Geometric and Photometric Affine Invariant Image Registration

Ángel Cuesta Contreras

Thesis submitted for the degree of Doctor of Philosophy

Heriot-Watt University

School of Engineering and Physical Sciences

May 2009

Abstract

This thesis aims to present a solution to the correspondence problem for the registration of wide-baseline images taken from uncalibrated cameras. We propose an affine invariant descriptor that combines the geometry and photometry of the scene to find correspondences between both views. The geometric affine invariant component of the descriptor is based on the affine arc-length metric, whereas the photometry is analysed by invariant colour moments. A graph structure represents the spatial distribution of the primitive features; *i.e.* nodes correspond to detected high-curvature points, whereas arcs represent connectivities by extracted contours. After matching, we refine the search for correspondences by using a maximum likelihood robust algorithm. We have evaluated the system over synthetic and real data. The method is endemic to propagation of errors introduced by approximations in the system.

To my parents,

Acknowledgements

I would like to thank my supervisor Professor Andrew Wallace for his guidelines and willingness to help with my research throughout these years. I take the opportunity to thank Dr. Yvan Petillot for the interesting viewpoints and ideas about my work. My gratitude to Professor Manuel Trucco for his constructive insights. I am also thankful to BAE Systems and Selex Sensors & Airborne Systems for funding this work.

All my appreciation to my friends and colleagues Sergio, Arvind and Xun; and also Pierre-Yves and Matt. Always around whenever was needed, no matter for work or just for a laugh. You made the day by day in the lab much easier. It's been a pleasure.

A big part of this thesis is especially dedicated to my best friends Quique, Sergio again, Beatriz, Rui and Marion. For all the good times we had and for being always there… I have no words to express my gratitude for all what you've done for me these years. The beautiful city of Edinburgh would have never been the same without you. I won't forget either Lucía, Patricia and all the wonderful people I had the chance to meet during these years and can't carry on naming.

And my most special dedication goes to my parents and my brother Javier. For all the love, support, understanding and sacrifice from the distance. I know how difficult it has been. There would have been no way to do that without you. This is for you.

ACADEMIC REGISTRY
Research Thesis Submission

Name*:*			
School/PGI:			
Version: *(i.e. First, Resubmission, Final)*		Degree Sought (Award **and** Subject area)	

Declaration

In accordance with the appropriate regulations I hereby submit my thesis and I declare that:

1) the thesis embodies the results of my own work and has been composed by myself
2) where appropriate, I have made acknowledgement of the work of others and have made reference to work carried out in collaboration with other persons
3) the thesis is the correct version of the thesis for submission and is the same version as any electronic versions submitted*.
4) my thesis for the award referred to, deposited in the Heriot-Watt University Library, should be made available for loan or photocopying and be available via the Institutional Repository, subject to such conditions as the Librarian may require
5) I understand that as a student of the University I am required to abide by the Regulations of the University and to conform to its discipline.

* *Please note that it is the responsibility of the candidate to ensure that the correct version of the thesis is submitted.*

Signature of Candidate*:*		Date:	

Submission

Submitted By *(name in capitals):*	
Signature of Individual Submitting:	
Date Submitted:	

For Completion in Academic Registry

Received in the Academic Registry by *(name in capitals):*			
1.1 Method of Submission *(Handed in to Academic Registry; posted through internal/external mail):*			
1.2 E-thesis Submitted (**mandatory for final theses from January 2009)**			
Signature:		Date:	

TABLE OF CONTENTS

Chapter 1 – Introduction

1.1 Background

In the registration of images taken from different points of views with uncalibrated cameras (no information on the camera parameters), there are two principal areas of interest: *narrow-baseline* registration for small separation between viewpoints and *wide-baseline* registration for broad angles between camera locations. The narrow-baseline case is very similar to the binocular human vision. Both views are similar and correspondences in both images can be even detected along a single spatial dimension in certain instances. The complexity of the geometric transformation between the images is lessened and consequently smaller degrees of occlusion occur. However, the narrower the angle between the sources the less accurate to recover depth. Wide-baseline registration is the subject of investigation in this thesis. The registration of images when the two cameras are wide apart can result in strong geometric and photometric differences that make the solution to the correspondence problem much harder. Therefore, that implies coping with scenarios where there are considerable translations between the camera centres, rotations of the cameras including rotations of the image about the principal axes of the camera and significant changes in the intrinsic camera parameters (*i.e.* focal length, location of the image centre in the image, effective size of the pixel and coefficient of distortion) [89]. The case of different types of cameras can introduce a different presence of noise added during the acquisition process, the previous changing geometric conditions and possibly frames taken at different times, produce a variation of the illumination conditions for quite disparate views. Moreover, several pixels in one image may match one single pixel in the other image as a result of different scales in wide-baseline situations. No doubt, the wide-baseline case implies greater difficulty for optimal registration, due to these difficulties in solving the correspondence problem. Both views may have fewer common elements and hence partial occlusions and depth discontinuities are more likely to occur. Therefore, image deformations cannot be approximated by simple transformations. In contrast to the narrow-baseline case, wide-baseline registration provides a much less uncertain recovery of the 3D scene.

The work in this thesis is concerned with the registration of 2D stereo images from uncalibrated cameras for wide-baseline scenarios. The setting can be indoor or outdoor visible images, containing man-made objects or natural scenes. As a by-product of not having knowledge of the nature of the scene, the projective transformation that can better model the projection from the 3D scene to the image plane is also unknown. For example, the 3D-to-2D projection for aircraft images can be modelled by an orthographic projectivity, whereas other imagery generally has stronger perspective effects. Therefore, the only information available is the pixel values of the images. The system should be able to register the images by finding correspondences in both images. The solution to the correspondence problem is difficult when the two cameras are wide apart since strong photometric and geometric distortions occur.

The ample, existing literature mainly covers three different approaches for the description of the information that the images contain based on: *a)* the detection of geometric features, *b)* the analysis of the appearance of the image pixels or *c)* a combination of both approaches. Scenes containing human-made artefacts will embody objects with well-distinguishable geometric characteristics. The description of the geometry of the scene may be therefore a good approach for these type of images. Likewise, highly textured images of natural scenes or even camouflage may not exhibit sufficient geometric support and the analysis of the photometry in the image is preferred. Methods that combine geometry and photometry claim to combine and exploit the best of both disciplines. That is something that seems sensible according to the nature of the images. Despite that in the last years some methods have displayed a quite reliable performance [83,5], there is no common framework to image registration and the success is still dependent on the *friendliness* of the image towards each method.

Notwithstanding, there is a common strategy or methodology [131] that most of the registration techniques share:

- Feature detection. This consists of the extraction of significant features from the images. These features can be corners, edges, intersections, contours, regions, saliencies, etc. Control points are representations of these features, being for instance the termination of edges, high curvature points, centres of gravity of regions or others. Many of these features will be detected in both images, some will not. Therefore, the selection of features to look for in the scene plays a

determining role to carry out a successful registration since it will lay the foundation for the following steps of the process.

- Feature matching. Once features have been found in both images, the problem is formulated as identifying their counterparts. To this end, feature descriptors, similarity measures and ways to disambiguate matches are used. Pairs of detected features have suffered the aforementioned changes (geometry, photometry, noisy pixels…), and so the descriptors and measurements of similarity must be flexible and consistent for the right discrimination between correct and false matches.

- Transformation model. The matching function applied to the sensed image that best maps its counterpart to the reference image must be estimated. The parameters of this objective function are usually iteratively computed until a maximum (or minimum) of this function is achieved.

- Image re-sampling and transformation. This final step is based on an improvement of the accuracy and the mapping of every pixel of the sensed image into the reference image by means of the transformation model.

1.2 Contributions

We propose a method for registration of wide baseline images from a pair of uncalibrated cameras. Our approach consists in the description of the properties of the image views by means of geometric and photometric invariants. We trim the information in the image to regions nearby contours that lie over highly informative points. These geometric regions are defined by an affine arc-length metric and extracted along the contours. The difficulty of working with contours is that these can be partially detected, susceptible to occlusions and assigned a different label at junctions. The usefulness of the affine arc-length metric in our system is subordinated to finding contours that are reliably extracted in both images, *i.e.* both endpoints are corresponding points and thus the affine arc-length is an invariant under a local affinity. We propose a strategy to overcome the weaknesses of contour detection and the affine metric by extracting view-point reliable, high-curvature points that lie over contours or in their proximity. The information - contours and high-curvature points - is organised in a

graph structure, where the edges are contours and the nodes are the high-curvature points. We use the affine arc-length metric along contour segments to define an affine invariant geometric descriptor. This descriptor defines affine invariant regions where to analyse the photometry, which is incorporated to the descriptor.

The system attains advantage over other methods in the sense that it can be adapted to work with different photometric descriptors over the affine geometric regions defined. For instance, it can be expanded to multi-modal applications as long as the contour maps are accurately detected.

We make use of robust iterative methods to discern consistently counterpart correspondences within the dense feature space of invariant descriptors. An important property of the method relies on the fact that each descriptor encapsulates two points of interest (the two end-points that delimit the contour segment where to extract the information along). The advantages of that approach are that either reduces the computational load of the RANSAC-based algorithm since the number of iterations required to convergence is drastically reduced or expands the power of the algorithm to deal with larger proportions of outliers at the same cost.

1.3 Thesis structure

The thesis is organised in the following way:

Chapter 2 is split in two parts. The first one is a compound of brief definitions, concepts and state of the art in image registration techniques. The second part narrows down image registration to the wide-baseline case. The literature is wide, and the most significant works on feature extraction, descriptors, invariance and robust estimation of matching parameters are presented.

Chapter 3 shows methodologies and practical examples on the extraction of geometric features. Edges are extended to contours by using vicinity, orientation and good continuation criteria. Points of interest are also defined, that together with contour maps, are reorganised in the form of a graph where edges are contours connecting points of interest. We also extract geometric regions around contours for photometric support.

Chapter 4 deals with the analysis of affine invariance over geometric contours and over photometric patches. The affine arc-length and the affine invariant colour moments are analysed. We describe how we define the descriptors and perform some matching experiments based on distance among descriptors. We also include a study on error analysis.

Chapter 5 describes robust methods to identifying and rejecting outliers from the set of correspondences given by the descriptors defined in Chapter *4*.

Chapter 6 gathers final conclusions and future work.

Chapter 2 – A review of image registration and wide-baseline matching

This chapter discusses briefly basic definitions of image formation concepts and transformations in this context, *2D* image projections from the *3D* world to the image plane and transformations that approximate one image to the other one for the stereo case. After that, we start a brief review on image registration methods according to a classification based on the common steps involved in registration processes. Finally the last section narrows down image registration to the wide-baseline case, where the state of the art is thoroughly covered.

2.1 Registration of images

Image registration is a pre-processing step for mapping two images of the same setting which are taken from different points of view, sensors or over a period of time. According to these imaging conditions, there will be respectively a multi-view, multi-modal or multi-temporal analysis of the image data. Figure 1.1 represents two stereo images of an indoor scene. The images have been taken from different points of view and there is also a change in the photometric conditions.

Figure 1.1. Stereo images of a scene in the registration problem.

The input for the registration process can be pixel values, features or higher-level decisions (objects) extracted from the images. As stated before, the final objective is the alignment of the two (or more) images of a scene, the sensed and the reference images, into a common framework or co-ordinate system by finding a correspondence function.

Although there are some methods that rely on the manual extraction of control points; the work herein, as the vast majority of the current works, is centred in the automatic registration of images from uncalibrated cameras.

Registration techniques have been widely used for many years in different research areas such as [13]:

- Computer Vision and Pattern Recognition: for tasks in automatic object recognition, segmentation, shape recovery, motion analysis, stereopsis and character recognition.

- Cartography: for reconstructing our three-dimensional world by finding control points in images.

- Medical Image Analysis: for clinical diagnosis and to monitor the evolution of illnesses, especially to gather information from different sensors such as CT (computed tomography) which is a specialised X-ray technique, MRS (magnetic resonance spectroscopy), MRI (magnetic resonance imaging), ultrasound, SPECT (single photon emission computed tomography), PET (positron emission tomography), NMR (nuclear magnetic resonance), etc.

- Satellite and airborne imagery: for civilian and military intelligence uses such as agriculture, meteorology, oceanography, geology, earth resource and environmental issues among others.

2.1.1 Basic definitions

If I_1 and I_2 are 2D arrays representing two intensity images, the mapping between them is given by [13]:

$$I_2(x,y) = g(I_1(f(x,y))) \tag{2.1}$$

where g is a 1D intensity transformation and f is a 2D geometric or spatial transformation:

$$(x', y') = f(x,y) \tag{2.2}$$

Consequently, neglecting the intensity transformation and focusing only on the geometric transformation suffered, which is a major difficulty in registration, this can be expressed as a two single-valued functions f_x and f_y:

$$I_2(x,y) = I_1(f_x(x,y), f_y(x,y)) \tag{2.3}$$

the mapping in equation (2.3) according to equation (2.2) can be expressed as:

$$I_2(x,y) = I_1(x', y') \tag{2.4}$$

2.1.2 Domain of transformations

2.1.2.1 Geometric distortions

In our context, there must be considered the estimation of the image transformations according to two different cases: 3D-to-2D camera projections from a 3-dimensional point in the space to a 2-dimensional point in the image plane and 2D-to-2D planar homographies, *i.e.* projections of local planar patches in the image can be approximated by a transformation.

3D-to-2D camera projections. This kind of projection deals with the mapping of every point $(x,y,w)^T$ in the 3D space onto the corresponding point $(x',y')^T$ in the image plane. The function that maps 3D to 2D points is the camera. The simplest model of a camera is the widely used *pinhole camera model*, also referred as *perspective model* [109,36]. From a geometric point of view, the perspective model of a camera defines the focal

length as the distance between the pinhole *(O)*, the co-ordinate origin of the camera frame, to the (virtual) image plane along the optical axis *(Z)*. The optical axis is the axis which has its origin in the pinhole and is perpendicular to the image plane *(Π)*. The intersection of the optical axis with the image plane is called the image centre or principal point. The 2D point *p* is the image of the 3D point *P*, *i.e.* $p=[x,y,z]^T$ and $P=[X,Y,Z]^T$. The camera frame characterises the following equations of perspective projections:

$$\begin{aligned} x &= f\frac{X}{Z} \\ y &= f\frac{Y}{Z} \end{aligned} \tag{2.5}$$

The perspective projection does not preserve a one-to-one size map between the image of the object and the real object. Indeed, objects further away are represented smaller than closer ones.

There are other approximations that may be applied in our case, notably the *affine projection models*. One of these is the *weak-perspective camera model*, appropriate when the relative distance δZ between two objects along the depth coordinate Z (optical axis) is very small compared with the distance Z' from the scene objects to the camera frame. Typically, $\delta Z < Z'/20$. The weak-perspective model can be approximated from the full projection model as:

$$\begin{aligned} x &= f\frac{X}{Z} \approx \frac{f}{Z'}X \\ y &= f\frac{Y}{Z} \approx \frac{f}{Z'}Y \end{aligned} \tag{2.6}$$

Another affine model is the *orthographic projection, which* supposes that the camera is always in a far and constant distance from the scene, the focal length $f\rightarrow\infty$ and then also $Z'\rightarrow\infty$ being $f/Z'=1$ and having all the light rays parallel to the optical axis (figure *2.2*).

$$\begin{aligned} x &= X \\ y &= Y \end{aligned} \tag{2.7}$$

For a more detailed information on cameral models we refer to [36,54].

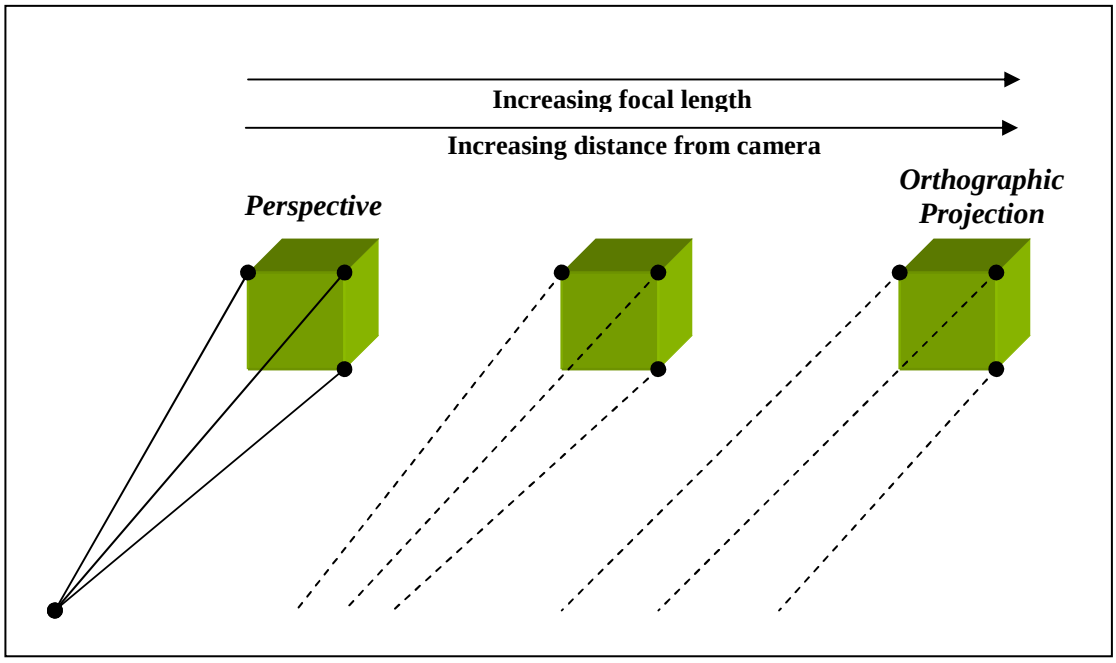


Figure 2.2. Perspective and orthographic projections (from reference [50]).

2D homographies

The two-dimensional homography refers to the mapping between *2D* images or patches. Given a point $p=(x,y)^T$ in the plane of an image, the corresponding point $(x',y')^T$ in the other image is found by estimating the *2D* projective transformation $T : P^2 \rightarrow P^2$.

The *3x3* general transformation matrix *T* [124] can represent most of the basic geometric transformations that may occur between any two *2D* images (translation, rotation, scaling, shearing, reflection and perspective). The mapping expressed in term of homogeneous coordinates is given by:

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = T \begin{pmatrix} x \\ y \\ z \end{pmatrix} \tag{2.8}$$

$$T = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \tag{2.9}$$

In equation (2.8), the third dimension could be neglected since the locations of the cameras with respect to the world reference frame are unknown and we are dealing with 2D-2D projections, *i.e.* (x',y',1)' = T (x,y,w)'.

Decomposing equation (2.9) we have:

$$T_s = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \tag{2.10}$$

the sub-matrix of T, T_s, represents scaling, shearing and rotation in equation (2.9). Translation in T is due to $[a_{13}\ a_{23}]^T$ and perspective transformation is defined by $[a_{31}\ a_{32}]$. Finally, a_{33} sets the scaling.

There can be different sorts of more complex matching transformations defining the spatial transformations or displacements that images undergo [54,124,114]. These distortions in the images are usually combinations of some basic transformations. Their definitions are as follows:

- Isometries take place when the origins and basis vectors of both coordinate systems are not the same. They are a combination of single transformations such as translation, rotation and mirror reflection:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} \varepsilon\cos\varphi & -\sin\varphi & t_x \\ \varepsilon\sin\varphi & \cos\varphi & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{2.11}$$

  being φ the rotation angle, $[t_x,t_y]^T$ the translation vector and $\varepsilon=\pm1$. When $\varepsilon=-1$, the mirror effect occurs.

Ignoring the reflection, it has three degrees of freedom (φ, t_x,and t_y) and invariants are length, angle between lines and area.

- Similarity transformations extend the previous transformation to isotropic scaling *s*:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} s\cos\varphi & -s\sin\varphi & t_x \\ s\sin\varphi & s\cos\varphi & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{2.12}$$

This transformation has got four degrees freedom. Ratios of length and angles between lines are preserved.

- Affine transformations map at any dimension straight lines to straight lines maintaining parallelism. Every affine transformation is a decomposition of a linear matrix transformation and a simple translation.

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{2.13}$$

It has six degrees of freedom, relating to parameters $a_{11}…a_{22}$ and t_x, t_y. It is the most commonly used transformation since it allows the overlay of images taken from the same angle of view but from different positions as well as skew. Invariants are parallelism, ratios of length of parallel lines ratios of areas and centroids.

- Projective or perspective transformations map straight lines onto straight lines in the other image, but parallelism is not usually preserved.

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{2.14}$$

Nine independent parameters define this transformation $a_{i,j}$, $i,j=1,2,3$. Projective transformations take place when $[a_{31}\ a_{32}]$ is non-zero. Affine transformations are thus a particular case of projective transformations when $[a_{31}\ a_{32}]$ is zero.

The transformation matrix can be normalized so that $a_{33}=1$, having then equation (2.14) eight degrees of freedom and allowing planar quadrilateral-to-quadrilateral mapping. The most characteristic invariant is the cross-ratio of four collinear points.

- Bilinear transformations are similar to projective transformations. Horizontal and vertical straight lines are mapped onto straight lines but lines of any other direction will be transformed to curves.

$$\begin{aligned} x' &= a_0 + a_1 x + a_2 y + a_3 xy \\ y' &= b_0 + b_1 x + b_2 y + b_3 xy \end{aligned} \tag{2.15}$$

This transformation is defined by eight independent parameters (a_i,b_i), $i=0,1,2,3$. It copes with the problem of non-planar quadrilaterals.

In R^3, since bilinear transformations are generated from affine transformations the cross-ratio of four points is an invariant under bilinear transformations.

- Curved transformations may map any straight line onto a curve in the other image. Therefore, they are also called elastic or non-linear transformations:

$$\begin{aligned} x' &= a_0 + a_1 x + a_2 y + a_3 x^2 + a_4 xy + a_5 y^2 + \ldots \\ y' &= b_0 + b_1 x + b_2 y + b_3 x^2 + b_4 xy + b_5 y^2 + \ldots \end{aligned} \tag{2.16}$$

It can consider the following division of transformations: those applied to planar mappings (affine and perspective) and those that allow non-planar mappings (bilinear and curved transformation).

The domain of transformations depends on whether the image transformation involves defects on the whole image or just part of it. Hence the change of one parameter in

global matching transformations will affect the entire image, whereas local matching transformations will only change part of the image. Local matching considers images as a composition of patches. It is usually suitable for medical and aerial applications, where the images go through some local deformations.

Transformations can also be classified according to the accuracy required. Interpolating functions map exactly the control points of the sensed image to those of the reference image; while approximation functions take into account certain trade-offs between accuracy and other constraints required [131].

