategy was introduced in Sec. 2.2. In its first step, correlations were sampled using a resolution of $\Delta r$ pixels. Large $\Delta r$ speeds up the search significantly, but the probability of mismatch increases correspondingly. (see Fig. 10.)

With $\Delta r=1$ the strain error was the smallest, as expected. It can be seen that when $\Delta r=6$ the standard deviation of the error increases significantly. Sampling intervals from 0 to 5 lead to approximately the same errors if error margins (dotted lines) are included in the analysis. Because the transition from 1 to 2 gives most of the complexity reduction, larger steps are not recommended unless computation speed is crucial for the application. The landscape is again nonmonotonic and noisy, which suggests that advanced optimization methods such as genetic algorithms are needed when optimizing parameters.

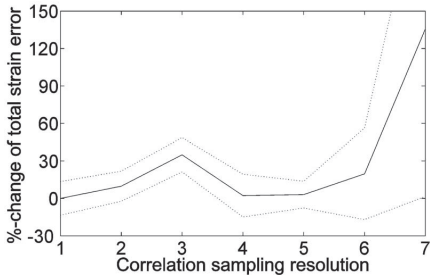



**Fig. 10** The relative effect of $\Delta r$ on the strain error. The mean error for $\Delta r=1$ is used as the base level, and the relative differences with respect to it are given. Dotted lines indicate the confidence interval of one standard deviation.

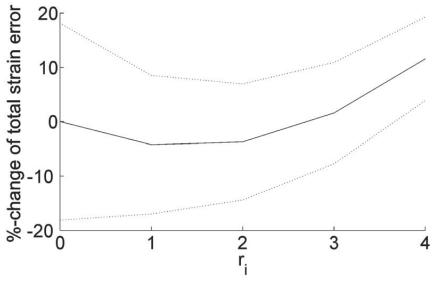



**Fig. 11** The relative effect of the subpixel interpolation radius $r_i$ on the strain error. The error when $r_i=0$ (i.e., positioning without subpixel interpolation) is used as the reference value.

Subpixel pattern localization is applied to improve accuracy. Correlation samples around the pixel of the maximum correlation are used to fit a parabola, whose peak is regarded as the subpixel position of the pattern. The radius of the neighborhood ($r_i$) that is used in the interpolation affects the localization and thus the strain measurement error. If $r_i=1$, a $3\times3$ neighborhood is used.

In Fig. 11, strain error without subpixel correlation interpolation ($r_i=0$) is used as the base level. When $r_i \in [1,2]$ the mean error is slightly less than without correlation interpolation. Hence, it is probable that subpixel localization improves the positioning accuracy, but again the error margins are wide.

Another attempt to reduce complexity and increase accuracy was to use a smaller template in the coarse search phase and a larger one in the subpixel interpolation phase. The amount of template growth between these phases is denoted by $Z_s$. Figure 12 shows how much the error was reduced when the template size was grown from 32 pixels by 2, 4, and 6 pixels. The results do not give a clear image of the influence of the dual template size strategy, for the coarse-phase template size and $Z_s$ obviously have interactions and computational complexity should also be included in the analysis. Nevertheless, it can be concluded that subpixel localization with a larger template probably decreases the strain measurement error, and the corresponding increase in complexity is relatively small, because only a few correlations are resampled.

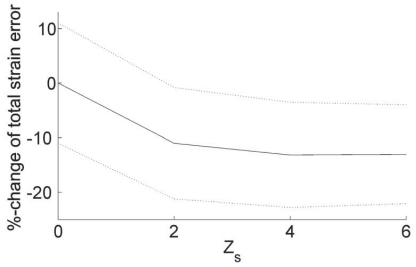



**Fig. 12** The relative effect of $Z_s$ on the strain error. The error when $Z_s=0$ is the base level.

**Table 1** The ranges of the objective parameters in GA optimization.

| Parameter | Range |
|---|---|
| $Z_{xy}$ | {9, 11, …, 37} |
| $Z_s$ | {0, 2, …, 28} |
| $\Delta r$ | {1, 2, 3} |
| $r_i$ | {1, 2, 3} |
| $\Delta k$ | {5, 6, …, 50} |
| No. of repeats | {0, 1, …, 4} |
| $T_\sigma$ | [0, 20] |
| $T_{DG}$ | [0, 0.4] |
| $Z_i$ | {4, 6, …, 16} |
| $R_i$ | {0,1, …, 4} |