2.1.2.2 Photometric distortions

Photometric distortions are due to variations in the photometry of the scene that are related to changes in the illuminant, to the geometry and reflectance properties of the surface [56] and the sensors used. Reflection models differentiate between diffuse and specular surfaces. Models can be complex but often diffuse surfaces are considered as Lambertian. Generally the camera and the illumination source are far away from the objects of interest within the scene. Therefore, it is normally assumed the existence of planar surfaces or even a whole image where the light arrives with the same orientation. A change in the illumination colour corresponds to different scaling of the RGB values over Lambertian surfaces, whereas a change on the position of the illuminant results in an equal scaling of all RGB bands [43]. We can consider three different models of photometric distortion for RGB images:

a) Diagonal:

$$\begin{pmatrix} R' \\ G' \\ B' \end{pmatrix} = \begin{pmatrix} s_R & 0 & 0 \\ 0 & s_G & 0 \\ 0 & 0 & s_B \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \tag{2.17}$$

b) Scaling plus offset:

$$\begin{pmatrix} R' \\ G' \\ B' \end{pmatrix} = \begin{pmatrix} s_R & 0 & 0 \\ 0 & s_G & 0 \\ 0 & 0 & s_B \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} + \begin{pmatrix} o_R \\ o_G \\ o_B \end{pmatrix} \tag{2.18}$$

c) Affine:

$$\begin{pmatrix} R' \\ G' \\ B' \end{pmatrix} = \begin{pmatrix} a_{RR} & a_{RG} & a_{RB} \\ a_{GR} & a_{GG} & a_{GB} \\ a_{BR} & a_{BG} & a_{BB} \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} + \begin{pmatrix} o_R \\ o_G \\ o_B \end{pmatrix} \tag{2.19}$$

Indoor images can be approximated by the first two models, whereas an affine approximation can be a valid model for outdoor images [82]. However, non-linear photometric distortions generally occur in reflective surfaces or when sensors saturate being that instance more difficult to model.

Figure *2.3* shows an input image and transformed versions undergoing combinations of affine geometric transformations with scaled photometric transformations.

2.1.3 Review of existing research

This section presents a very brief overview of image registration techniques. The organization of the discussion is based on [13] and the comprehensive survey compiled by Zitová and Flusser [131], which is an excellent source of references.

The feature space can be defined as the overall data representation available in the image to undertake the registration process. These data can be complex features extracted on the image but also intensity distributions. The kind of data to search is dependent on the sort of transformations suffered by the images as well as the nature of the imagery and the content of the scene to solve the correspondence problem.

The methods are classified according to the aforementioned common steps for registration described in Section *1.1* and according to whether the approaches are based on intensity (area-based methods) or features (feature-based methods). In this section we are discussing only a few of the approaches that we consider most relevant for us. Consequently, we refer to the taxonomies above for further information over registration methods.

Figure 2.3. Photometry and geometry distortions. a) Original image, b) 20º rotation, c) [0.1 0.1] shear, d) [0.3 0.3] shear, e) [0.9 0.9] scale, [0.1 0.1] shear, [0.7 0.65 0.75] RGB scaling *type D*, f) [0.9 0.9] scale, [0.1 0.1] shear, [0.4 0.4 0.4] RGB scaling *type D*, g) [0.9 0.9] scale, [0.2 0.2] shear, [0.6 0.55 0.65] RGB scaling *type D* and h) 20º rotation, [0.1 0.1] shear, [0.6 0.6 0.6] RGB scaling *type D*.

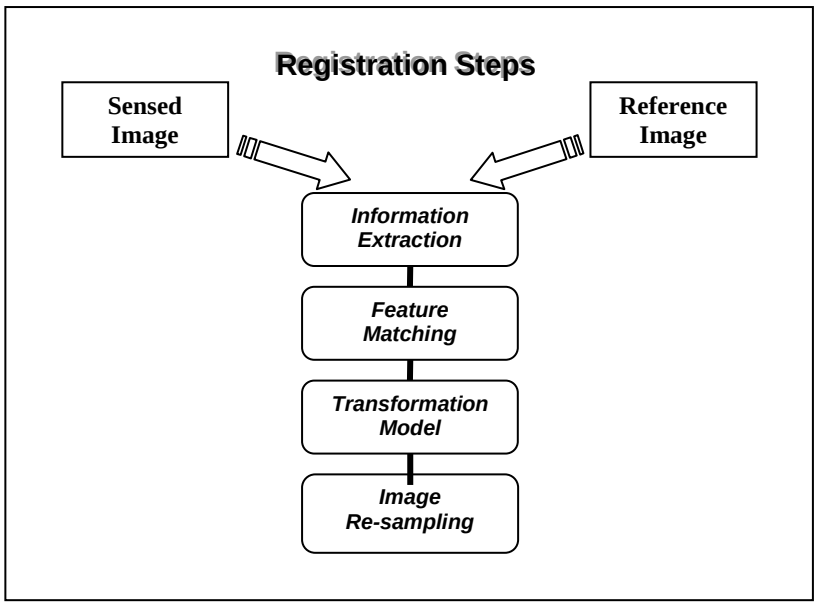


Figure 2.4. Image registration methodology.

2.1.3.1 Area-based methods

Many images do not have readily identifiable features and for this reason area-based methods are preferred. There is no initial step for the detection of features since these appearance-based methods rely on intensity distributions within a region of an image. These methods perform the two first steps of registration, the extraction of information and the posterior matching itself, in a common step by fusing both.

They usually consist of opening a window to define the area to work in. The restrictions of early area-based methods were: first, most of them are only invariant to translation, a simple rotation between the two images will provoke an impossibility of registration; and second, windows covering smooth and non-distinguishable areas cause the failure of area-based methods.

The main families of area-based methods are presented in figure *2.5*. Herein, we discuss upon methods based on mutual information and salient features, since we consider them more relevant to the wide-baseline case.

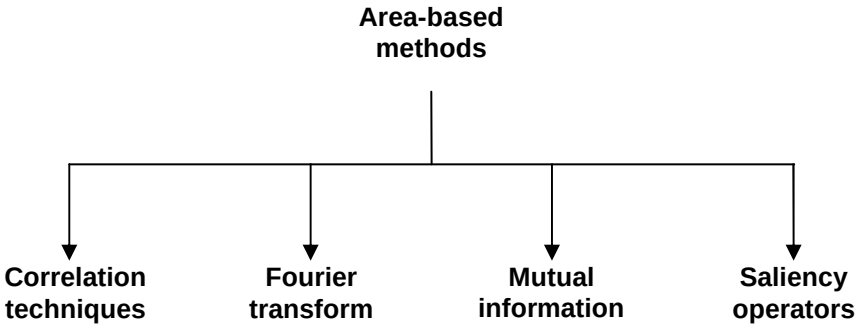


Figure 2.5. Classification of area-based methods.

Mutual Information (MI) methods. Mutual information comes from the discipline of Information Theory and is a recent technique used in image registration. There have been very promising approaches in the field of multi-modal registration, such that these methods are at the forefront of current research. Theoretically, the mutual information can be expressed with respect to a set of coordinates, *x*, as a relation of marginal, conditional and joint entropies *h(·) [9*]:

$$\begin{aligned} I(v(T(x)),u(x)) &\equiv h\big(v(T(x))\big) - h\big(v(T(x))\big|u(x)\big) \\ &= h\big(u(x)\big) - h\big(u(x)\big|v(T(x))\big) \\ &= h(u(x)) + h(v(T(x))) - h(u(x),v(T(x))) \end{aligned} \tag{2.20}$$

where *v(T(x))* is part of the target data which should be registered with a model *u(x)* and *T* is the transformation or pose which links the model and image co-ordinate frames. In the expression above, the marginal entropy $h\big(v(T(x))\big)$ gives a measurement of the degree of prediction of the target data (random variable). The lower the entropy, the more likely the variable to be predicted. The higher the entropy means the higher the degree of uncertainty. The conditional entropy $h\big(v(T(x))\big|u(x)\big)$ is a measurement of the uncertainty left in the target data after the model is observed. Therefore, the difference is the information that one variable gives about the other.

The entropies of one and two random variables are given by expressions (2.21) and (2.22), respectively:

$$h(y) = -\int p(y)\ln(p(y))dy \tag{2.21}$$

$$h(z,y) = -\int p(z,y)\ln(p(z,y))dzdy \tag{2.22}$$

with *p* the probability of a variable (e.g. probability density function of the image).

The work of Viola and Wells [117] has been very influential although it was not the first to make use of methods derived from information theory. They used a maximisation of the mutual information, both to align two different MRI images of the same object and to align an object model and an image. This allowed them to register separate MR images and to find object pose by registering 3D object models to real scenes. In the latter case, they assumed that the image was a derivable function of the model e.g. they presumed Lambertian surfaces and the existence of a *consistency* measure between intensity and normal of the model when the two images are aligned. Other approaches for mutual information are in [80,63,87]. See [2] for mutual information for feature selection over characteristics of edges such as location, strength and orientation; edges and junctions in [74], registration of images by combining gradients in [107], over neighbourhoods [92] or a comparison to a new gradient-based measure [51].

Saliency Operators. This set of operators extracts unpredictable characteristics of the geometric properties of the image regions with the aim of estimating feature descriptors to solve the matching problem. Kovesi [61] worked with phase congruency to perform a saliency measure of edges and, achieved multi-scale analysis by using wavelets in [62]. In [123], close boundaries were extracted by connecting contours in terms of saliency over proximity and curvature. Gal and Cohen [41] presented salient-based descriptors of local surfaces. We will focus on the strategy proposed by Kadir and Brady [57]. They considered the saliency concept as a probabilistic measure calculated over a local multi-scale analysis [55]. Their implementation is invariant to rotation, scaling, some photometric changes and translation as well as robust to noise, viewpoint change and intensity scaling. Kadir *et al.* in [64] expanded the salient algorithm to attain invariance against affine transformation by defining adjustable ellipses at different scales instead of circular patches. The scale parameter *s* is replaced by the three coefficients that define the ellipse: the major axis *s/sqrt(ρ)*, the minor axis *s·sqrt(s)* and the orientation of the ellipse θ. The parameter ρ is the axes ratio. The adjustment of the parameters of the

ellipse is performed in an adaptive way by means of iterations, according to the strategy used in [4,81] and developed by Lindeberg and Gårding in [66].

The unpredictability of the images is analysed by means of the Shannon entropy over a range of scales $H_D(s)$. Therefore, the algorithm extracts circular patches at different scales around image pixels as samples to work with. The definition of entropy is defined by:

$$H_D(s) \cong -\int p(I,s,x)\log_2 p(I,s,x)dI \qquad (2.23)$$

being $p(I,s,x)$ the probability density function of the intensity I for the point x at scale s.

This probability density function can be approximated by means of a grey-value local histogram. Peaked histograms involve that the pixel information can be predicted since the intensity values lie within a reduced intensity range. At the other hand, spread out histograms show that the probability of finding the value of each pixel tend to be similar for all the pixels in the image, *i.e.* in a flat histogram all pixels have the same probability. A peaked histogram is considered very informative while a flat one not. However, this definition of saliency by means of entropy declares highly salient regions of the image with spread out histogram. Therefore, the salient descriptor is a measure of the difficulty that an intensity-based descriptor would have. If saliency is a degree of unpredictability, salient regions will not be easily available by a prior model description.

A set of scales s_p where the entropy measure peaks is selected according to:

$$s_p \cong \left\{ s : \frac{\delta H_D(s,x)}{\delta s} = 0, \frac{\delta^2 H_D(s,x)}{\delta s^2} < 0 \right\} \qquad (2.24)$$

where the first equality defines a stationary point but does not reveal a local maximum, minimum or point of inflexion. The second derivative yields the maximum.

After histogramming, all spatial information in the image is lost. Therefore, any order of the pixels within the sampling window gives the same entropy value. However, that

does not happen at different scales as the sampling windows do not cope with the same number of pixels. Indeed, the sampling windows are subsets of the largest one.

Referring again to the unpredictability aspect of the saliency concept, the reader may think that the method is highly dependent of noise. To avoid this dependence the *inter-scale saliency* constraint, W_D, is introduced.

$$W_D(s,x) \cong s\int \left|\frac{\delta}{\delta s}\right| p(I,s,x)dI \tag{2.25}$$

The inter-scale saliency measures the changes of the probability density function and its entropy with the variation of scale. In the discrete case, $W_D(s,x)$ is calculated between the scale at which entropy peaks s and $s-1$.

The final definition of saliency $Y_D(s_p,x)$ is the product of the maximum entropy $H_D(s)$ by the inter-saliency measure $W_D(s)$ at the scale which the entropy is maximum.

$$Y_D(s_p,x) \cong H_D(s_p,x)\cdot W_D(s_p,x) \tag{2.26}$$

Therefore, the inter-scale saliency measure should be maximised to obtain a high saliency measure.

We have performed some experiments for extraction of the salient features as in [57]. The first pair of images defines a scene with two vehicles in a car park, where the vehicle of interest (Land Rover) changes its position *45°* within the setting. The images form part of a set taken with a visible camera within a short period of time between snapshots (apparently similar photometric conditions) and are courtesy of BAE SYSTEMS. The second pair of images has been taken by the author with a digital camera at a lower resolution. This scene is challenging since many changes take place in the setting. The object of interest, a civilian car, remains static but there is a wide change in the position of the camera, besides remarkable photometric conditions occur, considerable occlusions take place and new objects appear in the scene (for instance, a four-wheel drive vehicle). The last pair of images has been taken by a camera in the medium infrared band. These are high-resolution pictures of a Land Rover (toy)

changing its position. The saliency detector is applied to every pixel in the image, with the support of a defined, surrounding region.

Visible imagery (I)

Figure *2.6* shows the setting composed of two visible grey-level images. The size of the pictures is *800x600* pixels. The camera remains static, only the object of interest shifts its position *45°*. Between the two frames, some photometric variations occur as it can be appreciated in the reflectance of light over the civilian car. In figure *2.7* the top plots represent the *3D* maps of the intensity values of the pixels in the images. The next two figures below denote the entropy map of both images. This entropy is the maximum entropy $H_D(s)$ extracted over the multi-scale analysis performed at every pixel. Notice how geometric objects exhibit values of entropy higher than the background. Likewise, the morphologies of the background can be better distinguished by the human eye when applying the false-colour map of the entropy measure than in the original images. The next pair of figures represents the scale selected at every point of the image, *i.e.* the scale within the given set which shows higher entropy. The predefined set of scales is composed by five different circular scales of radius *2*, *4*, *6*, *8* and *10* pixels. The figures clearly illustrate that the system prefers large scales, as it is usually more likely to find a wide diversity of pixel values and then, higher entropy. Table *2.1* shows the percentage of use of every scale.

SCALE (pixels)	**0 degrees (%)**	**45 degrees (%)**
2	0.33	0.21
4	1.62	1.55
6	4.86	5.05
8	7.57	7.27
10	85.62	85.91

Table 2.1. Percentage of the use of every scale in the images.

The inter-scale saliency measure $W_D(s,x)$ (equation *2.25*) is depicted in the next pair of figures. It gives a dimension of the change of the *pdf* and the entropy with the scale. It is calculated between the scale at which the entropy registers a maximum and the previous scale. This measure is a sort of evaluation of the self-dissimilarity of the local region in the space of scales. That gives rise to a more robust performance in the sense that self-similar regions will not be extracted, reducing therefore the possibility of false matches. The inter-scale saliency weights the entropy to produce the final descriptor, the saliency measure $Y_D(s_p,x)$ given by equation *2.26*. As can be seen in the final result, high values of saliency are common in the same features of both images. That can be understood as the saliency operator can be able to extract distinguished features in stereo images, performing thus the necessary basis in feature extraction to carry out the matching between the two images. Nevertheless, these salient features should be combined with some kind of geometric support or a descriptor that allows the extraction of some other parameters to define a descriptor vector.

Figure 2.6. Car setting.

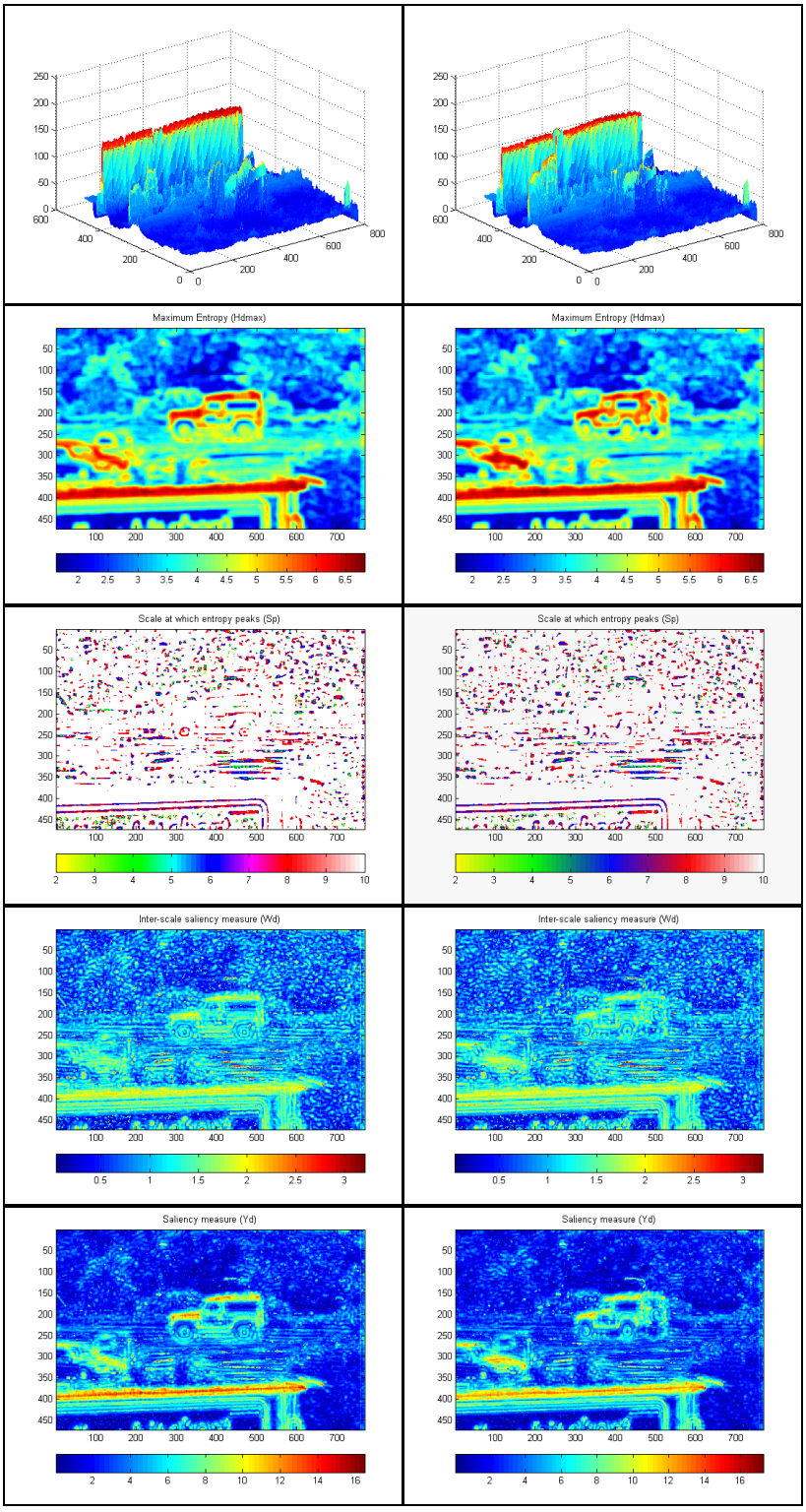


Figure 2.7. Extraction of saliencies of an outdoor scene. a) and b) Image intensity values; c) and d) maximum entropy within the given scales; e) and f) scale at which entropy peaks; g) and h) inter-scale saliency measure; i) and j) saliency measure,

Visible imagery (II)

In this case, the scene under study portrays another example of visible imagery with a more difficult layout. The two images exhibit considerable changes in lighting conditions and point of view, as well as occlusions occur and new objects appear in the scene (see figure *2.8*). The pictures were taken at Heriot-Watt University with a commercial digital camera. The distance from the camera to the object of interest is around *30* metres for the left-hand side picture and *50* metres for the other one. The resolution of the images is *320x240* pixels.

Examining the results in figure *2.9*, the results are very different from the ones obtained for the previous imagery. The blob-wise features obtained are more palpable in this set of images. Notwithstanding, these blobs are inherent to the algorithm and a consequence of the isotropic way the scales are defined (circles). Their major prominence is due to the lower resolution of the images and the profiles we are coping with (see that the image intensities in the graphs are very discontinuous). These effects are more outstanding in the right-hand side picture, where the scene is hardly recognized. Despite the blob effect, the left-hand side picture still depicts the main objects in the setting. The object of interest is identified as a high-entropy value blob but keeping a perceptible shape of the vehicle.

Figure 2.8. Complex wide-baseline setting.

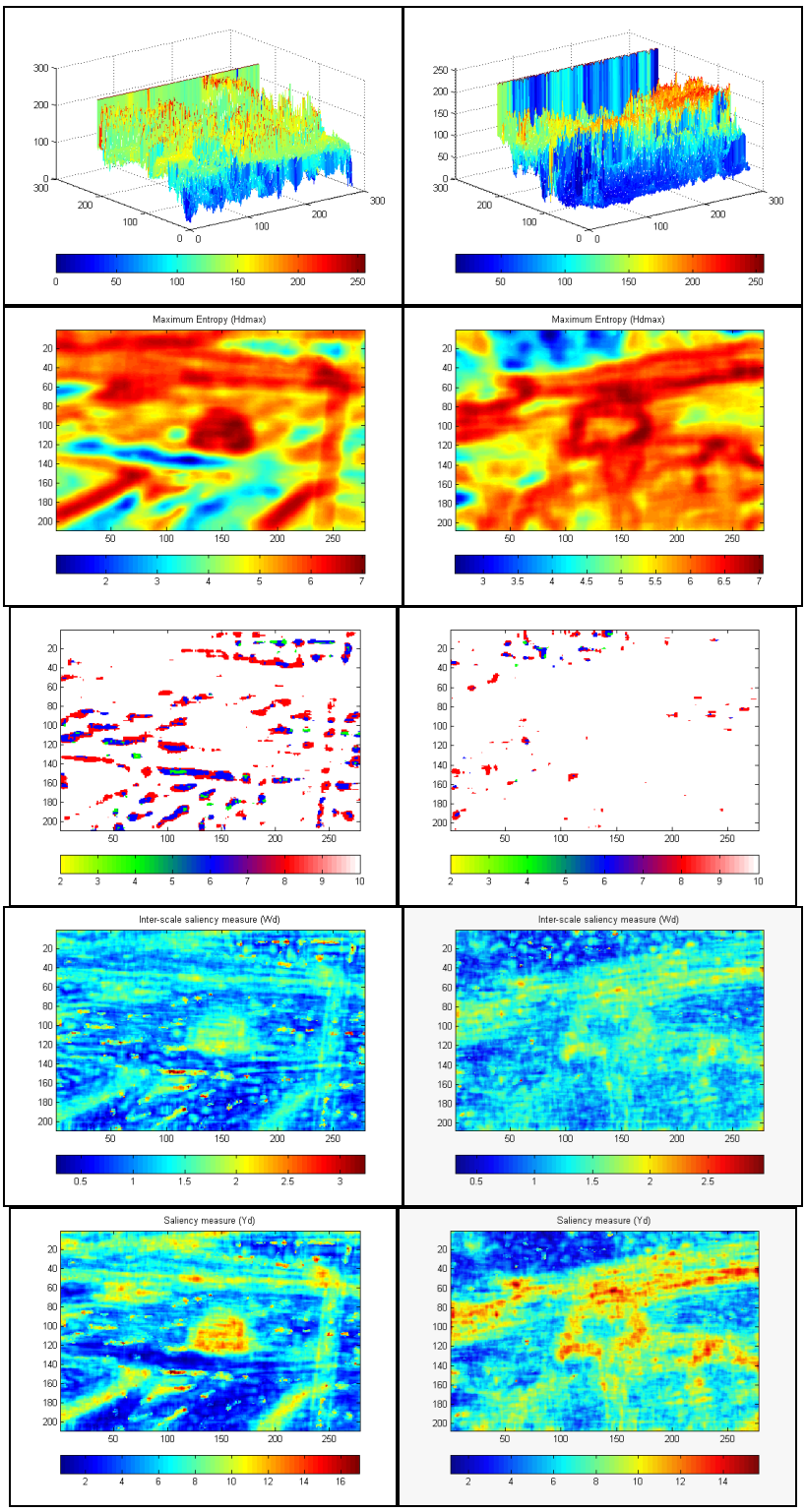


Figure 2.9. Extraction of saliencies in visible images of a complex scene. a) and b) image intensity values; c) and d) maximum entropy within the given scales; e) and f) scale at which entropy peaks; g) and h) inter-scale saliency measure; i) and j) saliency measure .

Regarding the scale maps, the excessive abundance of white regions or maximum scales in these figures comes to confirm that as much texture or wider variation of pixel levels in the image patterns there exist, the highest entropy is found within the largest window (maximum scale).

SCALE (pixels)	Left image (%)	Right image (%)
2	0.01	0.00
4	0.77	0.11
6	4.61	0.50
8	8.35	1.69
10	86.26	97.70

Table 2.2. Percentage of the use of every scale in the images.

The inter-scale saliency measures present a similar behaviour than their counterpart entropies, maybe even more blurred. The regions in the scene can still be distinguished in the figure at the left side, but the visual information is almost missed in the other one. The saliency measures improve slightly these intermediate steps but this is a complicated, low resolution image

Infrared imagery

The pair of toy images of the Land Rover in figure *2.10* is taken in the medium infrared with a resolution of *421x337* pixels. The object of interest is presented in high-resolution and there is neither background nor other objects in the scene. By having a look at the infrared images, different materials in the vehicle present different grey tones, such as the door, the wheels, the glasses, etc. The entropy map reflects high entropy values at the lines which define the figure. This behaviour is similar to the visible one and it is described in figure *2.11*. The scale map presents a bigger predominance of smaller scales in the background but little presence on the object of interest. Thus, the behaviour is similar to previous examples. The inter-scale saliency measure (W_D) also defines the lines of the vehicle resulting in a saliency map which stresses the outlines of the figures. Figure *2.12* shows the results obtained after

modifying the scales of the window function. The left image corresponds to window functions of twice the size used in the example above, *i.e.* radii of *4*, *8*, *12*, *16* and *20* pixels. The saliency map obtained presents a more blurred result than the one with smaller scales. Furthermore, the saliency measure ranges up to double values than the previous ones.

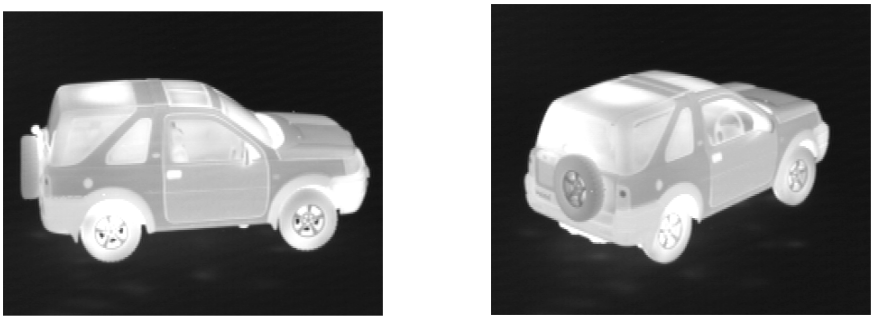


Figure 2.10. Wide-baseline image of a toy landrover.

Extraction of saliencies over seed points

The previous part embraced the saliency analysis according to a pixel-wise approach. Every pixel in the image was evaluated. That has the inconvenience of the weaknesses of correlation-based method which are sensible to photometric and scale changes. Basing the saliency measure on geometric features makes the system stronger against spatial transformations. Figure *2.13* shows a short example of the performance of this process together with some results on saliencies on regions around anchor points.

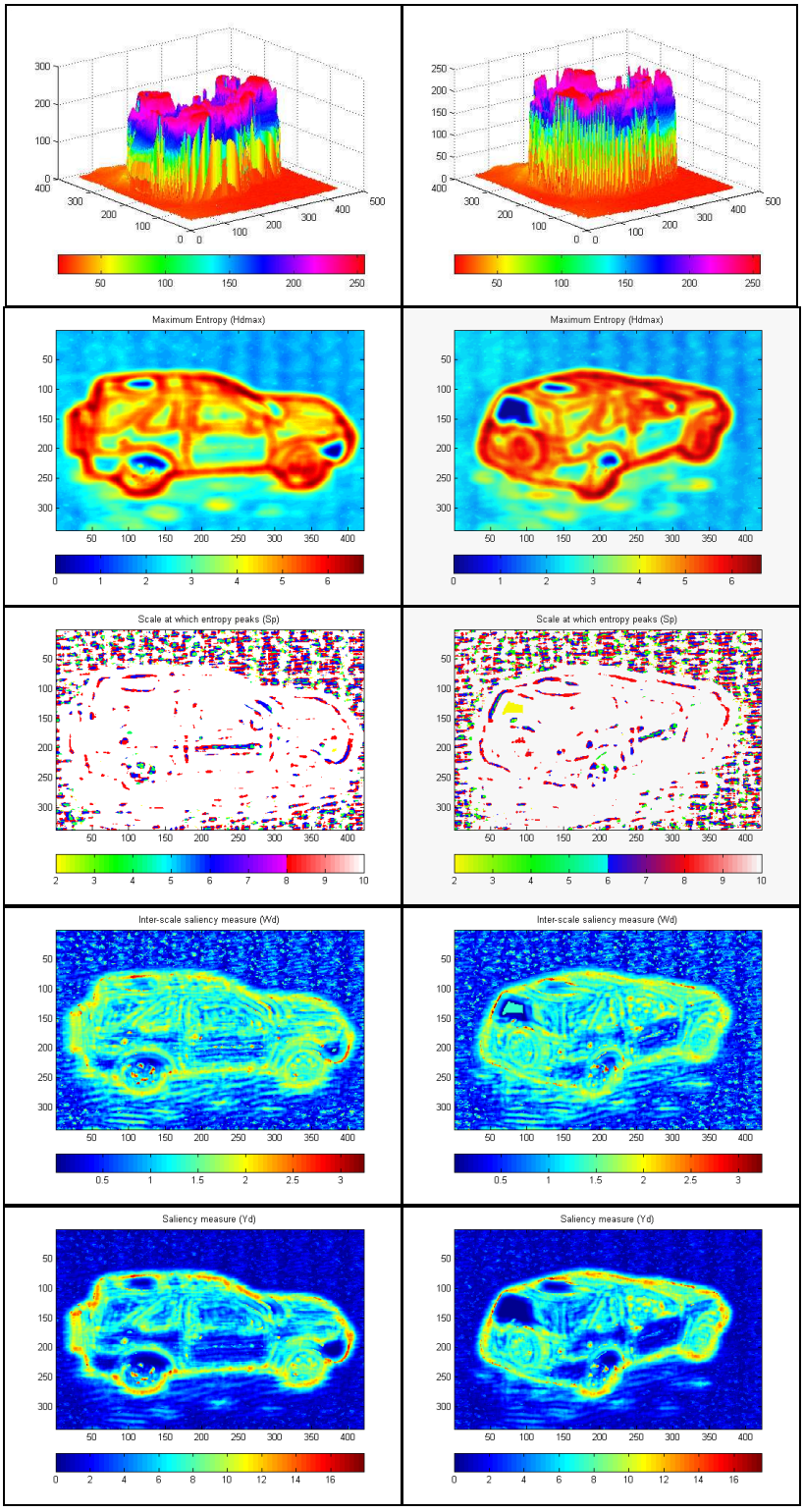


Figure 2.11. Extraction of saliencies in infrared images. a) and b) image intensity values; c) and d) maximum entropy within the given scales; e) and f) scale at which entropy peaks; g) and h) inter-scale saliency measure; i) and j) saliency measure.

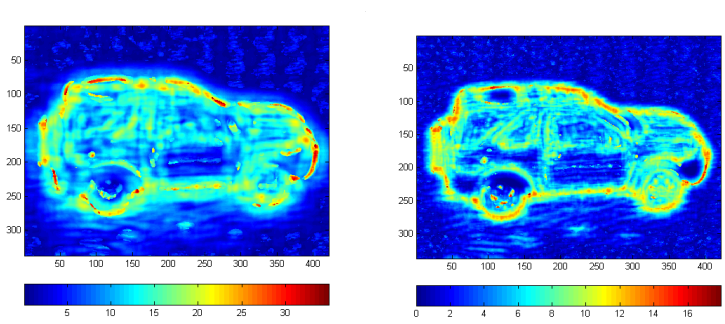


Figure 2.12. Comparison of saliency maps for different set of scales for the window functions. The possible window sizes in the image at the left-hand side are double than the ones in the other image.

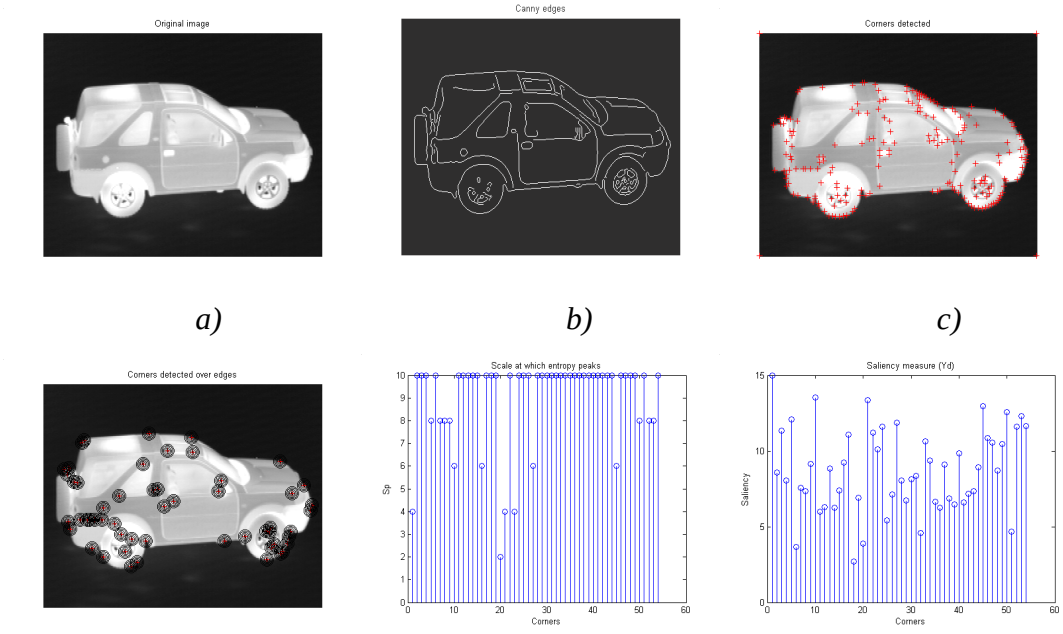


Figure 2.13. Different-scale saliencies over seed points. a) Original image, b) extraction of Canny edges, c) Harris corners found, d) corners detected over Canny edges, e) scales at which the entropy is maximum and f) saliency values for each seed point

2.1.3.2 Feature-based methods

These consist of the extraction of distinctive, detectable and scattered features such as regions, lines and points of interest over the pictures by means of invariant feature

detectors. Hence, feature-based methods are preferred when the image has distinctive objects, features or details to be detected. A proper detection is of vital importance.

Feature extraction

Region features are closed-boundary regions, e.g. forests, lakes, ponds, buildings, shadows, etc. which are usually detected by segmentation methods and can be represented for instance by their centre of gravity. The centre of gravity has the property of being invariant to rotation, scaling and skewing. The co-ordinates of the centre of gravity are also rather stable against random noise and grey-level variations. Region features have also been studied in a multi-scale hierarchy using invariant neighbourhoods around points of interest [124]. This particular case based on the invariant properties of the images will be developed in depth in the next section.

Line features are line segments, contours of objects, roads, etc, usually described by end-line and mid-line points. Typically one uses an edge operator, such as the Canny edge detector [17] followed by a contour tracking process. Finally, point features include corners, T-junctions, and Y-junctions as well as any other salient points in the scene [102]. Examples of points of interest are road crossings, line intersections, centroids of regions, local extrema, high-curvature points and so on.

Schmid *et al.* [102] conducted an evaluation study of the performance of detectors of interest points based on repeatability and information content criteria. The most extensively used methods for the detection of points of interest have been the Harris detector [52] and *SUSAN* (*Smallest Univalue Segment Assimilating Nucleus*) [96].

In the Harris-Stephens corner detector [52] the first step is to apply a Gaussian to smooth the image in order to reduce the image noise and prevent false corner detection. That is done over images containing the square image derivatives.

From the following moments matrix (gradients) of a grey-level intensity function $I(x.y)$:

$$M = \begin{bmatrix} \left(\frac{\delta I}{\delta x}\right)^2 & \left(\frac{\delta I}{\delta x}\right)\left(\frac{\delta I}{\delta y}\right) \\ \left(\frac{\delta I}{\delta x}\right)\left(\frac{\delta I}{\delta y}\right) & \left(\frac{\delta I}{\delta y}\right)^2 \end{bmatrix} \tag{2.27}$$

it can be found if a point is a corner by calculating the two eigenvalues of the moments matrix *M*. If the eigenvalues have large values, therefore a small motion at any direction will produce a considerable change in the grey-level value, specifying a corner is lying at this spatial co-ordinate.

The corner strength response function is defined by:

$$R = \det M - k(traceM)^2 \tag{2.28}$$

with k=0.04 as a value proposed by Harris.

Corners are given by local maxima of *R*. A threshold can be set in order to reduce the number of corners if required or to order corners according to significance. By means of a quadratic approximation of a neighbourhood of local maxima, sub-pixel accuracy can be obtained.

Feature matching

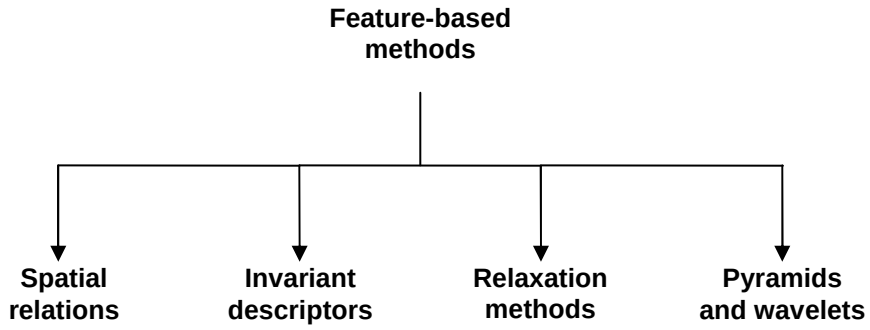


Figure 2.14 . Classification of feature-based matching.

Methods using invariant descriptors. Invariant descriptors characterise sparse features which do not change under a given photometric or geometric image deformation with the purpose of solving the correspondence problem. To consider a geometric instance, if we have a segment line its length will not change under a translation or rotation but it will under other transformations. For example, a circle under an affinity will be transformed into an ellipse. In the photometric case, the transformation will rely on extrinsic and intrinsic parameters of the cameras and the lighting conditions. Therefore, it is fundamental to know the kind of transformation the images will undergo and the set of features to work with in order to find descriptors invariant to this transformation. There is a vast group of methods based on the application of moment invariants to closed-boundary regions as well as many other describing image features or combinations of them. We refer once more to [131] for wider information and also to Section *2.2* where some methods based on invariant descriptors and focused on our practical case will be broached in depth.

2.1.3.3 Transformation of the model

Once the features have been extracted and their counterparts found, the mapping function which establishes the correspondence should be estimated. As mentioned before, the choice of the function relies on the image transformation; with the acquisition of the images and the registration accuracy in mind. Optimization techniques aim at finding a (minimum) maximum of an objective function which estimates the (dis-)similarity measure between two templates. The difficulty of the problem depends on the number of degrees of freedom of the transformation suffered by the image, as well as the complexity of the transformation function, *i.e.* the existence of multiple local minima or maxima.

2.1.3.4 Image re-sampling and evaluation

We have two images, each with a different coordinate system and a transformation that maps both coordinate systems. If the images need to be aligned, due to the discrete nature of images, the transformation of the input image onto the output image will entail the creation of new pixel locations. Therefore, image re-sampling comprises two steps:

the conversion of the image from the discrete to the continuous domain and the sampling at the new spatial positions.

The procedure consists of applying an inverse transformation to the pixels in the transformed coordinate system, generating the resampling grid. Next, the input image is converted onto the continuous domain with the aid of an interpolation function and then sampled at the resampling grid locations. Hence, interpolation and sampling determine the intensity value at a given position in between discrete samples. The infinite bandwidth of the discrete pixels of the image is limited to a finite bandwidth by the interpolator. There are many interpolation methods and the right choice depends on the accuracy desired and the computational cost that can be afforded. For some insight into the main interpolation kernels (nearest neighbour, linear interpolation, cubic convolution, cubic splines, sinc functions and exponential filters) we address to [124]. Re-sampling is useful for example for mosaicing, however it is not always needed as it is the case of the estimation of rigid transformations.

The evaluation part is related to the assessment of the accuracy of the registration process. Errors may take place and will accumulate during the features extraction phase (localization error), the matching of features (matching error) and the mapping (alignment error).

2.2 Wide-baseline registration

This section surveys different methods for single image modality, wide-baseline image matching. A variety of methods are presented. A good number of them share the common approach of extracting interest points and defining invariant regions in the surroundings. The use of local features is a matter of robustness, *i.e.* the system performs better when occlusions occur, when other objects present in the image divert the attention from the object of interest and when there exist changes in the background. Moreover, local region detection leads to a better chance of dealing with planar surfaces, which makes correspondence and transformation much simpler. Figure *2.15* illustrates a wide-baseline scene, where rotations, translations, scale changes and photometric variations take place. As a result of the wide baseline, the views exhibit occlusions and new objects occur. Figure *2.16* illustrates the basic blocks of the image registration process with some possible methods.

Figure 2.15. Wide-baseline scene. Significant changes in the viewpoint and photometric conditions. Moreover, new objects and occlusions take place in the scene.

2.2.1 Extraction of features

The vast majority of wide-baseline stereo algorithms [116,108,4,100,133,32,105] discussed herein use an intensity-based approach extracting geometric features, following the influential paper of Schmid and Mohr [101]. They seek to combine the virtues of feature detection and appearance modelling, *i.e.* geometric invariance based on the former and photometric invariance based on the latter approach. In a nutshell, they use the advantages of appearance-based methods but their system is stronger to spatial transformations due to the geometrical constraints which are imposed. The invariance does reduce the scope of the correspondence problem.

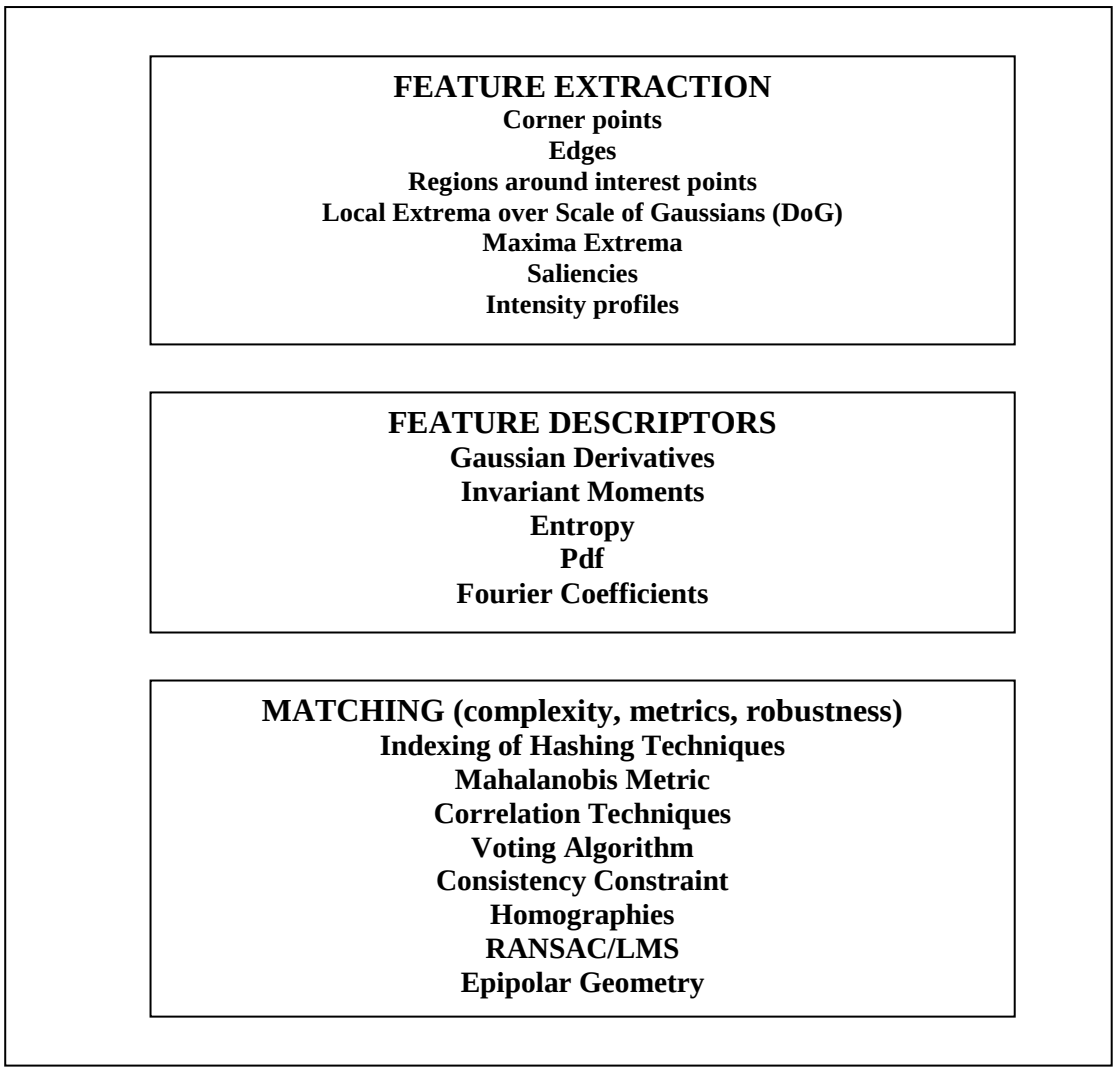


Figure 2.16. Wide-baseline blocks and methodologies.