### 4.3 *Efficiency of Dynamic Template Size Control and Optimization*

In this subsection, dynamic template size control (DTSC) and parameter optimization by genetic algorithms are used. Their efficiency in comparison with the static parameter setup is studied.

The results in Fig. 8 indicated that the template size has a breakdown level that depends on the noise level. Moreover, enlarging the template does not significantly reduce errors above the breakdown level. Therefore, it is presumed that DTSC is beneficial particularly in conditions where the SNR varies spatially, whence the template size control should be able to grow the template above the breakdown level if necessary.

When DTSC was tested, two noise parameters, blur radius and pixel noise intensity, were varied while the internal algorithm parameters were kept constant from run to run. DTSC was dynamically controlling template and ROI size parameters during the strain measurements.

DTSC is often able to improve accuracy without significant increase in computation time. However, DTSC controls only the template and ROI size parameters, not the other internal parameters. It was seen that, e.g., the error landscape of the update interval was multimodal and nonlinear. Therefore, optimization of the internal parameters is presumed to improve accuracy significantly in some cases.

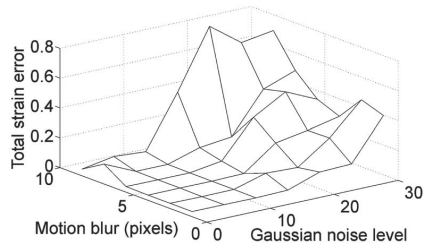



**Fig. 13** Total strain errors with different levels of motion blur and noise. Fixed template size.

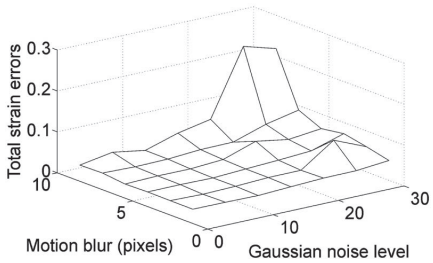



**Fig. 14** Total strain errors with different levels of motion blur and noise. Dynamic template size control activated.

In GA optimization, parameters of the cost function were: $k_{acc}=226$ (226 instead of 281 images were used), $a=4\times10^{-4}$, and $b=0.9$ ($a$ and $b$ are used to balance between minimizing error and computational complexity[33]). The internal GA parameters were $p=8$ (population size) and $n=20$ (number of iterations).

The initial population was randomly created using the ranges of the objective parameters given in Table 1. It can be seen that the search space is vast and only 20 cost function evaluations were available. Nonetheless, usually the optimization result was good enough and even better than what the plain DTSC gave.

Figures 13, 14, and 17 show how noise affected strain measurement accuracy when the template size was constant, when it was controlled by DTSC, and when parameters were optimized by GA, respectively. The conclusions from Fig. 13 are that both pixel noise and motion blur increase the strain measurement error a little until the breakdown level is reached. Moreover, pixel noise and motion blur together give a greater increase in the error than separately.

Because the pattern localization accuracies in the $x$ and $y$ directions should be equal in an ideal image, the ratio of the $y$ and $x$ strain measurement errors should be inversely proportional to the corresponding strain sampling resolutions (in pixels). Without motion blur or Gaussian noise, the ratio of the $y$ to the $x$ strain error was 0.74. The theoretical value based on the strain sampling resolutions was

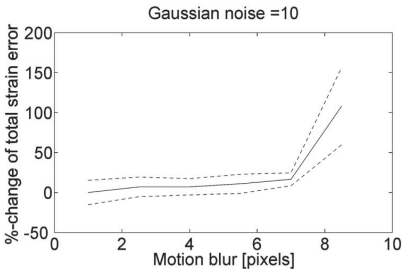



**Fig. 15** Relative change of total strain errors with respect to motion blur, when Gaussian noise was fixed at 10. Dynamic template size control was not activated.