The implementation of the extraction of the affinely invariant regions comprises the definition of the local regions around anchor points or landmarks. Both geometric and intensity-based methods have that in common. They search for anchor points which should be easily detected, produce stable invariant regions and most important, comply with the repeatability criterion (reliability on detecting the same anchor points with a strong independence on the changes in the imaging conditions). The selection of these points also avoids the analysis of every pixel and trims the complexity of the problem. The Harris-Stephen corner points[1] [52] are suggested as seed points by [101,116,108,4,133,32,69,110,105] and local intensity extrema by [108,110,116]. A study of the comparison of the performance of different methods can be found in [102]. This study reveals that the Harris corner detector provided better performance due to its

[1] For simplicity, the Harris-Stephens corner may be mentioned just as Harris corner in the text

repeatability under different transformations robust against rotations, translations and photometric changes) and high information content (or distinctiveness content, important for grey-value-based algorithms) than other detectors. The detection of Harris corners is generally carried out in a multi-scale fashion (scale-spatial Harris features). For the multi-scale Harris approach, there are two scale parameters: t which denotes the local scale at which the derivatives are calculated, and σ which is the integration scale in the second moment matrix. Baumberg [4] used t proportional to σ in his multi-scale wide-baseline approach. Local intensity extrema, however, cannot be located as accurately as Harris corners but can resist geometric change and any monotonic transformation of intensity levels. Besides, they do not usually lie near the border of objects, accomplishing better the planar constraint than Harris corner points.

Once these methods have found anchor points, many rely on other features. Fraundorfer and Bischof [32] started from Harris corners as anchor points and proposed a matching algorithm based on multi-scale salient operators introduced by Kadir and Brady [57,58] which are centred on the maximum entropy of features for saliency, scale and content description of image aspects (for an improved version of Kadir and Brady saliency detector refer to [100]). After extracting the Harris corners and the salient regions (circles with a defined diameter around these corners), *sub-salient regions* within the initially detected salient regions are obtained. The employment of *sub-salient regions* offers a deeper accuracy than salient regions, which also yields even better description than local interest points. In fact, salient features are the ones deemed to be difficult to be misclassified. The utilization of Harris corners instead of grey-level values as in [57] is reasonable for their aforesaid better geometric and photometric robustness. The authors also propose the combination of different descriptors as a matter of extraction of more information from the regions of the scene, for instance Gabor texture features. Saliencies in images are also estimated in [120] by means of a statistical analysis using the image histogram as a measure of the probability density function. In [108,110,116] the region extraction commences not only with the finding of the Harris corner points but also with the detection of the existing edges in the neighbourhood, performed by the Canny edge detector. Tell and Carlsson [105] extracted Harris corner points but did not define a region around them but formed pairs of interest points in order to trace a segment line between them and read the intensity profile along the line. They stated that points which are far away from each other are very likely to not accomplish the planarity constraint (the points are not co-planar). However, points which are too close

must have a lack of intensity information content along the line segment that matches them. Therefore, a threshold on distance value is fixed to determine the possible pairs of corners.

Pritchett and Zisserman [89] extracted four-line bounded regions to compute local planar homographies that restrict the search for correspondences. They match the parallelograms by exhaustive search and generate putative corner matches from those local homographies. Two strategies are defined: either consider only matches consistent with a single image transformation (global) or search for matches consistent with local homographies and use these homographies to search further matches. Matas *et al.* [79] introduced the novel concept of *Maximally Stable Extremal Regions* (MSER). These regions are invariant to affine transformations, are stable and allow multi-scale detection. The set of all extremal regions is of complexity $O(nlog(log(n)))$, with n the number of pixels. Intensity pixels are classified in order by their intensity value. Furthermore, a list with the group of connected components defines the detection of *distinguished regions* which have distinctive, invariant and stable properties. Then the maximally stable extremal regions are computed on the intensity image. Maximally stable extremal regions are produced by storing each connected component according to their intensity values. Components are merged, mixing the pixels of both components results in another set larger due to the combination of groups. At last, intensity levels which are local minima are selected as thresholds. It generally generates many small regions in order to be robust to occlusions and favour planarity of features. Some variations of MSER are [35] to work with colour and an expansion to affinities in [75].

Something worthy of note is the definition of a suitable local invariant region or window function. Once the region for calculating invariants in the reference image is calculated its affinely-invariant region (*i.e.* deformed) counterpart in the other image must be found in order to be able to describe invariants under the appropriate area to work in. Indeed, these regions must take into account the image transformation that the scene undergoes due to the different viewpoint. Small measurement regions have the advantage of better planarity but are less discriminative. Therefore, measurement regions must be relatively big but take into account the trade-off between discrimination and taking parts of the background absolutely different to the ones of interest. These measurement regions can be selected in a multi-scale way, *i.e.* the distinguished regions and scales of them are used to have discrimination of large regions and the planarity of

small ones. In [108] and [4] it is pointed out that these regions must be deformable somehow in order to cover the same area in both views. That is related to the concept of *affine Gaussian scale-space* developed by Lindeberg and Gårding [66] and [68]. We present some of the basic steps in their automatic scale detection. Let us consider the second moment descriptor μ_L:

$$\mu_L(.;\Sigma_t,\Sigma_s) = g(.;\Sigma_s) \otimes ((\nabla L)(.;\Sigma_t)(\nabla L)(.;\Sigma_t)) \tag{2.29}$$

with $L(.;\Sigma)$ the affine Gaussian scale-space representation of an image (.) and the covariance matrix Σ. Σ_s and Σ_t are the covariance of σ and t respectively. $g(x;\Sigma)$ is the Gaussian kernel,

$$g(x;\Sigma) = \frac{1}{2\pi\sqrt{\det\Sigma}} e^{\frac{-x^T\Sigma^{-1}x}{2}} \tag{2.30}$$

The use of these "affine Gaussian scale-space" elliptical windows can be used with associated covariance matrices producing affine scale-space to be generated by a linearly transformed elliptical Gaussian kernel instead of the conventional scale space which is usually generated by convolution with a rotationally symmetric Gaussian. The covariance matrices are adjusted iteratively and the second moment matrices (image descriptors) result invariant under affine transformation.

$$\begin{aligned} \mu_L(q_L;\Sigma_{t,L},\Sigma_{s,L}) &= M_L \\ \Sigma_{t,L} &= tM_L^{-1} \\ \Sigma_{s,L} &= sM_L^{-1} \end{aligned} \tag{2.31}$$

Then the square root of the second moment matrix M_L is used to transform the local image (equation (2.32)) and for the other image (equation (2.33)):

$$I_{L'}(M_L^{-\frac{1}{2}}x) = I_L(x) \tag{2.32}$$

$$I_{R'}(M_R^{-\frac{1}{2}}x) = I_R(x) \tag{2.33}$$

Lindeberg showed that under a linear transformation of image coordinates B, the following property for affine scale-space second moment matrices occurs:

$$\mu_L(q;\Sigma_t,\Sigma_s) = B^T \mu_R(B_q; B\Sigma_t B^T, B\Sigma_s B^T)B \tag{2.34}$$

For the normalized case:

$$\mu_{L''}(q';tI,sI) = I \tag{2.35}$$

with *I* the *2x2* identity matrix.

The transformation between $I_L^{'}$ and $I_R^{'}$ is a rotation *B'*:

$$I = \mu_{L'} = B'^T \mu_{R'} B' = B'^T B' \tag{2.36}$$

The process is iterated until the second moment matrix converges to the identity matrix *I*. Then there is a normalization for lighting changes. Finally, the effects of the rotation are cancelled by using rotation invariants.

Dufournaud *et al.* [27] presented a novel approach to attempt the matching of two images at different resolutions, up to a *6*-scale factor, where the high-resolution image is a small region of the low-resolution one. The high-resolution image is tackled by means of a scale-space interpretation, while the low-resolution one is not represented at different scales. The method detects interest points in both images and proposes for this purpose an improved version of the Harris corners detector, which is scale-space adapted for the wide scale factor between the images. Therefore, the matching lies in a one-to-many correspondence problem.

Lowe [72] proposed extrema over scale space filtered by difference of Gaussian (DoG) filters. The image is convolved with Gaussian filters at different scales and points of interest are detected as extrema within neighbourhoods of current and consecutive lower and higher scale. [73] improved the location of the interest points by finding the interpolation of the maximum when other extrema lie in the proximity. Then applying some thresholding low contrast extrema are rejected and the interpolated extremum kept as a feature. Still, the system also achieves better stability by suppressing features which are not well located but present high edge responses. Finally, the features attain invariance to rotation from dominant gradient orientations. In the next subsection we

complete the SIFT detector with its descriptor vector, which allows strong resistance to variations of illumination and affine transformations.

Forssén and Lowe [34] developed an affine invariant descriptor by computing SIFT over MSERs detected at the different scales of an image pyramid. The multi-resolution MSER attains higher scale invariance and contributes to the descriptor with robustness to illumination changes and local occlusions and the SIFT acts as a shape descriptor of the MSER. Nearby features are grouped in order to provide more prominence to features that repeatedly appear over many images of a dataset. The authors admit that it does not outperform SIFT over planar images but it does over 3D scenes. Obdrzadek and Matas [75] built *Local Affine Frames* (*LAF*) from MSERs. MSERs stem from local shapes in the image and from them there can be extracted geometric primitives that can constrain the six degrees of freedom that define an affinity. These geometric primitives are centre of gravity, curvature, covariance matrix of the region, directions, etc. and combinations of them define the LAF. Next a geometrical normalization of the region of measurement is performed from the change of every local affine frame with respect to the canonical reference system. The region is also normalized in photometry. The matching is performed by Euclidean distance between regions. The descriptor is affine invariant to geometric and photometric transformations. In [22], geometric hashing is used to matching LAFs.

Also inspired by SIFT, Bay et al. [5] used an approximation of the determinant of the Hessian as a detector of points of interest at different scales in their SURF (Speed Up Robust Features) descriptor. The Gaussian filters of the Hessian matrix are approximated by box filters, increasing the speed of the calculations and still achieving analogous results. The points of interest originate from non-maximum suppression over multi-scale neighbourhoods of the determinant of the Hessian matrix. The authors compare the repeatability of their detector with others, such as difference of Gaussians (DoG), the detector of SIFT and the Harris- and Hessian-Laplace detectors giving a better or at least comparable performance for the experiments run.

For a complete and recent reference upon local features extraction and descriptors see [65].

2.2.2 Feature descriptors and invariance

There are no general invariants when working with *2D* image points obtained from *3D* scene points - geometric invariance is almost always restricted to *2D* rotations and translations of planar objects; for instance circles become ellipses under affine transformations. Thus, [116] considers that many *3D* objects can be approximated in a local way by means of planar surface patches in order to use the *2D* invariants on the local scale selected. A similar approach was considered by [4], asserting that smooth surfaces can be locally approximated by planar surfaces. However, local regions on or near borders and occlusions do not fulfil the planarity constraint. Therefore they consider *2D* invariants as *"quasi-invariants"* when dealing with 3D objects. This latter assumption [7] permits the use of a variety of invariants for planar objects: moment invariants, algebraic invariants, differential and semi-differential invariants, Fourier invariants, reflectance ratios, Gaussian derivatives, etc.

Differential illumination invariants are used in [101], describing each interest point by a nine-dimensional rotation invariant vector of local characteristics. The Gaussian derivatives in the neighbourhood of the interest point allow invariance against rigid transformations between images. The set of derivatives is given by:

$$J^N[I](x,\sigma) = \left\{ L_{i_1 \ldots i_n}(x,\sigma) \in I \times \Re^+ ; n = 0 \ldots N \right\} \tag{2.37}$$

$$L_{i_1 \ldots i_n}(x,\sigma) = I \circ G_{i_1 \ldots i_n}(x,\sigma) \tag{2.38}$$

where $i_k \in \{x_1, x_2\}$ and the parameter σ denotes the smoothness effect of the Gaussian and also has to do with the next multi-scale approach step.

The set of invariants is calculated up to third order. The nine elements of the vector are computed according to:

$$V[0...8]=\begin{bmatrix} L \\ L_i L_j \\ L_i L_{ij} L_j \\ L_{ii} \\ L_{ij} L_{ji} \\ \varepsilon_{ij}(L_{jkl} L_i L_k L_{li} - L_{jkk} L_i L_l L_l \\ L_{iij} L_i L_k L_{kk} - L_{ijk} L_i L_j L_k \\ -\varepsilon_{ij} L_{jkl} L_i L_k L_l \\ L_{ijk} L_i L_j L_k \end{bmatrix} \tag{2.39}$$

with ε_{12}=- ε_{21}=1 and ε_{11}= ε_{22}=0.

So after that, a multi-scale approach is undertaken in order to be also insensitive to scale changes:

$$\begin{aligned} f(x) &= g(u), \\ g(u) &= g(u(x)) = g(\alpha x) \end{aligned} \tag{2.40}$$

In the multi-scale approach derivatives are described according to:

$$\int_{-\infty}^{\infty} I_1 G_{i_1 \ldots i_n}(\vec{x}, \sigma) d\vec{x} = \sigma^n \int_{-\infty}^{\infty} I_2 G_{i_1 \ldots i_2}(\vec{u}, \alpha\sigma) d\vec{u} \tag{2.41}$$

where $G_{i_1 \ldots i_2}$ are the Gaussian derivatives. In a discrete approximation, the size of the Gaussian and processing window are changed; scale quantization is a necessary condition for working with several scales. Therefore, the vector of invariant features is finally computed over several circular neighbourhoods of different sizes around the point of interest. However, Mohr's approach is not invariant to some general transformations, e.g. an affine transformation. Although this method is not wholly invariant it is worth mentioning since it set a strategy followed by other authors.

Zisserman and Schaffalitzky [133] stated that for viewpoint and photometric changes in a scene it suffices to reach an invariance of the description tools to geometric and photometric affine transformations of the geometry and intensity values of the image, respectively. Affine invariance has been pursued by [116,108,79,4,89,133,76,110] and [105], the last one even aiming at some projective distortions. Of these, [116,108,4,133]

used descriptors based on second moment matrices. Van Gool *et al.* [116] looked for geometrical invariance by bounding a region defined by two edges in the neighbourhood of a corner. There exist two cases according to the nature of the edges:

Curved edges. Starting from the Harris corner point and the two neighbour edges, two affinely invariant parameters *l1* and *l2* are defined using an arbitrary curve parameter (affine curve arc length, for instance) and the first derivatives of the edge *e1* and *e2* with respect to the curve parameter.

$$l_i = \int abs\left(\left| p_i^{(1)}(s_i)\, p - p_i(s_i) \right|\right) ds_i \qquad i = 1,2 \tag{2.42}$$

From the corner *p*, the two points move along the edges describing a parallelogram region *Ω(l)*, with *l* referring to *l1=l2* when a point in one edge *e1(l1)* is affinely invariant to the one in the other edge *e2(l2)*. Region *Ω(l)* where a given function(s) reaches its extrema in an invariant way for geometrical and photometrical variations is evaluated and searched. These are the functions:

$$\begin{aligned}
f_1(\Omega) &= \frac{M_{00}^1}{M_{00}^0} \\
f_2(\Omega) &= abs\left(\frac{\left| p - p_g \quad q - p_g \right|}{\left| p - p_1 \quad p - p_2 \right|} \right) \frac{M_{00}^1}{\sqrt{M_{00}^2 M_{00}^0 - (M_{00}^1)^2}} \\
f_3(\Omega) &= abs\left(\frac{\left| p_1 - p_g \quad q_2 - p_g \right|}{\left| p - p_1 \quad p - p_2 \right|} \right) \frac{M_{00}^1}{\sqrt{M_{00}^2 M_{00}^0 - (M_{00}^1)^2}}
\end{aligned} \tag{2.43}$$

$$M_{pq}^n = \int_\Omega \left[I(x,y) \right]^n x^p y^q dxdy \qquad p_g = \left(\frac{M_{10}^1}{M_{00}^1}, \frac{M_{01}^1}{M_{00}^1} \right)$$

The functions utilized are composed of two factors, a ratio of two areas, one of which depends on the centre of gravity weighted with intensity values of the local region, and an expression of moments up to the second order.

Straight edges. If the edges are straight (quite common), $l = 0$ and the method explained before cannot be applied. Then, the local extrema is sought in a 2D space with two

arbitrary parameters as co-ordinates, s_1 and s_2, for the two edges instead of the invariant parameter l. The two functions $f_2(\Omega)$ and $f_3(\Omega)$ are combined and the intersections of their two valleys selected to define the invariant region.

For objects with a lack of texture, the use of the above functions may fail due to the difficulty of the extraction of extrema. In this case, local extremum of $f_4(\Omega)$ is searched:

$$f_4(\Omega) = \frac{1}{xy}\left[\sum_{j=0}^{y} D_x I(x, y_j) \cdot \sum_{i=0}^{x} D_y I(x_i, y)\right] \tag{2.44}$$

where D_x and D_y are pixel differences and (x,y) co-ordinates on the straight edges.

A drawback is the possible difficulty of finding the same edges in the other image, for these can be non-connected, interrupted or connected differently. The intensity-based method which follows endeavours to compensate for this.

Photometric invariance takes into account changes in the lighting conditions of the different views of the scene. For their case, Van Gool and Tuytelaars prefer using a photometric invariant based on generalised colour moments, although the method can work with gray scale images, to obtain colour information in the neighbourhood extracted according to the aforementioned local region extraction, which should be more or less planar.

The intensity-based region extraction [110] is dependent on local extrema in intensity as the seed points. The rays which emanate from this local extremum are evaluated by working with the Euclidean arc length along the ray, the intensity and the intensity extremum:

$$f_1(t) = \frac{abs\left(\left|I(t) - I_0\right|\right)}{\max\left(\frac{\int_0^t abs}{\frac{abs\left(\left|I(t) - I_0\right|\right)}{t}}, d\right)} \tag{2.45}$$

The points where the rays reach an extremum are geometrically and photometrically affinely invariant. Extrema usually occur when the intensity changes severely along the

line. The points are all linked by enclosing an affinely invariant region. An elliptical surface surrounding this invariant region is created so that this elliptic region has the same moments (up to the second order) as the initial region. The authors doubled the elliptic region size, in a heuristic way, to ease the matching process but putting the planar restriction at risk.

As a conclusion, the geometry-based methods have problems since they depend on an accurate detection of the corners and edges. The intensity-based methods are also sensitive to noise in the case of weak extrema. Nevertheless, the experiments of the authors showed good performance in spite of the above said difficulty of accurate detection of local extrema. In short, finding reliable invariant regions in both images can be difficult due to false matches, non-planarity, perspective deformations, occlusions, and noise, although the methods are not designed for any special sort of images.

Strecha *et al.* [104] tackled multiple wide-baseline views matching. They extracted ellipses using the affine invariant method used in [110]. This way, the affine invariant ellipses are defined from Harris corner points and maxima extrema. The definition of this extrema can be compared with the point fingerprints concept [103]. Point fingerprints rely on the extraction of geodesic circles around points of interest on real range data. The projections of these geodesic circles onto the tangent plane are *2D* contours which are view invariant. Fingerprints must be discriminative enough so as to discern among a big set of features. Coming back to Strecha's ellipses, these try to cover planes although covering more than one as can be seen in the examples of the article. The areas of the ellipses are well-defined and also expanded. For example, keeping our attention on the first figure on the paper, it can be appreciated that there are not ellipses on the cover of the book on the shelf due to the extrema in there is very "diffuse" or "prominent" because of the existence of dense letters in the book cover. That might be a handicap for using it on sort of images like the ones we have to work with. The fact that the extrema does not take place close to the borders avoids discontinuities. The system works with colour moments, therefore if the context is restricted to gray intensity value images, a more convenient descriptor could be used instead.

Affinely invariant Fourier descriptors were used by [105] for intensity profiles across planar surfaces. Six Fourier coefficients are calculated:

$$f_m^{\sin} = \frac{1}{N}\sum_{i=0}^{N-1} p(i)\cdot \sin\left(\frac{2\pi mi}{N}\right)$$
$$f_m^{\cos} = \frac{1}{N}\sum_{i=0}^{N-1} p(i)\cdot \cos\left(\frac{2\pi mi}{N}\right) \qquad (2.46)$$
$$m = 1,2,3$$

where *p(i)* is the intensity profile and *N* is its length. As in the previous section, the segment between pairs of interest points should lie in the same plane. Every profile is normalized to its maximum intensity value in order to achieve affine photometric invariance (offset and scaling of profile).The authors declare that the use of affine invariance, which can be thought of as a weakness of their method if any other harder transformation occurs, is not problematic due to the possible distortions of the image. Usually there exist some directions within a plane which suffer only affine deformations. The algorithm looks for these affine deformations during the matching stage.

A review of classical and modern techniques based primarily on Fourier analysis for the problem of geometrical invariance can be found in [125]. They assumed that the nature of the invariance group is known *a priori*. The techniques use integral transforms, algebraic moments and neural networks in the invariance problem. Short-time Fourier analysis are also used in [3], together with wavelets and spline techniques. Illumination and invariance to affine transformations, noise, rigid motion and perspective transform is achieved. They state that their method, which works over colour and shape information over different scale levels, does not require the use of high-order derivatives. Fraundorfer and Bischof [32] worked with salient descriptors that are invariant to translation and rotation (calculation by histogram) and scale changes (multi-scale approach) and also robust to intensity and viewpoint image variations (corners are considered photometric invariants). They take into account the possibility of using several descriptors, assuming their combination will give more support to better discrimination of correct matches. They state that the method is scale invariant (*i.e.* it can work with different image resolutions) and can perform well for changes in viewpoint (0° to 40°). Recently, Escalera *et al.* [30] have extended the work in [57] by

combining their gray level entropy based saliency with a measurement of the entropy of histograms of orientation of regions. The authors claim that their detector shows a better repeatability than other state-of-the-art detectors.

Affine invariant texture descriptors were presented in [133] together with affine invariant point descriptors. Although many authors do not consider textures for the wide-baseline case since their repetitive pattern may produce many correlation peaks during the matching, Chetverikov and Matas [21] defended the convenience of dominant texture patterns for matching of regions. However, whether texture is of potential use, depends on the nature of the targets. For example, texture is not usually present in vehicles (unless camouflaged). We observe solely that texture descriptors do not require the finding of any invariant neighbourhood around interest points since it works itself with the statistics of the texture in the images.

It is also pertinent to mention the work of Weiss [121], which provided invariants related to the physical formation of images taken from different systems: IR, sonar, radar, etc. Physical invariants to translation and rotation are calculated by means of symmetries in images or in the imaging process (irradiances, energy conservation...), and are potentially useful to find correspondence points in the same or in different image modalities. Viola [112] studied sets of local complex features as a whole instead of single geometric features. These complex features are learnt from experience with model objects. It mentions *oriented energy* as a pre-processing tool to decrease the effect of photometrical and pose changes between different scenes.

The SIFT descriptor [73] expands the scale-rotation invariance of the detected extrema to quasi-invariance in changes of viewpoint and illumination. The method rotates the spatial coordinates of the region of interest according to the orientation computed in the detector, achieving orientation invariance. The magnitude of the gradient inside the region is smoothed with a Gaussian to reduce the effect of discontinuities and lower the weight of pixels close to the boundaries of the region of interest. The descriptor is composed of a *4x4* subdivision of the region, each containing orientation histograms of *8* bins. Thus the feature space is composed of 128 dimensions. The method is affine invariant to photometric changes since scaling the magnitude of the gradient gives scale invariance and also due to the fact that the gradient, as result of being pixel differences, is invariant itself to offsets in intensity levels. Besides, robustness to non-affine

invariant changes in illumination is achieved by giving more importance to gradient orientations and thresholding gradient magnitudes. Dorkó and Schmid [26] rely on SIFT for their Maximally Stable Local SIFT Description (MSLSD). Their method anchors to multi-scale Harris and Laplacian points and find for stable regions by using SIFT. Maximal stable regions are found where the change of the descriptor at consecutive scales is minimum. The descriptor benefits from the repeatability of the corner detector and from the robustness of SIFT to changes of illumination, invariance to rotation and noise.

A comparison of the main methods presented above is given in [83]. SURF is a novel scale and rotation invariant descriptor by Bay *et al.* [5] that extracts information inside a rectangular region centred at interest points detected by an approximation of the Hessian, as mentioned in the previous subsection. The rectangular region is oriented according to the output of Haar wavelets along the *x* and *y* directions of a circular region of a radius proportional to the scale at which the point of interest was detected. The descriptor is a *64*-element vector, of summations of Haar wavelet responses smoothed with a gaussian for spatial robustness. The authors indicate their descriptor has better level of performance than GLOH, SIFT and PCA-SIFT for the images they tested, and especially surpasses in lower computational time.

2.2.3 Complexity, metrics and robustness of the matching

Vincent and Laganière [113] assessed some different matching strategies for validation of constraints established in matching algorithms. These are *unicity* (for each feature point, only the strongest match in the other image is considered), *symmetry* (the relation between matches should be a reciprocal correspondence) and *confidence measure* (the similarity of the matches should be similar to the ones of their neighbours, *i.e.* both features of the match should have a neighbourhood with alike properties). They proposed the *disparity gradient* as a measure of the compatibility between pairs of features. Zitová [131] also added *invariance* (both features of the match should be described by the same descriptor), *uniqueness* (different features should have different descriptors, related to *symmetry*), *stability* (small deformations of the feature should be closely described like the initial feature) and *independence* (the elements of descriptor vector should be independent).

Complexity. An exhaustive search to compare feature vectors between images, or alternatively between an image and a pre-formed database such as a *DTM*, has complexity $O(n^2)$ where n is the number of features. Where necessary, features can be stored in a data structure such as a kd-tree to perform efficient storage and fast access to the matching features, e.g. $O(nlogn)$. This can be very important, for example, when searching for corresponding features in large databases of aerial or other photographs, in order to perform registration and difference comparison to detect changes in ground movement.

A kd-tree is a data structure for storing k-dimensional points. Figure *2.17* shows an example of the structure of a kd-tree for the case of the spatial distribution of *3D* points. The range of the values in each dimension are (x_{min},x_{max}), (y_{min},y_{max}) and (z_{min},z_{max}) in the *3D* case, and with the median as criterion. A first partition is done according whether the *x*, then *y* and *z*, co-ordinate is greater than the median. The procedure is iterated cyclically, until all the sub-volumes are empty. Therefore, the structure stores the k-dimensional points in sub-volumes according to the median criterion.

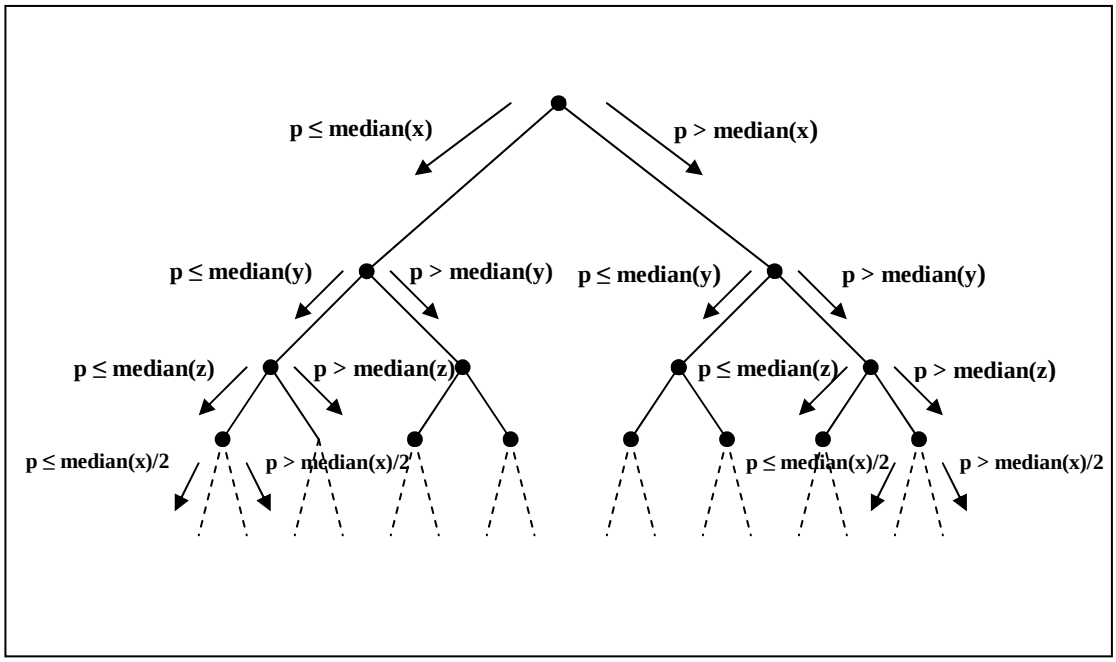


Figure 2.17. Kd-tree structure for 3D-space points.

Geometric hashing [32,22] is also used to match feature vectors in data bases.

Metrics. A cost function is generally minimized to estimate the projective transformation or homography. The function can be for example the Mahalanobis distance or an algebraic distance. The Mahalanobis distance metric is used by [101,116, 108,4,133,110,105] to assess the similarity of invariant vectors. The expression is given by:

$$d_M(b,a) = \sqrt{(b-a)^T \Lambda^{-1} (b-a)} \tag{2.47}$$

That measure considers random variables with Gaussian distribution as well as their covariance matrix Λ to give an estimation for the comparison of the vectors. The square of d_M is a random variable which follows a χ^2 distribution. There is an option to set a threshold to $d_M(b,a)$ and reject a certain percentage of the matches which are deemed false.

In contrast, Matas *et al.* [79] considered the Mahalanobis distance as not reliable enough since a single corrupted data may ruin the match. They believe their mapping is robust enough; they gain advantage from the distinctiveness of large regions which are not very affected by non-planar constraints and the use of a voting system. Schmid and Mohr [101] imposed a geometric constraint to reject possible false matches by establishing a threshold of consistency. The geometric constraint used is an algebraic distance under a given threshold (equation *2.48*). The affine transformation between two affinely invariant regions describes an approximation of the projective transformation which defines a nine-dimensional space of *3x3* matrices (equation *2.14*). The geometric constraint is defined as (with δ_g denoting the threshold):

$$\det \begin{pmatrix} a_{23}-b_{23} & b_{13}-a_{13} & a_{13}b_{23}-b_{13}a_{23} \\ a_{22}-b_{22} & b_{12}-a_{12} & a_{12}b_{23}-b_{13}a_{22}+a_{13}b_{22}-b_{12}a_{23} \\ a_{21}-b_{21} & b_{11}-a_{11} & a_{11}b_{23}-b_{13}a_{21}+a_{13}b_{21}-b_{11}a_{23} \end{pmatrix} \leq \delta_g \tag{2.48}$$

The metric evaluates the distances between the descriptor vectors encoding distinctive characteristics. That generates a confusion matrix of distances between descriptors from both images. A *voting algorithm* [101,116,108,79,4,22,105] was used for selecting

tentative correspondences from that confusion matrix. The distance metric, being *b* and *a* in equations *2.47* and *2.48*, the vectors at the reference and sensed image respectively. For every model M_A^i for a region *A* and an invariant descriptor *i*, the *k* nearest models in the other image $M_{B_j}^i$ (j=1…k) from *k* regions in the image are found. All the models similar to M_A^i are given a vote every time the distance result is below an arbitrary threshold. This way, the model which gets the largest number of votes is selected as the best one. The experiments in [79] work with *216* invariants and *4* scales, a total of *864* invariants. The authors assert that the experimentations showed good performance for a value of *k* of *1%* of the number of distinctive regions.

Robustness. An initial set of correspondences has been already estimated. Notwithstanding, this set is under an approximate (not exact) solution which point-position errors are assumed to describe a Gaussian distribution. Nevertheless, practical situations show the existence of outliers or high-disturbing mismatches which do not follow the Gaussian distribution but may follow any other. They should be detected in order to compute the homography only with the set of inliers within the set of initial correspondences. Robust estimation algorithms such as RANSAC, Least Median of Squares (LMS) or M-estimators are used for this purpose. These algorithms are able to deal with a large proportion of outliers.

Therefore after voting, some methods opt for the epipolar geometry (Appendix *A.3*) to reduce the scope of the matching problem. Pritchett and Zisserman [89] stated that many algorithms which use epipolar geometry with no other support fail in the wide view case. The use of homographies allows the definition of a viewpoint invariant affinity measure as well as a reduction of the complexity of the search when putative corner matches are created. A *3D* scene structure, together with the epipolar geometry, defines the many local homographies that exist in an image pair. Their algorithm generates the homographies between pairs of images and sets of putative (parallelogram) matches are verified. The fundamental matrix (representing the epipolar geometry) and a consistent set of matches are calculated using *RANSAC* (RANdom Sample Consensus) which selects a subset of these matches which are consistent with the homography. Warping by a homography makes cross-correlation geometrically invariant. Therefore, putative parallelogram matches are verified by means of the projective homography and calculating the cross-correlation of the projectively warped

region enclosed by the parallelogram. It is worth mentioning that this approach is highly dependent on the geometry of the image scene, since it relies on the existence of well-defined parallelograms and large planar regions for feature extraction. In [79], some randomly selected potential matches are also modelled by correlation techniques using the centres of gravity. After the application of *RANSAC* to these, coarse epipolar geometry is estimated. Nevertheless, *RANSAC* is applied another time in a very narrow threshold and finer epipolar geometry utilized once more to the remaining good matches after the second application of *RANSAC*. Baumberg [4] identified potential matches and found putative correspondences by means of ambiguity measures. It was argued that the number of successful matches is greater than the number of mismatches. The last step in the method is also the application of the epipolar constraint to eliminate the few outliers still remaining. Fraundorfer and Bischof [32] also extracted the epipolar geometry from regions of interest (saliencies). Zisserman and Schaffalitzky [133] verified matches using the Lucas-Kanade algorithm and other matches found from the obtained homographies. They also apply RANSAC algorithm to select the correct matches for the epipolar geometry extraction.

Tell and Carlsson [105] asserted that the sieve of mismatches created by directly applying RANSAC and then epipolar geometry could be computationally intensive. So the authors propose to establish a *consistency constraint* in order to reduce the number of false matches still remaining. To set this constraint they supposed they knew the camera model. Knowing the model of the camera and with a set of interest points, they constrained the coordinate points of their counterparts in the other image by applying equation (equation *2.51*) once they have eliminated camera parameters from the epipolar constraint.

They use the scaled orthographic camera model and five points randomly extracted from two regions, each one from every image.

$$\begin{aligned} a_{i,j} &= \left(x_i^a - x_1^a\right)\left(x_j^a - x_1^a\right) + \left(y_i^a - y_1^a\right)\left(y_j^a - y_1^a\right) \\ A_i &= \left(a_{i2}, a_{i3}, a_{i4}\right)^T \end{aligned} \tag{2.49}$$

$$b_{i,j} = \left(x_i^b - x_1^b\right)\left(x_j^b - x_1^b\right) + \left(y_i^b - y_1^b\right)\left(y_j^b - y_1^b\right)$$
$$B_i = \left(b_{i2}, b_{i3}, b_{i4}\right)^T \tag{2.50}$$

$$\begin{bmatrix} [B_2A_3A_4]+[A_2B_3A_4]+[A_2A_3B_4] & [B_2B_3A_4]+[B_2A_3B_4]+[A_2B_3B_4] \\ [B_2A_3A_5]+[A_2B_3A_5]+[A_2A_3B_5] & [B_2B_3A_5]+[B_2A_3B_5]+[A_2B_3B_5] \end{bmatrix} = 0 \tag{2.51}$$

where x_k^a and x_k^b for *k=[1…m]* denote the *m* points extracted for a region *A* in the sensed image and for a region *B* in the reference one respectively. *[.]* is the determinant.

When the points are not mismatches, the data follow the constraint in (*2.51*). Then a counter for every match is increased. The process starts again selecting other five points randomly and keeps on iterating. It stops when an average level of increments reaches a threshold. With this method they presumed the cancellation of *50%* of the outliers.

Finally, RANSAC and the epipolar are estimated for the reduced group of matches. Their experimental results were based on *400* corners from each image. Most of the time complexity is due to the data structure (kd-tree) used for storing feature vectors. The algorithm fails for reflective surfaces – recall it is based on intensity profiles – and some curved objects – there is a need for a planarity constraint. However, it carries out good behaviour for projective transformations of the image since it works with lines between many points allowing therefore the search for not very distorted lines.

Photometric and geometric changes and noise are responsible for mismatches. Some kind of constraint should be imposed in order to maintain the affine invariance and immune to these undesirable effects. Paying attention to the distribution of the features of the profiles [105] in the image by using the covariance matrix of all the features, allows some discrimination between vectors. However, it does not work for intensity changes. The magnitude of the distribution of a profile feature has some relation to the distribution of the feature over the whole image. The variance of the features is proportional to the diagonal elements of the covariance matrix of all the feature vectors in the image. The proportionality constant allows the distinction between feature vectors, being more discriminative for small values. To avoid many matches of some feature vectors to others, a normal distribution of the feature vectors is considered

and the ones which are close to the mean are discarded for the matching stage for they are very likely to have many matches.

2.3 Summary

A brief introduction to 3D-to-2D camera projections has been presented. These are not the only projections existent, the two images also undergo deformations between them, 2D homographies deal with them.

Section 2.1.3 recalls and updates the work in [131], which provides a wide overview of general methods for image registration. In this article, the state-of-the-art is organised according to the nature of the methods (appearance-based or feature-based) classifying them into the stages that are common to all registration tasks. These are: the extraction of a feature space, the matching of the descriptors defined by the characteristics detected in the raw images and finally, the transformation model used to establish the final correspondence. Evaluation of the overall results can be performed to refine the final outcome.

Appearance-based methods are generally less complicated to implement, offer a dense mapping, which is useful for a smooth reconstruction, and work well with textured images. They present the inconvenience of being invariant to small geometric image distortions; for instance most of them can only cope with translations, a simple rotation prevents satisfactory results. However, despite that descriptors such as correlation ratios which similarity measure can only deal with rotations and translation; these have shown to perform good results in multi-modal applications. In the same way, very promising research based on mutual information methods has been developed in the last years. Therefore, these methods can be of great help when implemented together with feature-based methods; for the latter can offer a better contribution to achieve a coarse-to-finer counterpart matches search (reducing the spatial transformation problem) and the former contribute to the intensity transformation problem.

Feature-based methods take advantage when the image has distinctive objects, features or details to be detected. These methods are more robust to photometric changes in the scene and usually have a faster response since they do not have to process the whole

image. The methods that use invariant descriptors have centred the attention of many authors due to the interest in finding or characterising features on images that do not change under certain photometric or geometric transformations. Nonetheless, the full knowledge of the transformation is fundamental.

Both the consideration of constraints in the matching and the use of pyramid techniques for multi-scale approaches are welcome and highly desired when working with both area-based or feature-based methods. The use of optimisation techniques to maximise the similarity cost function between two templates is essential.

The second part of this chapter dealt with wide-baseline methods for image registration. The following tables show a taxonomy that summarizes some of the most important approaches to the correspondence problem for wide-baseline scenarios. The majority of methods rely on interest points that can be reliably matched between images. Generally this means that they are easily extracted, repeatable, have high information content, and if possible are invariant to the relevant geometric and photometric transformations. Most methods rely on Harris corners as seed points, as they fulfil many of these criteria, at least where there are not significant photometric changes. However, these feature-based methods themselves are effective when the displacement between frames is small and a local window can suffice to finding correspondences. The majority of methods look for photometric support, typically around these anchor points, such as intensity extrema or intensity profiles. This photometric support is searched within a quasi-planar local region (or line segment). This region should again be invariant to the geometric and photometric distortions that occur in the images. The definition of this invariant area is difficult, yet fundamental. The assumption of planarity to match between images is a major limitation. Planarity is very useful because finding region correspondences based on planar homographies is much easier. However, many of the most significant points in image data occur precisely where this planar constraint is violated.

METHOD	FEATURES DETECTED	LOCAL/ GLOBAL	DESCRIP-TOR	INVARI-ANT TO	MATCHING METHOD	NOVEL CONCEPT	IMAGE	OBSERVATIONS
Van Gool & Tuytelaars [116, 108]	Harris corners, Canny edges and local intensity extrema	Paralle-logram local regions	Moment Invariants	Affine transf., occlusions, partial visibility, scene clutter, wide baseline and photometric changes	Mahalanobis, cross-correlation, homographies and voting algorithm	Local affinely invariant regions	Wide baseline 3D indoor and outdoor scenes	Difficulty for finding edges for the geometric method Uncalibrated camera conditions. Quasi-invariant planar surfaces)
Schmid & Mohr [101]	Intensities and Harris corners	Local circular neighbo-urhoods	Gaussian deriva-tives	Occlusions, rotations, scales and viewpoint	Mahalanobis, voting algorithm and indexing techniques	Definition of regions around anchor points	Greyscale paintings, 2D, aerial and 3D	Short baseline Multi-scale approach
Matas et al. [79]	Extremal properties of intensities	Local planar regions	Complex moments	Affine transf., scale(3.5x), illumination, rotation, occlusion and translation	Robust similarity measure, voting system, correlation techniques, RANSAC and epipolar geom.	-Maximally Stable Extremal Regions -Robust Similarity Measure	Wide baseline 3D indoor and outdoor scenes	Stable and multi-scale detection for wide-baseline stereo case Extended to colour in [35]
Walker, Cootes & Taylor [120]	Feature vectors and obtaining of saliencies	Pixel level	Probabili-ty density function of feature vectors		Density of feature space		Faces	Hard calculation

METHOD	FEATURES DETECTED	LOCAL/ GLOBAL	DESCRIP-TOR	INVARI-ANT TO	MATCHING METHOD	NOVEL CONCEPT	IMAGES	OBSERVATIONS
Lowe [72]	Intensity extrema over scale-space by DoG filters	Local	Gaussian deriva-tives	Scale, translation, rotation and partially invariant to lighting, affine and 3Ddistortion	Modification of k-d tree algorithm Hough transform and hash table	Scale invariant feature transform	Indoor dense scene of 3D objects	Only partially invariant to lighting, affine distortion Works with a scale space and feature vectors
Baumberg [4]	Harris corners	Local regions around interest points	Second moment matrices	Wide-baseline, scaling, affine and lighting changes	Mahalanobis, ambiguity measure scores, epipolar geometry	Affine gaussian scale-space (Lindeberg et al. [66])	Objects Wide-baseline (15°-65°)	It fails for wide angle views (65°) It uses an iterative procedure [66] for the finding of the optimal invariant window
Zisserman & Schaffa-litzky [133]	Harris corners	Local invariant regions	Second moment matrices	Viewpoint and lighting affine changes, scaling	Mahalanobis, Lucas-Kanade algorithm, homographies, RANSAC and epipolar geometry		Wide-baseline outdoor scenes (church)	Extension of Baumberg's work Two methods: affine invariant point and texture descriptor
Fraundorfer & Bischof [32]	Harris corners and saliencies	Local regions around corners	Entropy	Rotation and scale and robust to intensity and viewpoint changes	Geometric hashing (and epipolar geometry)	Sub-salient regions	Outdoor images (church and objects)	Not absolutely (robust) invariant to intensity and view-point Multi-scale method

METHOD	FEATURES DETECTED	LOCAL/ GLOBAL	DESCRIP-TOR	INVARI-ANT TO	MATCHING METHOD	NOVEL CONCEPT	IMAGES	OBSERVATIONS
Pritchett & Zisserman [89]	4-line bounded regions	Local planar paralle-lograms		Affine transforma-tions	RANSAC and homographies		Synthetic image of a house	Parallelograms and planar regions must be present in the image
Tell & Carlsson [105]	Harris corners and intensity profiles	Local planar regions	Fourier coefficient	Photometrica l and affine changes - even some projective distortions	Mahalanobis distance, voting algorithm, consistency constraint, RANSAC and epipolar geometry	Consistency constraint [18]	Example pictures are indoor objects	Able to face some projective distortions. High-computational cost of the kd-tree. Needs planar surfaces and distinctive regions (no constant brightness). Consistency constraint method.
Lowe [73] SIFT	Same as [72] and interpolates and thresholds extrema	Local	Gaussian deriva-tives	Same as [72] and adds robustness to non-affine light changes	Modification of k-d tree algorithm Hough transform and hash table	Scale Invariant Feature Transform	Indoor dense scene of 3D objects and outdoors	Improves the estability of [72] Widely used Achieves highest accuracies.
Bay *et al.* [5] SURF	Approxima-tion of the determinant of the Hessian	Local within interest point neighbourh ood	Hessian	Invariant to scale and rotation and strong to photometric changes	Thresholded euclidean distance	Approximati on of the Hessian with box filters	- Indoors and outdoors. - Oxford sequence [86]	Comparable or even outperforms state of the art in accuracy and especially in speed.