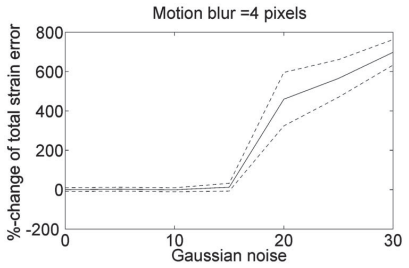



**Fig. 16** Relative change of total strain errors with respect to Gaussian noise, when motion blur was fixed at 4 pixels. Dynamic template size control was not activated.

0.61. Thus the longitudinal strain errors relative to the transverse strain errors were slightly greater than expected.

If Fig. 14 is compared with Fig. 13, it can be concluded that the dynamic template size control is able to push the breakdown noise level further. Without motion blur or Gaussian noise, the ratio of the $y$ to the $x$ strain error was 0.74. The theoretical value based on the strain sampling resolution is 0.61. Thus $y$ strain errors were again in relative terms a bit greater than expected.

In order to see more closely how the error behaves below the beakdown level, two cross-sections of errors are given in Figs. 15 and 16. Figure 15 shows errors with respect to motion blur, when the pixel noise intensity was fixed at 10. Figure 16 shows errors against pixel noise, when the motion blur was set at 4 pixels. The results indicate that on increasing the motion blur, the strain error increases a few percent until the breakdown level. Pixel noise, on the other hand, does not seem to affect pattern localization implemented using cross-correlation until total mismatch occurs at the breakdown noise level.

These observations are in line with the results reported by Robinson and Milanfar,[30] who studied fundamental performance limits in image registration. They observed a gradual increase in error as a function of image bandwidth, whereas a flat error landscape with a breakdown level was detected when the SNR was varied and a direct correlator method was used.

Figure 17 indicates that accelerated optimization by genetic algorithms is able to reduce strain errors in noisy conditions. On the other hand, accelerated optimization with

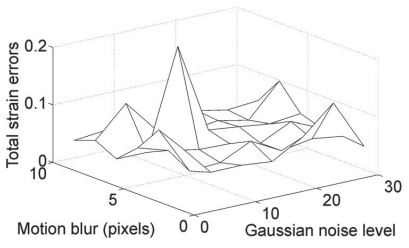



**Fig. 17** Total strain errors with different levels of motion blur and noise. Parameters optimized by GA.

only 20 cost-function evaluations was unstable; large errors sometimes occurred even in good SNR conditions.

Without motion blur or Gaussian noise, the ratio of the $y$ to the $x$ strain error was 2.0. The theoretical value based on the strain sampling resolution was 0.61, i.e., GA tends to minimize $x$ strain, more than $y$ strain, for an unknown reason.

In order to evaluate the advantage of DTSC and GA optimization, differences of strain errors between the three algorithm setups (i.e., error reductions) were computed. The histograms of the differences are shown in Figs. 18–21. Figures 18 and 20 show the histograms of all noise-level combinations, while Figs. 19 and 21 use only the 16 test runs with the highest noise levels. It was found that DTSC and GA optimization gave more advantage at high noise levels.

Figures 18 and 19 show that the DTSC strategy reduced the strain error by 0.13 on average and by 0.20 for the noisiest images, with respect to the constant size template strategy. Furthermore, the histograms show that large error increases are rare while large error reductions are quite frequent.

Figures 20 and 21 compare GA with DTSC in terms of error reduction. Now it is evident that no error reduction occurs in the mean sense, mostly due to the instability of the accelerated optimization. However, GA optimization was able to reduce strain measurement errors significantly in some noisy cases. As a conclusion, GA optimization should be used if the default parameters give inferior results.