METHOD	FEATURES DETECTED	LOCAL/ GLOBAL	DESCRIP-TOR	INVARI-ANT TO	MATCHING METHOD	NOVEL CONCEPT	IMAGES	OBSERVATIONS
Escalera *et al.* [30]	Salient regions from intensities and gradient orientations	Local	Entropy	Robust to viewpoint changes	Complexity score	Complex Salient Regions	Caltech database [16] and outdoors	Better repeatability under changes of scale, rotation, light and affinities than Harris, Hessian Laplace and gray level saliency detectors
Dorkó & Schmid [26]	- Harris and Laplacian points. - Maximally Stable Local SIFT regions	Local	Harris, Laplacian and SIFT	Affine changes of viewpoint and illumination	Nearest neighbour	Stable region based on SIFT	Oxford sequence [86]	Better matching and repeatability than Harris and Laplacian points.
Forssén and Lowe [34]	-MSER -SIFT	Local	Multi-Resolution MSER and SIFT	Robust to illumination, occlusions and invariance of MSER and SIFT	Dissimilarity score	Multi-resolution MSER and combination with SIFT	- Outdoor and indoor sequences. - Oxford sequence [86]	MSER improved against scale changes
Obdrzalek and Matas [75]	-MSER -Geometric primitives	Local	Local Affine Frame (LAF)	Affine geometric and photometric transformat-ions	Similarity measure (Euclidean distance)	Affine frames from geometric features	Synthetic, indoor and outdoor sequen-ces	No comparison with main state-of-the-art methods ([83]).

Chapter 3 – Extraction of features

3.1 Introduction

This chapter presents the employed methodology and experimental results for the extraction of shape information from images. Figure *3.1* shows the organization of the chapter. Contours are built by grouping edges using some perceptual organization rules. These contours are labelled in terms of their closeness and curvature, which assists with the search of intersections among contours and also, in Appendix A, as a test of suitability for the analysis of contours in the frequency domain. The contours are also partitioned into straight segments in order to facilitate the task of finding intersections among contours from the projection of straight, endpoint segments and, also, in order to delimit a ribbon-like region for the analysis of the photometry at both sides of the contours. We present some experiments about the extraction of these regions around the contours but the method is discarded due to its inherent lower reliability compared to the affine invariant approach presented in the next chapter. We also perform a spline approximation of contours used to compute metrics in Chapter *4*.

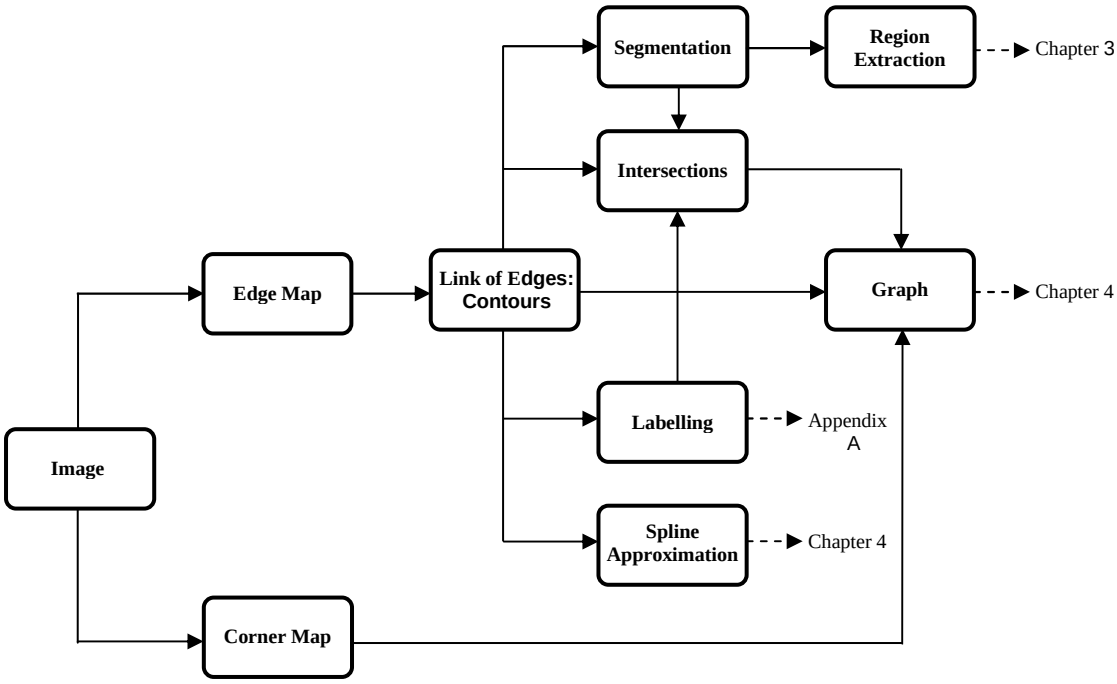


Figure 3.1. Organization of the chapter.

A corner map from the original image is also extracted. The data from this map, together with the contours and the intersections found, is reorganized in a graph. This graph approach, containing the spatial relations between points of interest interconnected by contours, is the output of this chapter. The structural information which is relevant in the image is preserved in the graph and used in the next chapter to define affine invariant regions where to analyse the photometry.

3.2 Contours

This sub section deals with the preliminary step of extracting contour information from the input images. It can be argued that the understanding of contours, boundaries or shape cognition is inherent to human visual perception for the interpretation, classification and/or identification of our surrounding world. By analogy, in Computer Vision the use of contours is also very sensible since they provide robustness in geometry against changes of the conditions of illumination, in particular because their dependence is not directly related. Moreover, the computational complexity is drastically reduced as a result of not considering the processing of the totality of the pixels of the image or patches of it. This is a significant difference comparing with another subfamily of feature-based methods such as region matching - outline plus the interior intensity information. When comparing with other primitive features such as corners, edges, *et cetera*, contours also possess the definite advantage that they are higher-level entities that conglomerate added informational content. On the other hand, boundary information can be sensitive to noise and occlusion. Structural methods treat features as composed of sub-features, and can better handle partial occlusions.

3.2.1 Extraction of edges

The process starts with the detection of edges from the images by using the widely used Canny edge detector [17] that extracts discontinuities in image intensities, which are likely to correspond to structural parts of the scene. The image is smoothed by convolving it with a Gaussian filter in order to reduce the effects of noise and perform a multi-scale analysis. The magnitude and direction of the gradient over the smoothed data is computed from spatial derivatives:

$$\Gamma = \sqrt{\Gamma_x^2 + \Gamma_y^2}$$

$$\Theta = \arctan\left(\frac{\Gamma_y}{\Gamma_x}\right)$$

The direction of the gradient Θ is quantized to *0°*, *90°*, *45°* and *135°* in order to trace the edge within the *8*-connected image grid. The detector optimises a thin edge response by applying non-maximum suppression over local pixels in the direction of the gradient, *i.e.* a pixel is considered as edge if its magnitude gradient is greater than the gradient in the direction perpendicular to its quantised direction of the gradient.

Rather than using a single threshold to discern pixels of higher edge response, the algorithm carries out a hysteresis thresholding that is stronger against pixel gradient values drifting around a single threshold and causing, therefore, discontinuous detections along the edges. Thus if the gradient magnitude is lower than a threshold t_{low} the pixel is discarded as a part of the edge, whereas it is considered as an edge pixel when its magnitude is higher than a threshold t_{high} and also whenever a pixel gradient is higher than t_{low} and is connected to a pixel already deemed as an edge (figure *3.2*).

An example and all the internal steps of the detector are shown in the plots of figure *3.3*. The input image is a grey level image with a resolution of *646x527* pixels. The parameter σ of the Gaussian filter is *1* and the hysteresis thresholds are set to *0.025* and *0.062*. There are *144* edges found. Contours are traced to form longer and more reliable features. As will be explained in subsequent subsections, according to the gradient and direction maps, proximity, continuity and certain distance constraints contours are linked with each other to form more significant and informative entities. At the same time, short contours, less than a minimum length are discarded. The result is an improved version of the Canny edge map. Figure *3.4* shows the contour map after tracing and linking contours.

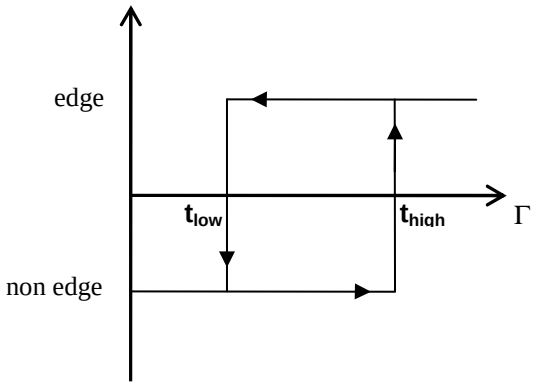


Figure 3.2. Hysteresys.

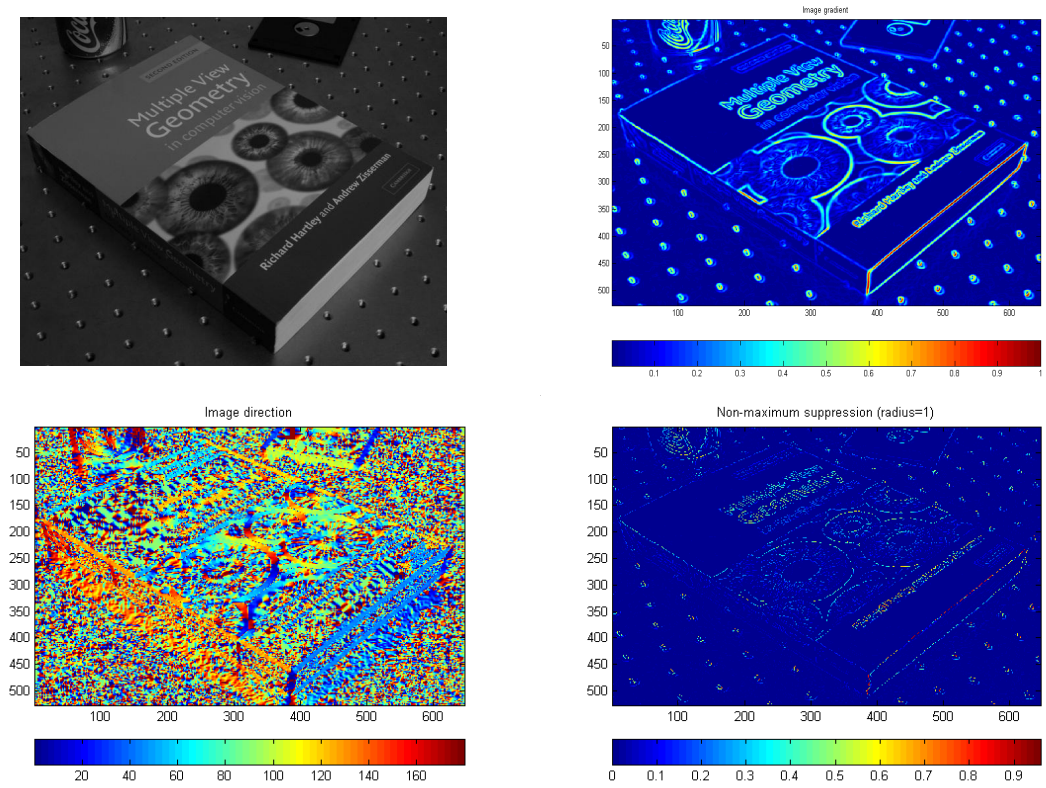


Figure 3.3. Canny edge detection. a) original image, b) magnitude map, c) direction map, and d) non-maximum suppression.

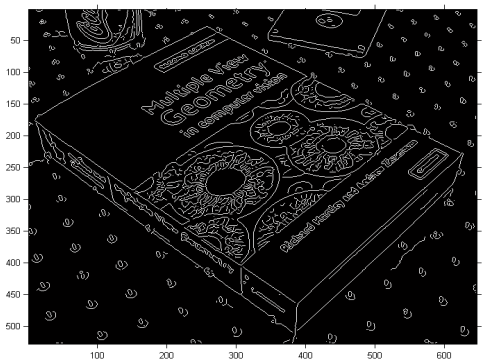


Figure 3.4. Canny edge map.

3.2.2 Linking of edges

An edge map extracted from an edge detector does not usually provide by itself meaningful information about the structure of the scene. Edges are quite sensitive to noise and changes of illumination and can result badly connected in relatively complex images. Therefore, edges are generally linked to form higher, more informative entities (contours) by using normally some local, systematic, cognitive biases (section *3.2.2.1*). Other approaches, however, use global techniques to link edges such as the Hough transform or graphs [46]. Indeed, we also process a further contour linking by organizing the information in a graph structure as it will be presented in section *3.5*.

Our procedure for tracing contours of complex shapes is based on the method used in [119]. The starting point of each contour is assigned to the strongest point of a thresholded gradient magnitude map. The contour is traced by searching for the next point with strongest gradient magnitude which is within the *8*-pixel neighbourhood and which is also within a certain angular marching direction given by the direction of the gradient. Once the end is reached the contour is traced back and labelled till reaching the starting point where the procedure starts again tracing in the other direction. Figure *3.5* shows the contour map for the input image from figure *3.4*.

Figure 3.5. Linked contours

3.2.2.1 Perceptual grouping

We extend the method above by adding some perceptual grouping cues. That leads to the cognitive theory of *Gestalt*. The *Theory of Gestalt* was developed by Max Wertheimer [122] in the *20s* of the past century. It is a descriptive theory in modern psychology that states that the operation of the human brain aims for global perception rather than processing smaller components in isolation. The stimuli are interpreted according to perceptual laws that are dependent on each other and are called *Gestalt laws*.

These laws are centred mostly in the visual domain and we will only adjust to a short definition incumbent upon our application. The *Law of Prägnanz,* which generalizes the concept, declares that the information perceived is organized in such a way so as to have as much simplicity as possible. “Incomplete” images are completed according to how we perceive the world. These natural laws about perceptual grouping are:

- The law of proximity. Similar stimuli or elements in the proximity tend to be perceived as a unique instance.
- The law of good continuation. Elements that follow a certain pattern (e.g. curvature) are considered as linked.
- The law of similarity. Elements sharing similar properties (e.g. colour, or orientation) can be grouped into a single set.
- The law of closure. Perception completes figures that are not closed, by adding the missing parts.

Early work on perceptual grouping in Computer Vision dates back to [78,128] with works on grouping features into larger structures. Lowe [71] proposed a measure of *significance*, which quantifies in terms of proximity, parallelism and collinearity how likely straight lines may belong to the structure of the original scene rather than to viewpoint projections. More recently Elder [29] performed Bayesian statistical analysis over position, length and luminance along a contour based on Gestalt cues for its correct extraction. For us, the purpose of organizing in a human-like fashion the information that a computer has to process is not to provide the computer a higher, human-like ability of abstraction but to organise the data in higher and more meaningful entities. For that intention, the perceptual grouping laws can be a tool for grouping contours or

form larger clusters to describe in a more discriminative way characteristics in the scene.

We present a variation where once an endpoint is reached a search window is opened and the endpoints of neighbouring contours in the vicinity are sought. Both neighbouring contours are then bridged, by alleviating the threshold restriction during the edge detection process, and relabelled thus forming a single and more informative entity. When more than one endpoint of neighbouring contours is found, the one corresponding to the longest contour is preferred. The principle of proximity is used when opening the searching window, whereas good continuation and similarity are reflected by the consistency within a certain angle tolerance of consecutive points. In terms of proximity and collinearity, the measure of significance proposed by Lowe [71] is of importance as a tolerance to accept or reject parts within the structure. The significance measure on the basis of proximity of endpoints is the inverse of *N*:

$$N = \frac{2D\pi r^2}{l^2}$$

being *D* a scale-independent density of line segments (set to *1* and not relevant since all line segments will be rated by *D*), r the radius of the searching neighbourhood and l the length of the contour.

And on the basis of collinearity the significance is the inverse of *E*:

$$E = \frac{4D\theta s(g + l_1)}{\pi l_1^2}$$

with θ the angle between both allegedly collinear segments ($\theta=0$ if collinear), *s* the perpendicular distance from the midpoint of the shortest segment to the projection of the other segment, *g* the gap distance between both segments and l_1 the shortest segment. Figure *3.6* shows the graphical representation of the significance in terms of proximity and collinearity.

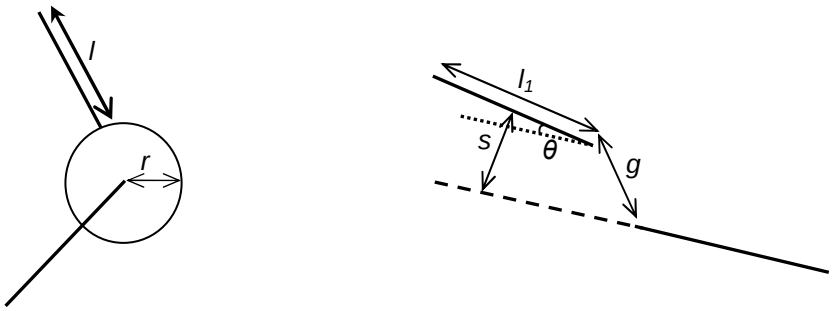


Figure 3.6. Perceptual organization of segments: a) proximity and b) collinearity.

3.2.3 Segmentation of contours

Edge/contour information can be represented by approximations of linear and/or higher order splines (Section *3.2.5*). In some applications this can be considered as a much handier way to deal with the spatial coordinates of the edge features, thereby encoding the interconnections of the contour segments. For instance, we make use of segmentation of contours in order to define intersections between contours and, also, to define a photometric region at both sides along the contours. This is showed in section 3.3, although this approach is discarded in the final system and the affine invariant approach presented in Chapter *4* is preferred.

Rosin and West [93] segmented contours by using combinations of straight lines, circular, elliptical and superelliptical arcs, and polynomial curves. They claimed the process allows reduction of data (storing only vertex coordinates), it is not dependent of any initial parameter and the representation achieves invariance to *2D* rigid transformations since the segments are normalized by the length of the curve. Their method links the two endpoints of the contour and computes the point of maximum deviation of the curve with respect to that line linking both endpoints. Each endpoint is linked to this point of maximum deviation and the process is repeated to each one of the primitives (see figure *3.7*). The process iterates by calculating again the maximum deviation for every sub feature until it stops due to impossibility to represent the feature. All the sub features are stored in a tree structure and assigned a *significance value* measure. According to this, the features primitives are selected by visiting the nodes of the tree. The final contour segmentation is shown in figure *3.8*.

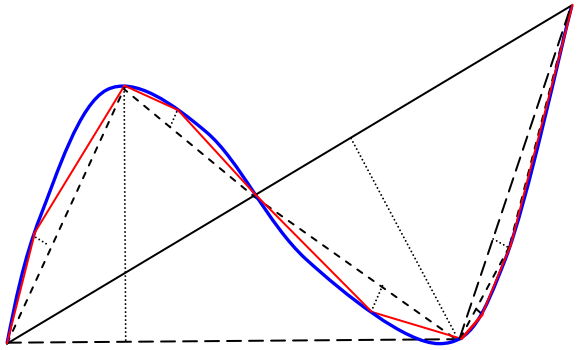


Figure 3.7. Recursive curve segmentation. Original contour in blue, final segmentation in red.

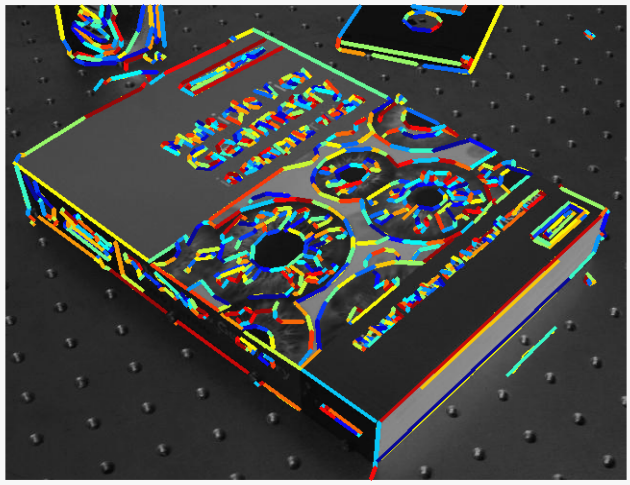


Figure 3.8. Contour segmentation.

For the simplest case of straight segment approximation, the algorithm simply returns the endpoints of all the segments. In our case, we require the spatial coordinates of the segments of the contours, *i.e.* the pixels that correspond to a piecewise linear approximation (otherwise just keep the points that segment the contours). The simplest way of finding the pixels of the segments is simply using the equation of the line and then rounding the values. However, a much reliable option from the computer graphics literature is the Bresenham's line drawing algorithm [12]. An example is shown in figure *3.9*.

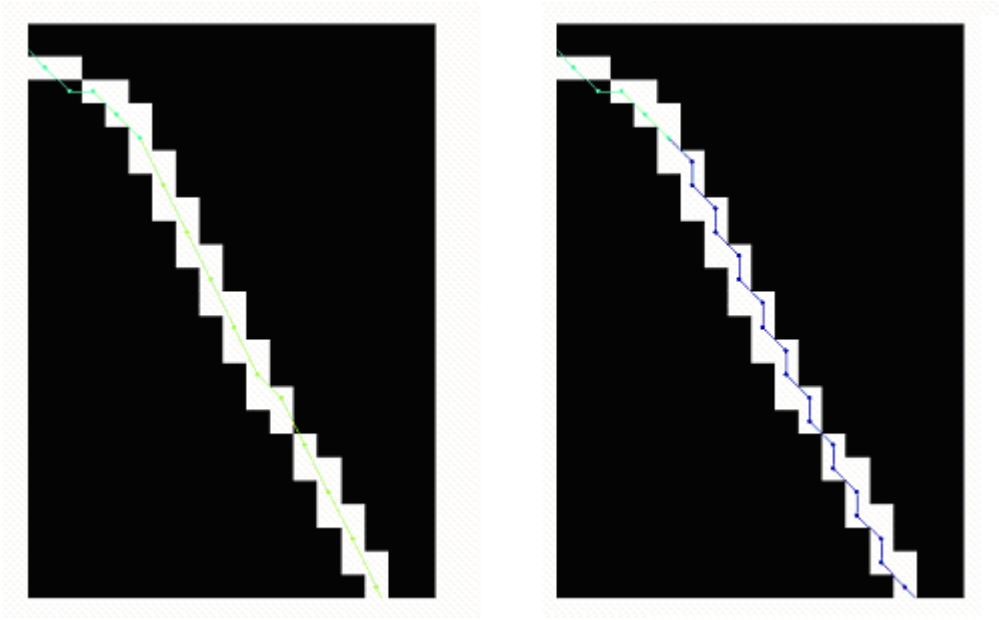


Figure 3.9. Example of Brenseham's method. In white original line pixels. In coloured dots, pixels coordinates plotted after: rounds (left) and Bresenham's line drawing (right).

3.2.4 Labelling

First of all, let us establish a few definitions:

- A "line" is a contour that is approximated by a single linear segment.
- A "curve" is a contour that is approximated by multiple linear segments, or by a single curved segment, or by a combination of linear and curved segments.
- A "closed" curve is one in which the start and end point are the same. Note, that there may happen the case of *loop contours*, *i.e.* a contour that could be segmented as a closed contour plus, at least, one adjoined open segment curve.

We are confident that the tie points resulting from the intersections of two straight lines can be much more reliable than the intersection between two curves or one straight line and one curve. Therefore, it could be sensible to establish a certain priority order during the computation process, or even weight signatures lying on the crossing of straight segments. For the case of a contour composed of straight and curvilinear segments, the contour segmentation performed before can help to characterise the intersection where the interest point lies on according to the order of the spline of the curves that define it. Consequently, the curvature of the contours (or the primitives that intersect) is

calculated and each point of interest is classified according to the nature of the curves that produce it, *i.e.* a straight line or curved line.

The other issue to be concerned of is whether the contour is closed or open. This is related to the use of the Fourier-based matching algorithm developed in Chapter *5*, where there is a preference in the use of the code on closed contours. That is due to the need for periodicity when working in the Fourier domain. Figure *3.10* shows the four types of labelling.

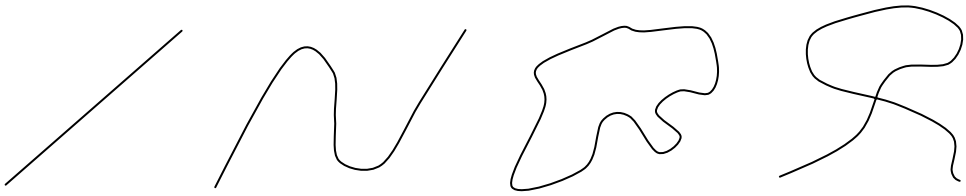


Figure 3.10. Labelling. a) line, b) curve, c) closed curve and d) loop contour.

3.2.5 Approximation by splines

In Chapter 4 we will require the spatial derivatives of the contours detected during the feature extraction process in order to compute our affine invariant operator. These derivatives can be computed by means of finite differences between samples. However, these can be very noisy and unreliable for up to second order derivatives. A better approach consists of approximating the contour by splines and computing the derivatives of these spline curves after.

Splines are piece-wise polynomial functions that permit a flexible design for shaping different curves smoothly. Among the different existing schemes in the literature of splines, we will focus on some aspects of our interest. Although the origins of these curves date back to the work of Lobachevsky in the nineteenth century, their modern conception as a curve approximant is due to the work of Schoenberg [98]. Some years after, the recurrence relations promoted by C. de Boor, M. Cox and L. Mansfield meant the appearance of more effective algorithms for B-spline calculations [11,33,99].

Definition. Let $S(t)$ be a parametric curve whose domain is defined in a finite interval $[a\ b]$ and subdivided by a strictly increasing sequence $U=[u_0 \leq u_1 \leq \ldots \leq u_{m-1} \leq u_m]$. These

$m+1$ elements of U are called the *knots* and the interval $[u_i\ u_{i+1})$, delimited by each u_i $\Delta i \in [a \quad b]$, is called the *knot span*. If r successive knots have coincident value, they are called knots of *multiplicity* r, otherwise they are *simple*. Notice that, therefore, multiple knots imply a null knot span.

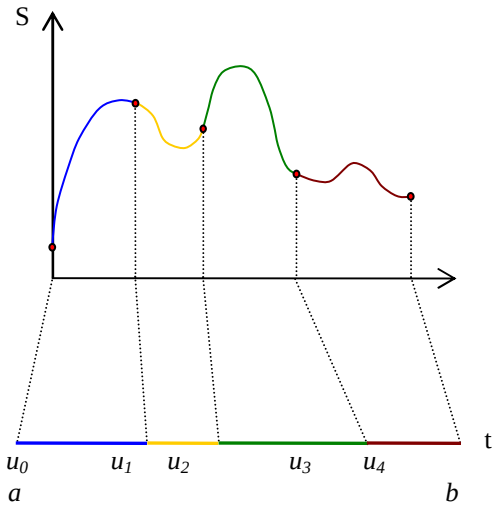


Figure 3.11. Input curve, knots and knot span.

A spline curve $S(t)$ of degree $k>0$, *i.e.* order $k+1$, is composed of piecewise polynomials of degree k called B-spline or basic splines. These basic spline functions, $\Delta t \in [a \quad b]$, are defined by the *de Boor-Cox recurrence relations*:

$$N_{j,0}(t) = \begin{cases} 1 & \text{if } u_j \le t < u_{j+1} \\ 0 & \text{otherwise} \end{cases}$$
$$N_{j,k}(t) = \frac{t-u_j}{u_{j+k}-u_j} N_{j,k-1}(t) + \frac{u_{j+k+1}-t}{u_{j+k+1}-u_{j+1}} N_{j+1,k-1}(t) \qquad (3.1)$$

$N_{j,k}(t)$ is non-zero in the interval $[u_j\ u_{j+k+1})$ and vanishes outside of it. A consequence of this is that within any knot span $[u_j\ u_{j+1})$ there are at most $k+1$ non-zero B-spline functions of degree k. The sum of all of these is unity:

$$\sum_{j=i-k+1}^{i} N_{j,k}(t) = 1 \text{ on span } [u_i\ u_{i+1}] \tag{3.2}$$

A linear combination of these B-spline functions forms the spline *S(t)*:

$$S(t) = \sum_{j=0}^{h} N_{j,k}(t) \cdot p(j) \tag{3.3}$$

where $p=[P_0,P_1,\ldots,P_{h-1},P_h]$ are the *B-spline coefficients* of *S(t)*. These coefficients are also called *control points* and represent the points of a control polygon, which defines the spline curve. The number of control points is *h+1*. There exists a relation between *h*, the order of the spline *(k)* and the number of elements of the knot sequence *U*, *(m+1)*:

$$h = m - k - 1 \tag{3.4}$$

The practical scenario is that the number of control points *(h+1)* is set by choosing *h* according to $n \geq h \geq k \geq 1$, *n* being the number of (parameterized) input samples. Therefore, the number of elements *m* of the knot sequence *U* is given by:

$$m = h + k + 1 \tag{3.5}$$

Figure *3.12* illustrates the control polygon of a spline to approximate a sine curve. Notice that in this example the spline rather than fitting the input data points approximates the virtual curve defined by these. That is due to our requirements of implementation. Spline approximation is introduced at the end of this section.

Since the conditions of continuity are given by the difference of the order of the spline and the multiplicity of every single knot, that affects the differentiability of the spline curve in a given knot. Therefore, a cubic spline (order *4*) would have only continuity of function for a knot with multiplicity of *3*, whereas it would have continuity and first derivative in a knot of multiplicity of *2*. Likewise, a knot of multiplicity *4* would imply no continuity not even in the function. As aforementioned, the knot sequence should be non-decreasing. When the first and last knots are simple *(multiplicity = 1)*, the spline curve is said to be *open* as its ends do not match the first and final control points, P_0 and P_m. However, if we fix the initial and last knots to a multiplicity *k+1*, *i.e.* a knot

sequence $U = [u_0 = u_1 = \ldots = u_k \leq u_{k+1} \leq \ldots \leq u_{m-k-1} \leq u_{m-k} = \ldots = u_{m-1} = u_m]$, the spline curve is *clamped* and starts and ends at both extremes of the control polygon[2]. The value of u_0 and u_m can be arbitrarily assigned values *0* and *1*, respectively, or set to the boundary conditions *a* and *b*.

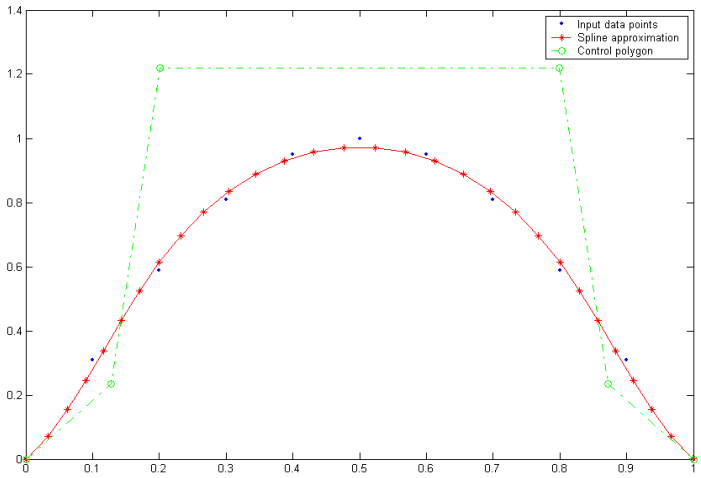


Figure 3.12. Example of data approximation by splines.

In any case the *m-2k-1*, *i.e. n-k*, remaining central knots can be either chosen equally spaced or dependent on the parametric vector of the input data. In the former case the definition of the vector for the uniformly spaced method is obvious:

$$\begin{aligned} u_0 &= a \qquad & u_n &= b \\ u_i &= a + i\frac{b-a}{n} \qquad & \text{for } 1 \leq i \leq n-1 \end{aligned} \tag{3.6}$$

Concerning the possible parameterizations of the input data points $D_0 \ldots D_n$, briefly these are the expressions of three widely used methods:

- The Chord Length method:

$$\begin{aligned} t_0 &= a \qquad & t_n &= b \\ t_i &= a + \frac{\sum_{k=1}^{i} |D_k - D_{k-1}|}{\sum_{k=1}^{n} |D_k - D_{k-1}|}(b-a) \quad \text{for } 1 \leq i \leq n-1 \end{aligned} \tag{3.7}$$

[2] Multiplicities *k+1* produce division by zero in the calculation of B-splines $N_{i,k}(t)$. As $N_{i,0}(t)$ can be zero, the case *0/0* is considered *0*.

- The Centripetal method:

$$t_0 = a \qquad t_n = b$$

$$t_i = a + \frac{\sum_{k=1}^{i} \left| D_k - D_{k-1} \right|^c}{\sum_{k=1}^{n} \left| D_k - D_{k-1} \right|^c} (b-a) \quad \text{for } 1 \le i \le n-1 \qquad (3.8)$$

where c is typically chosen as 0.5 (square root)

The interior knots can be the result of an average of the parameters.

$$u_i = avg\left(\left[t_{i-k} \quad \ldots \quad t_{i-1} \right] \right) \qquad for\ i = k+1, \ldots, m-k-1$$

Despite that the knot vector can be defined as a uniformly spaced sequence or as a function of the parametric version of the inputs, e.g. the average, there exists another strategy which also involves both the knot sequence and the parametric vector. It is called the *Universal method* or *Lim's method*. In that case the parametric vector, although also related to the knot sequence, is not needed for the definition of the knot sequence. Conversely, the knot sequence is allocated as a uniformly spaced vector (multiple knots are respected) and the parameterization is given by distance along the input data curve where the *n+1* B-spline functions defined by the equally spaced knot sequence peak. Searching for the maximum of every B-spline function, although a *1D* search, can involve a considerable computational effort. Shene [99] states that a few samples on each B-spline and assigning the abscise of each maximum to the corresponding *t* can suffice (figure *3.13*). Moreover, Lin's method has proved to be affine invariant. Actually, B-splines themselves are also invariant to affine transformations. Affinely transformed points can have their curve recovered providing the same knot and parametric vectors. Notwithstanding, interpolation/approximation methods using parameterizations like chord length or centripetal are not affine invariant anymore as they depend on the length of the segment. That is not the case of the uniformly spaced method. Even though a simple method, it is invariant as the knot sequence is equally spaced and thus the same in both images. Therefore, that invariance does require that every input data in one image is the exact affine map of its counterpart in the other image. Realistically this affine invariance property does no longer exists unless the contour map of the second image is affinely transformed from the contour

map in the original image. Therefore, this technique would increase the computational load of our system, gaining little or none affine invariance in a practice.

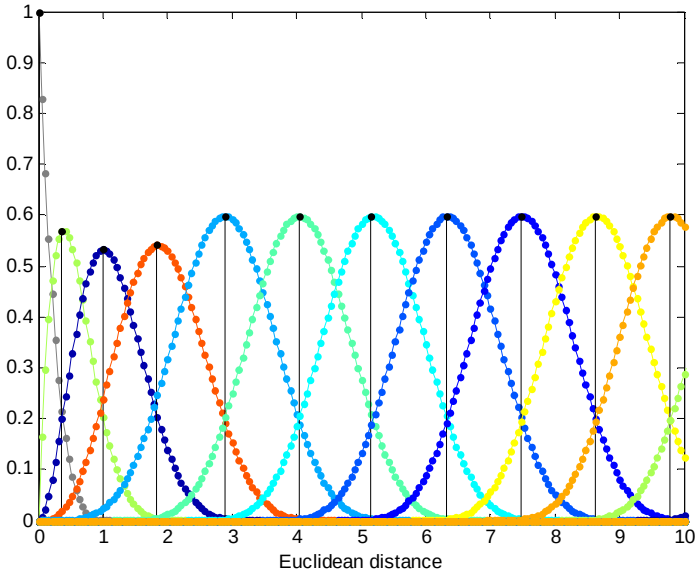


Figure 3.13. Universal method. B-splines and parameterization.

Splines as approximants. When fitting a spline to every given sample for data interpolation, the output can be different than that desired, such as a wiggled outcome around the input data. However if we smooth the accuracy requirements, we can permit a certain error and perform an approximation. In that case, the curve does not pass through every given data point but at a certain distance bounded by an error. The restriction of null error at the curve endpoints is kept. Therefore, the curve should track the control polygon within a distance. Note that that closeness of the curve to the control polygon is dependent on the order of the curve. Lower order curves track closer the polygon.

The least-square criterion is widely used as an approximant in the bibliography of splines. It consists of finding the control points $p=[P_0 \ldots P_h]$ that minimize the sum of squares of the deviation between the input data points and the resultant curve:

$$\delta(P_0,...,P_h)=\sum_{i=1}^{n-1}\left|D_i-S(t_i)\right|^2 \tag{3.9}$$

Since we establish as boundary conditions $S(t_0)=D_0$ and $S(t_n)=D_n$

$$\begin{aligned} D_i-S(t_i)&=D_i-\left[N_{0,k}(t_i)D_0+\left(\sum_{j=1}^{h-1}N_{j,k}(t_i)P_j\right)+N_{h,k}(t_i)D_n\right]= \\ &=\left(D_i-N_{0,k}(t_i)D_0-N_{h,k}(t_i)D_n\right)-\left(\sum_{j=1}^{h-1}N_{j,k}(t_i)P_j\right) \end{aligned} \tag{3.10}$$

Hence let us define:

$$Q_i=D_i-N_{0,k}(t_i)D_0-N_{h,k}(t_i)D_n \tag{3.11}$$

and the vector Q and the matrix N:

$$Q=\begin{bmatrix}\sum_{i=1}^{n-1}N_{1,k}(t_i)Q_i\\ .\\ .\\ .\\ \sum_{i=1}^{n-1}N_{h-1,k}(t_i)Q_i\end{bmatrix} \qquad N(i,j)=\begin{bmatrix}N_{1,k}(t_1) & ... & N_{h-1,k}(t_1)\\ . & ... & .\\ . & ... & .\\ . & ... & .\\ N_{1,k}(t_{n-1}) & ... & N_{h-1,k}(t_{n-1})\end{bmatrix} \qquad \mathrm{P}=\begin{bmatrix}P_0\\ .\\ .\\ .\\ P_{h-1}\end{bmatrix} \tag{3.12}$$

$$(N^TN)P=Q \tag{3.13}$$

$$P=(N^TN)\setminus \mathrm{Q} \tag{3.14}$$

Algorithm

Input: Data points $D=[D_0 \ldots D_n]$, new vector to interpolate X

Output: The spline $S(X)$

Procedure:

- Compute some parameterization t of the data
- Extract the knot sequence U
- Calculate the B-splines $N_{j,k}(t)$
- Compute Qi, $N(i,j)$ and Q
- Obtain the control points P
- Set the new interpolating sequence and parameterise
- Calculate B-splines $N_{j,k}(X)$ for the previous knot sequence U
- Compute the spline curve as $S(X) = \sum_{j=0}^{h} N_{j,k}(X) \cdot p(j)$

Figure *3.14* shows a comparison between a parametric spline interpolant and least-squares spline approximation to contour samples. The figure is a zoom-out over one of the contours in the book scene of figure *3.3*. See that the spline oscillates at both sides of the least-squares spline solution, being less precise.

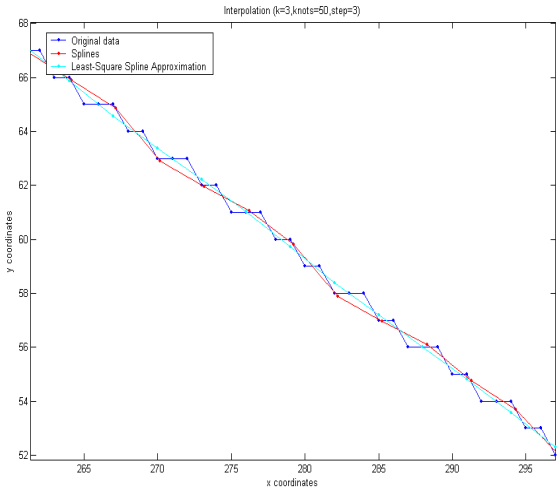


Figure 3.14. Comparison of spline fitting and least-squares approximation. a) Planar contour and spline fitting and approximation by least-squares. b) Zoom.

Derivative of a spline. The derivative of a spline *S(t)* is given by:

$$S(t)' = \sum_{i=0}^{n-1} N_{i+1,p-1}(t) Q_i \tag{3.15}$$

being:

$$Q_i = \frac{k}{t_{i+k+1} - t_{i+1}} (P_{i+1} - P_i) \tag{3.16}$$

For a clamped spline, *S'(0)* and *S'(n-1)* should be:

$$Q_0 = \frac{k}{t_{k+1}} (P_1 - P_0) \tag{3.17}$$

$$Q_{n-1} = \frac{k}{1 - t_{m-k-1}} (P_n - P_{n-1}) \tag{3.18}$$

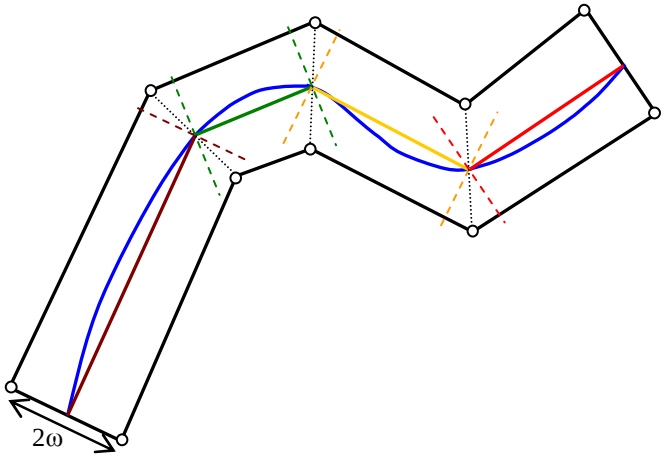
3.3 Extraction of regions around contours

We are aware that some works have performed registration based on only contours. For instance [71] did model matching from contours segmented into straight lines. However we consider that the use of only geometric contours cannot suffice for registering wide-baseline scenarios and we look for further support based on the photometry of the scene. Contour maps from dense scenes may contain a plethora of similar contours that together with the changes in viewpoint harden the matching. If we add some further support to our features such as, ideally, a photometric descriptor invariant to the lighting conditions in the scene, the search space of correspondences can diminish considerably. Herein, we propose the extraction of photometric information surrounding contours, obtaining a ribbon-like patch. Thus, the photometry and the geometry of the contour can be combined in order to extract more informative features for the matching process.

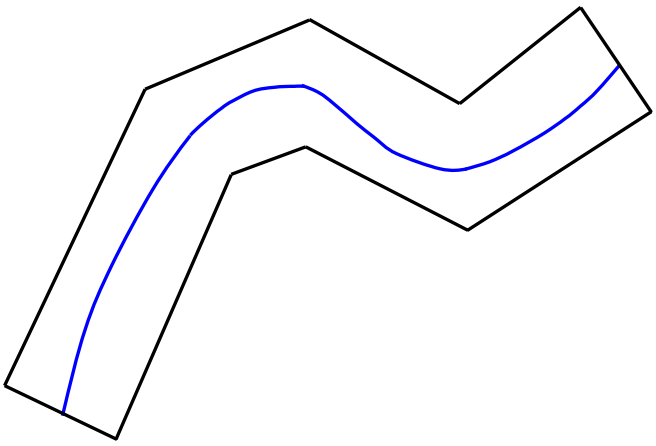
To extract a ribbon - a patch around a contour - the first step is to perform a segmentation of the contour. The contour segmentation in section *3.2.3* returns the endpoints of the feature primitives (segments) of the contour. For every endpoint that defines a segment we calculate points at a certain perpendicular distance at both sides of the contour. For the case of a single straight segment just computing the point at a

certain distance ω in the perpendicular to our segment would suffice. However, two consecutive segments will form an angle different to *180º*. Then the median of the perpendicular vector of the current segment with the perpendicular vector of previous and next segments, respectively at both endpoints of the current segment, will delineate guide landmarks that track the contour. Figure 3.15 shows a graphical representation.

Figure 3.16 shows a practical example. The contour map of an intensity image is depicted in figure *3.16a)*. The contour that concerns us in this example is highlighted in red. The output segments given by the segmentation are marked as blue asterisks.



(a)



(b)

Figure 3.15. Schematic of a ribbon. a) Original contour in blue, segmented contour and their perpendiculars in other colours, the circles represent the guide landmarks and final ribbon in solid black, b) contour and ribbon.

The red circles are the guide landmarks that track the contour at both sides. The ribbon is defined by the dotted yellow outline, which links the red-circled-guide landmarks by using the Bresenham's algorithm [12]. Finally, the green circles represent the pixels that are taken inside the ribbon to sample homogeneous photometry along the ribbon. Intersection of other contours with the contour of interest will segment the ribbon, "labelling" different photometric regions.

Notice in figure *3.16a)* that the contour also plays the role of a *"median strip"* inside the ribbon, defining what we call the *bright region of interest (broi)* and the *dark region of interest (droi)*, where to extract colour information. Pixel average is used to distinguish darker from brighter ribbons. Figure *3.16b)* shows the same patch with the contour and the outline close area. We start with the first sample inside the internal ribbon, shown as a magenta dot, and from this location we make the effect of flooding a whole close region. That region is delimited by the outline, the contour of interest and any other contour intersecting. Assuming that the contour map is accurately extracted, we can say that we are extracting a homogeneous photometric region around one of the flanks of the contour under inspection. Figure *3.18a)* shows that from the first location a whole homogeneous photometric region is filled. Travelling and flooding for next samples inside the internal region makes no effect since that region has already been filled by the first sample. See in figure *3.18b)* that finally one of the samples is located in a region still empty and can fill a new region defining another photometric patch (figure *3.18c*). Figure *3.18d* shows the result after performing the same steps for the external side of the contour. Internal and external homogeneous photometric regions in false colour are shown in the two bottom images.