### 4.4 *Accuracy of the Implicit Error Estimation*

In our previous studies,[31–33] strain measurement errors were evaluated implicitly from strain deviations. Now it is possible to evaluate how accurate the implicit error estimates are and subsequently to find out whether the conclusions of the previous papers were valid. Figure 22 shows the estimated $x$ and $y$ strain errors against the corresponding true errors obtained from one test run. Spearman's rank-order correlation between the estimated and true errors for $x$ and $y$ strains was 0.93 and 0.87, respectively. As a conclusion, usually the error estimates are in correct order. A quite linear relationship is also visible in Fig. 22. However, longitudinal strain error estimates are almost consistently larger than the transverse ones.

When error estimates are compared with the true values in the low strain region, the situation changes significantly (Fig. 23). With low strains the order of the estimated strains is often incorrect, and, particularly, $x$ strain error estimates seem to provide insufficient information on the true errors.

Error estimates are used in the cost function when optimizing the algorithm parameters by the genetic algorithm. It is obvious that the estimates can be used in coarse optimization, but if the exact minimum of error is looked for, the error estimates may be less reliable.

## 5 Conclusions

A method to measure strains using digital speckle correlation and a method to test such algorithms by artificially deformed images were introduced. The proposed measurement algorithm includes heuristics, including dynamic template size control, to improve the accuracy-complexity ra-

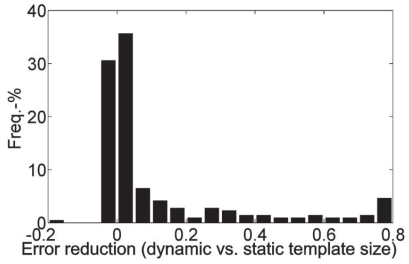



**Fig. 18** Distribution of total strain error reduction (positive values) and increase (negative values) when constant template size was replaced by dynamic template size control. The mean reduction was 0.13, i.e. a strain of 13%.

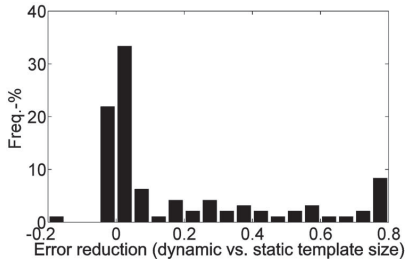



**Fig. 19** Error reduction distribution of 16 noisy image series when the fixed template size was replaced by dynamic template size control. The mean reduction was 0.20.

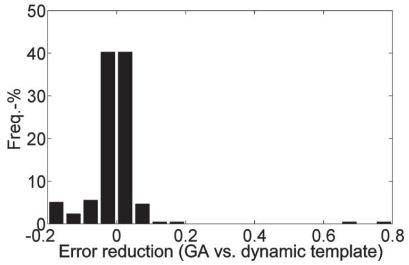



**Fig. 20** Error reduction distribution, when DTSC with fixed parameters was replaced by GA-optimized parameters. The mean error reduction was –0.009, i.e., errors increased on average.

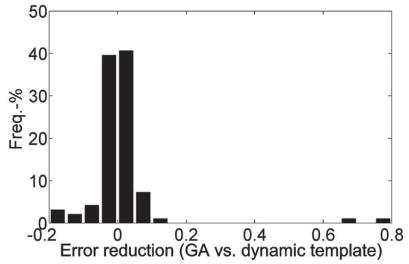



**Fig. 21** Error reduction distribution of the 16 noisy image series when DTSC with fixed parameters was replaced by GA-optimized parameters. The mean error reduction was 0.008.

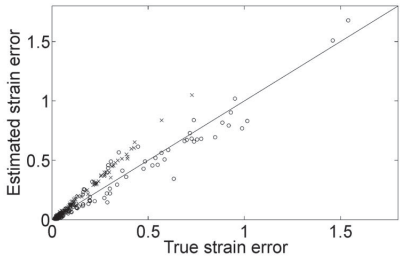



**Fig. 22** Correlation between the estimated and true strain errors is strong in the global scope. ×, *x* strain error; ○, *y* strain error.

tio. An optimization approach based on a genetic algorithm is also used in order to quickly find suboptimal internal parameters of the strain measurement algorithm. Optimization of the algorithm parameters is considered to improve the usability of the software, because multiple parameters should be adjusted to obtain reliable results.

Strain measurement based on images incorporates nonrigid-body registration. It is difficult to obtain reference images with an exactly known deformation field if the objects are subject to deformation. Synthetic test images solve this problem.

A method to generate realistic synthetic images that resemble images from a real uniaxial tensile test was proposed in this paper. The image generation process included: acquisition of strain fields from a real image series, determination of saturation and brightness change models with respect to local shear strain, determination of the spatial illumination distribution of a real image series, artificial image deformation using a real image as a seed, modeling of translational sinusoidal vibration by blurring, and application of additive colored pixel noise.

The results show that template size has a breakdown level, below which measurement errors increase rapidly due to mismatches of pattern localization. Moreover, the breakdown level depends on the noise and blur levels. On the other hand, template size and noise parameters have only a little influence on accuracy, when operating above the breakdown level. These results agree with the results obtained by Robinson and Milanfar.[30] The normalized correlation that was used in pattern localization is tolerant to noise and changes in lighting conditions. Results showed that correlation sampling with subpixel interpolation improved the accuracy; an optimal size of the interpolation

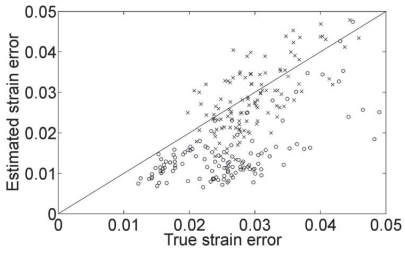



**Fig. 23** Estimated versus true strain errors in the low strain region.

neighborhood is around $3\times3$ to $5\times5$ pixels. However, the correlation sampling method is known to be biased, and other localization algorithms might be more accurate;[30] thus a comparison should be done.

Dynamic template size control managed to reduce errors significantly in comparison with using a fixed template size. The optimization scheme based on a genetic algorithm was considered useful in cases where *ad hoc* parameter setups turned out to be inferior. However, the accelerated optimization was somewhat unstable, sometimes causing large errors even in low-noise conditions. Because the fitness landscapes of some internal parameters were found to be nonlinear and multimodal, and the search space of the optimization is multidimensional and large, it is anyway justified to use heuristic global optimization methods such as genetic algorithms for parameter optimization.

## Acknowledgments

The Finnish Funding Agency for Technology and Innovation (TEKES) and the industrial partners of the research project Process Development for Incremental Sheet Forming have financially supported this research. Jukka Tuomi, Seppo Kivijärvi, Lotta Vihtonen, Tuomas Katajarinne, Timo Mantere, Olli Kanniainen, Timo Alho, and Annette Lönnqvist are acknowledged as collaborators.

## References

1. B. Zitove and J. Flusser, "Image registration methods: a survey," *Image Vis. Comput.* **21**, 977–1000 (2003).
2. J. V. Hajnal, D. L. G. Hill, and D. J. Hawkes, Eds., *Medical Image Registration*, CRC Press, Boca Raton, FL (2001).
3. S. R. McNeill, W. H. Peters, W. F. Ranson, and M. A. Sutton, "A study of fracture parameters by digital image processing," in *Proc. 18th Midwest Mechanics Conf.*, Vol. **12**, pp. 267–271 (1983).
4. T. C. Chu, W. F. Ranson, M. A. Sutton, and W. H. Peters, "Applications of digital-image-correlation techniques to experimental mechanics," *Exp. Mech.* **25**(3), 232–244 (1985).
5. G. Vendroux and W. G. Knauss, "Submicron deformation field measurements: part 2. Improved digital image correlation," *Exp. Mech.* **38**(2), 86–92 (1998).
6. H. Lu and P. D. Cary, "Deformation measurements by digital image correlation implementation of a second-order displacement gradient," *Exp. Mech.* **40**(4), 393–399 (2000).
7. D. Amodio, G. B. Broggiato, F. Campana, and G. M. Newaz, "Digital speckle correlation for strain measurement by image analysis," *Exp. Mech.* **43**(4), 396–402 (2003).
8. T. Schmidt, J. Tyson, and K. Galanulis, "Full-field dynamic displacement and strain measurement using advanced 3D image correlation photogrammetry," *Exp. Tech.* **27**(3), 47–50 (2003).
9. P.-C. Hung and A. Voloshin, "In-plane strain measurement by digital image correlation," *J. Braz. Soc. Mech. Sci.* **25**(3), 215–221 (2003).
10. C. Su and L. Anand, "A new digital image correlation algorithm for whole-field displacement measurement," in *Innovation in Manufacturing Systems and Technology, Proc. Singapore-MIT Alliance Symp.* (2003).
11. Gesellschaft für Optische Messtechnik, http://www.gom.com (cited May 12, 2008).
12. ViALUX Messtechnik + Bildverarbeitung GmbH, http://www.vialux.de/ (cited: May 12, 2008).
13. R. Mosqueron, J. Dubois, and M. Paindavoine, "High-speed smart camera with high resolution," *J. Embedded Systems* **2007**(24163), 1–16 (2007).
14. D. Garcia, J. J. Orteu, and L. Penazzi, "A combined temporal tracking and stereo-correlation technique for accurate measurement of 3D displacements: application to sheet metal forming," *J. Mater. Process. Technol.* **125–126**, 736–742 (2002).
15. M. Abbadi, F. de Haan, and P. Hähner, "Tensile strain measurement using mechanical and in situ optical techniques in bi-metallic Ni-base superalloys," *Measurement* **40**, 383–391 (2006).
16. J. Banhart and J. Baumeister, "Deformation characteristics of metal foams," *J. Mater. Sci.* **33**, 1431–1440 (1998).
17. E. Amsterdam, P. R. Onck, and J. T. M. de Hosson, "Fracture and microstructure of open cell aluminum foam," *J. Mater. Sci.* **40**, 5813–5819 (2005).
18. H. Lu, H. Shi, and M. Zhou, "Thermally induced deformation of solder joints in real packages: measurement and analysis," *Microelectron. Reliab.* **46**, 1148–1159 (2006).
19. J.-N. Périé, S. Calloch, C. Cluzel, and F. Hild, "Analysis of a multiaxial test on a C/C composite by using digital image correlation and a damage model," *Exp. Mech.* **42**(3), 318–328 (2002).
20. S. V. Lomov, P. Boisse, E. Deluycker, F. Morestin, K. Vanclooster, D. Vandepitte, I. Verpoest, and A. Willems, "Full-field strain measurements in textile deformability studies," *Composites, Part A* **39**(8), 1232–1244 (2008).
21. Y.-W. Cheng, D. T. Read, J. D. McColskey, and J. E. Wright, "A tensile-testing technique for micrometer-sized free-standing thin films," *Thin Solid Films* **484**, 426–432 (2005).
22. S. Bergonnier, F. Hild, and S. Roux, "Digital image correlation used for mechanical tests on crimped glass wool samples," *J. Strain Anal. Eng. Des.* **40**(2), 185–197 (2005).
23. S. Samarasinghe, D. Kulasiri, and K. Nicolle, "Study of mode-I and mixed-mode fracture in wood using digital image correlation method," in *Proc. Int. Wood Engineering Conf.* (1997).
24. G. Sinn, A. Reiterer, S. E. Stanzl-Tschegg, and E. K. Tschegg, "Determination of strains of thin wood samples using videoextensometry," *Holz Roh-Werkst.* **59**, 177–182 (2001).
25. J. Tyson, T. Schmidt, and K. Galanulis, "Biomechanics deformation and strain measurements with 3D image correlation photogrammetry," in *Biomechanics Series: part 3, Experimental Techniques*, pp. 39–42, Soc. for Experimental Mechanics (2002).
26. H. Marcellier, P. Vescobo, D. Varchon, P. Vacher, and P. Humbert, "Optical analysis of displacement and strain fields on human skin," *Skin Res. Technol.* **7**(4), 246–253 (2001).
27. T. Mantere and J. T. Alander, "Testing a structural light vision software by genetic algorithms—estimating the worst case behavior of volume measurements," in *Intelligent Robots and Computer Vision XX: Algorithms, Techniques, and Active Vision*, D. P. Casasent and E. L. Hall, Eds., *Proc. SPIE* **4572**, 466–475 (2001).
28. T. Mantere and J. T. Alander, "Automatic image generation by genetic algorithms for testing halftoning methods," in *Intelligent Robots and Computer Vision XIX: Algorithms, Techniques, and Active Vision*, D. P. Casasent and E. L. Hall, Eds., *Proc. SPIE* **4197**, 297–308 (2001).
29. N. P. Castellanos, P. L. D. Angel, and V. Medina, "Nonrigid medical image registration technique as a composition of local warpings," *Pattern Recogn.* **37**, 2141–2154 (2004).
30. D. Robinson and P. Milanfar, "Fundamental performance limits in image registration," *IEEE Trans. Image Process.* **13**(9), 1185–1199 (2004).
31. J. Koljonen, O. Kanniainen, and J. T. Alander, "An implicit validation approach for digital image correlation based strain measurements," in *Proc. IEEE Int. Conf. on the Computer as a Tool*, pp. 250–257 (2007).
32. J. Koljonen, O. Kanniainen, and J. T. Alander, "Dynamic template size control in digital image correlation based strain measurements," in *Intelligent Robots and Computer Vision XXV: Algorithms, Techniques, and Active Vision*, D. P. Casasent, E. L. Hall, and J. Röning, Eds., *Proc. SPIE* **6764**, 67640L (2007).
33. J. Koljonen, T. Mantere, and J. T. Alander, "Parameter optimization of numerical methods using accelerated estimation of cost function: a case study," in *Proc. Finnish Signal Processing Symp.*, M. Niskanen and J. Heikkilä, Eds. (2007).
34. J. Koljonen, T. Mantere, O. Kanniainen, and J. T. Alander, "Searching strain field parameters by genetic algorithms," in *Intelligent Robots and Computer Vision XXV: Algorithms, Techniques, and Active Vision*, D. P. Casasent, E. L. Hall, and J. Röning, Eds., *Proc. SPIE* **6764**, 67640O (2007).
35. J. Koljonen, T. Katajarinne, A. Lönnqvist, and J. T. Alander, "Validation of digital speckle correlation strain measurements with extensometer," in *Proc. Int. Conf. International Deep Drawing Research Group (IDDRG 2008)* (2008). CD-ROM.
36. S. Forrest, "Genetic algorithms: principles of natural selection applied to computation," *Science* **261**(5123), 872–878 (1993).
37. S. Wang, B. Guan, G. Wang, and Q. Li, "Measurement of sinusoidal vibration from motion blurred images," *Pattern Recogn. Lett.* **28**, 1029–1040 (2007).
38. Z. Myles and N. da Vitoria Lobo, "Recovering affine motion and defocus blur simultaneously," *IEEE Trans. Pattern Anal. Mach. Intell.* **20**(6), 652–658 (1998).

**Janne Koljonen** received his MSc degree in automation technology at the University of Vaasa, Finland, in 2004. He is currently working as a researcher and lecturer at the University of Vaasa. His main research interests include machine vision, optimization using genetic algorithms, near-infrared spectroscopy, and signal processing and automation applications in general. He is the author or coauthor of about 30 scientific papers.

**Jarmo T. Alander** received the MSc degree in physics at Helsinki University, Finland, in 1983; the MSc (Tech) degree in computer science, the Licentiate of Technology degree in computer science, and the DrTech degree in computer science and engineering at Helsinki University of Technology in 1985, 1989, and 2001, respectively; and the Licentiate of Philosophy degree in physics from Helsinki University in 1999. He has been a full professor of production automation at the University of Vaasa, Finland, since 1993. His main research interests include optimization using genetic algorithms, signal processing, automation, and computing in general. He is the author or coauthor of one book and about 200 scientific papers.