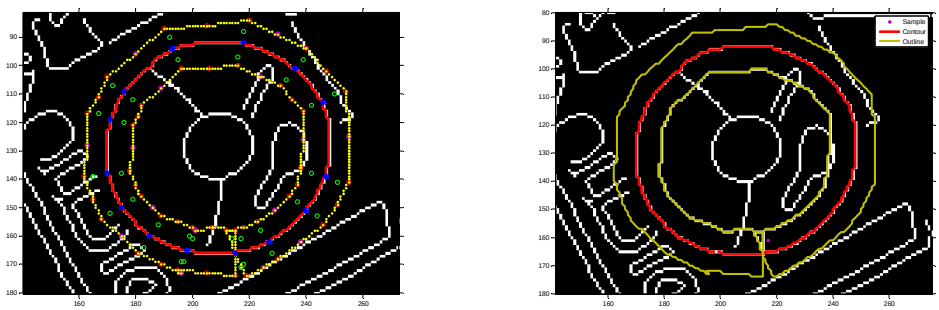


Figure 3.16. Extraction of regions of interest at both sides of contours. a) process of extracting region around a contour; and b) final region extracted

Figure 3.17. Input image

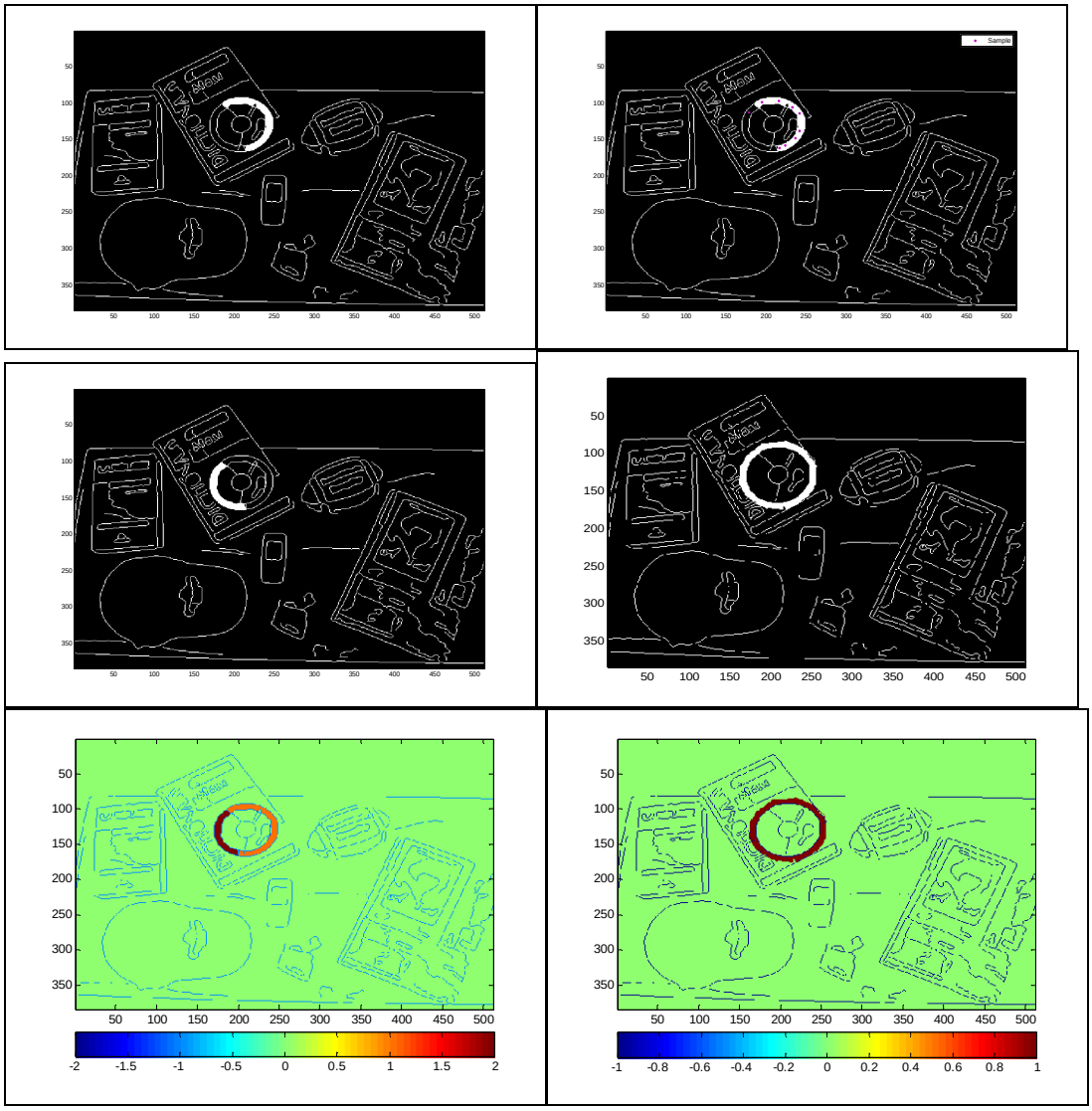


Figure 3.18. Extraction of homogeneous photometric regions. *a), b), c)* and *e)* internal flank; *d)* and *f)* external flank.

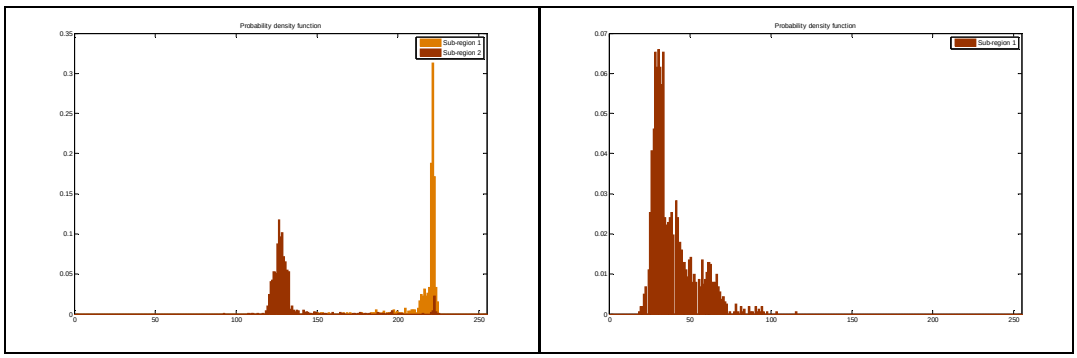


Figure 3.19. Histograms. a) Internal and b) external regions.

The histograms corresponding to the internal and external regions are shown in figure *3.19*. Notice that these sub-regions, and consequently the homogeneity of intensities, are strongly dependent on the edge detection and contour intersections. As an example, one of the contours does not intersect for few pixels the contour under inspection. The consequence is that two non-homogeneous intensity regions are not well separated. The probability density function in figure 3.*19a* shows that some pixels are classified as sub-region *2* while they belong to sub-region *1*.

We can organize the extraction of regions in different ways, namely:

i) Extract regions around whole contours and perform some *RGB* averages, entropy, etc. and define a descriptor. Figure *3.20* shows an example of the extraction of regions. The drawback is that, although we are adding photometric support, the method is still very dependent on the detection of the contours, their breaks and occlusions.

ii) Extract points of interest (corners, etc) that lie over contours. The ribbons at both sides of the contour emanate from the point of interest until they are intersected by other contours. It is also dependent on the extraction and intersection of contours but in a lesser extent than the above mentioned strategy since the ribbons are better delimited by points of interest – these are presumably more reliable than contour endpoints.

An input image and its transformed version are shown in Figure *3.21*. Points of interest are extracted by hand in this instance. Figure *3.22* shows a conglomerate of plots that represent for each row the ribbon(s) that emanate

from each point of interest. The first two columns correspond to the *droi* and *broi* regions of the original image, whereas the last two columns are the *droi* and *broi* regions of the transformed image.

The extraction of regions along contours based on ribbons is quite heuristic and will not be considered as part of the final system since a most elegant approach is presented in the next chapter. The parameter ω is only invariant to translations and rotations. A simple change of scale would imply that the regions extracted along corresponding contours in both images would not correspond to each other. However, if the contour map is able to separate different photometric regions in an efficient way; the overlap of the contour map with the ribbon would delimit regions with homogeneous photometry, *i.e.* same photometry although non-corresponding geometric regions. That could be valid for images where the photometry of the image can be easily segmented due to well-differentiate photometric regions (for instance, images with lighting conditions under control and well-distinguishable man-made objects). The region around each contour does not invade other photometric regions as far as contours are well extracted, no matter the transformation between the images. However, the assumption of being able to segment regions of homogenous photometry from contours is weak in complex, natural images.

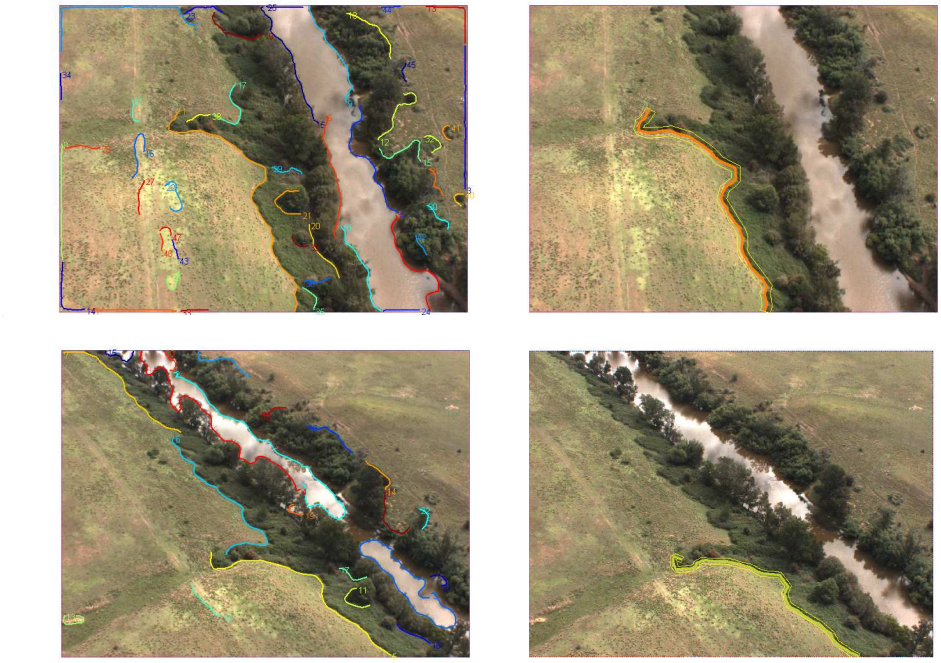


Figure 3.20. Contour and region extraction over a wide-baseline countryside setting. Left images: contour map. Right images: region extraction around the longest contour.

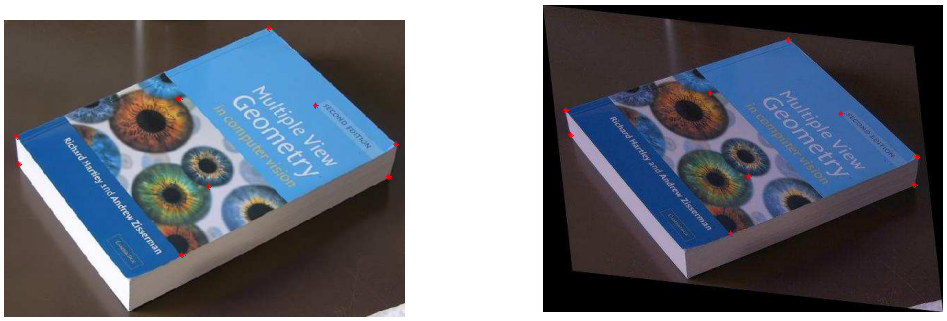


Figure 3.21. Left, original image and points of interest manually extracted. Right, affinely transformed image (0.9 and 0.1 geometric scale and shear, respectively with 0.7, 0.65 and 0.75 RGB scale)

3.4 Intersection and corner criteria

Intersection. We can define intersections (between open or closed contours) of the kind:

- Line-Line. This is a point that is (a) the intersection of two infinite lines, (b) exists within the image, and (c) is within a radius (*i.e.* "near" to each finite line segment. By definition, lines are "open".
- Line-Curve. This is a point that is (a) the intersection of the infinite line and one of the curve segments, (b) exists within the image (c) is within a radius.
- Curve-Curve: this is a point that is (a) the intersection of a segment on one curve with a segment on another curve, (b) exists within the image (c) is within a radius.

There exists the restriction that the projection of an end segment of a contour can never originate an intersection if it intersects itself previously.

We find intersections with other contours by opening over both endpoints a circular window where to search for a neighbour contour to intersect. That is implemented in the way described in figure *3.6* for perceptual group based on circular proximity. Figure *3.23* and *3.24* illustrate the process and restrictions imposed. Notice that for this illustrative example, the dimensions of the window have been magnified making them proportional to the length of the contour, with the only aim of easing the visualization of the circular regions. Figure *3.25* shows the propagation of the contours and the intersections found in the image.

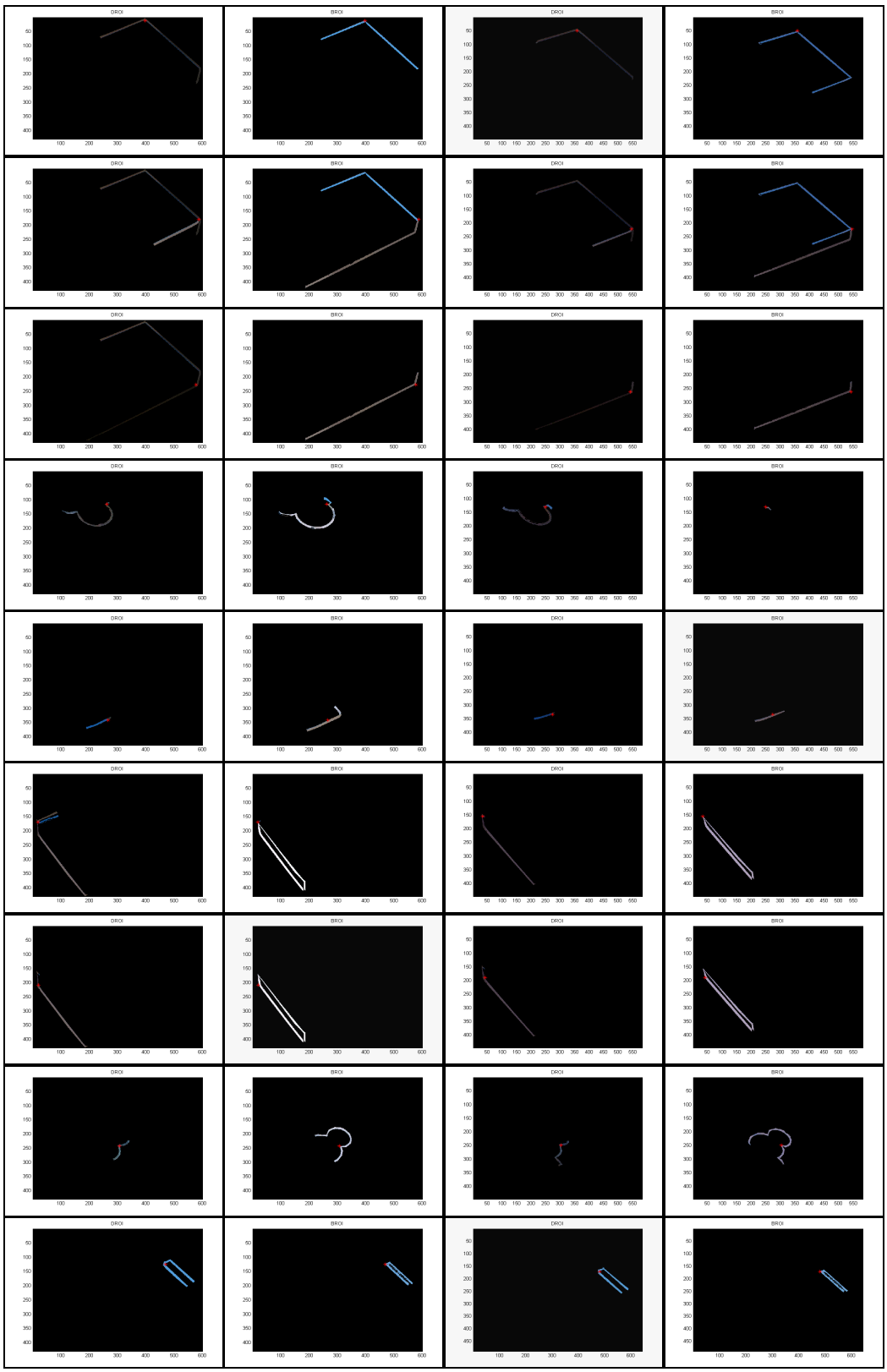


Figure 3.22. Extraction of regions around points of interest of a wide-baseline objects scene. The original images and points of interest were presented in Figure 3.21. Odd columns are *droi's* whereas even columns are *broi's*.

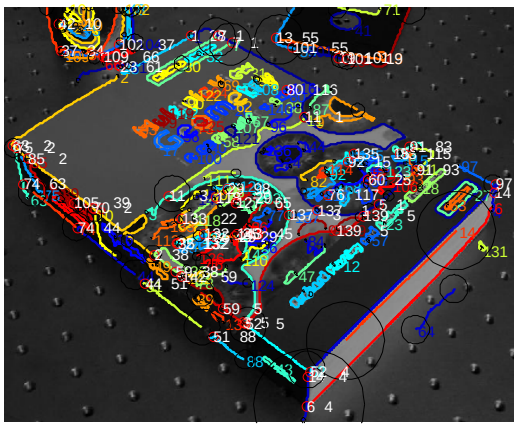


Figure 3.23. Contour map with windows where to search for intersections between two contours. Intersections found are numbered in white in the image by the number of two contours that intersect.

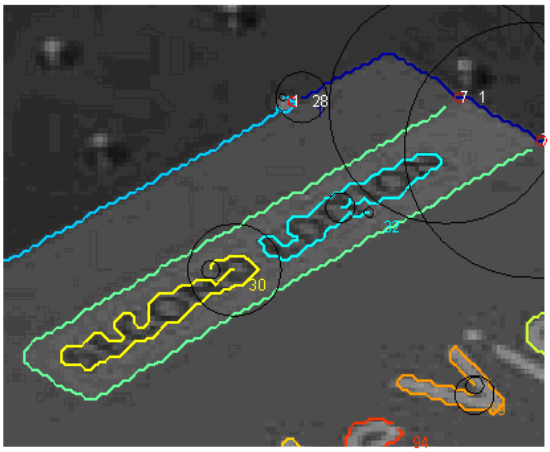


Figure 3.24. Search for intersections. Close-up of the intersection map. Intersection pair *30-32* is removed after as there is a restriction that a contour cannot intersect another contour if previously it intersects itself.

Rather than only finding intersections by propagating end-segments we could have also considered intersections at the connection of contours segments with ad-hoc constraints based on the angle formed by the junction and normalized lengths of the segments involved. However, changes of views will degenerate these as points of interest since these intersections are only invariant up to rigid transformations. The number of points of interest would also increase severely losing therefore the discriminative power

presumed to the definition attached to a point of interest. So this alternative was not considered.

Corners. Corner detectors show many responses in highly textured images due to the rapid local intensity variation they are defined from. Therefore corners lose their ability to discriminate as we can see in figure *3.26*. We discard the common association *corner = Harris corner* and consider a "corner" or 'point of interest' as a point of high curvature on a single open or closed contour. Scenes with man-made objects contain structural elements that can be described by contours and corners lying over them (figure *3.27*).

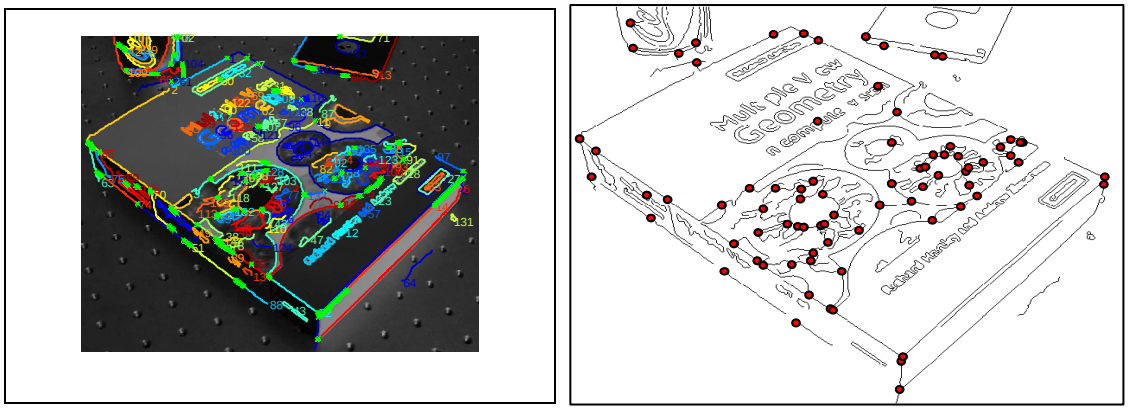


Figure 3.25. Contour propagation to search for intersections. a) The green crosses indicate the propagation of each contour endpoint to search for neighbour contours; b) intersection map.

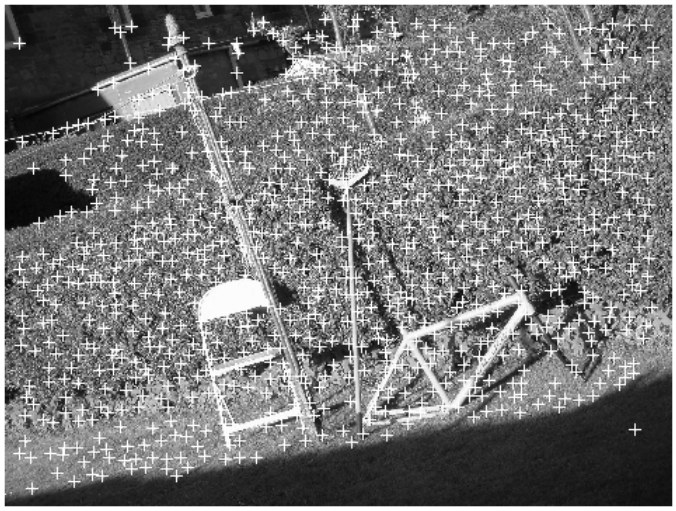


Figure 3.26. Harris-Stephens-Noble corner over a highly textured image. Smoothing Gaussian of 1.5 pixels width.

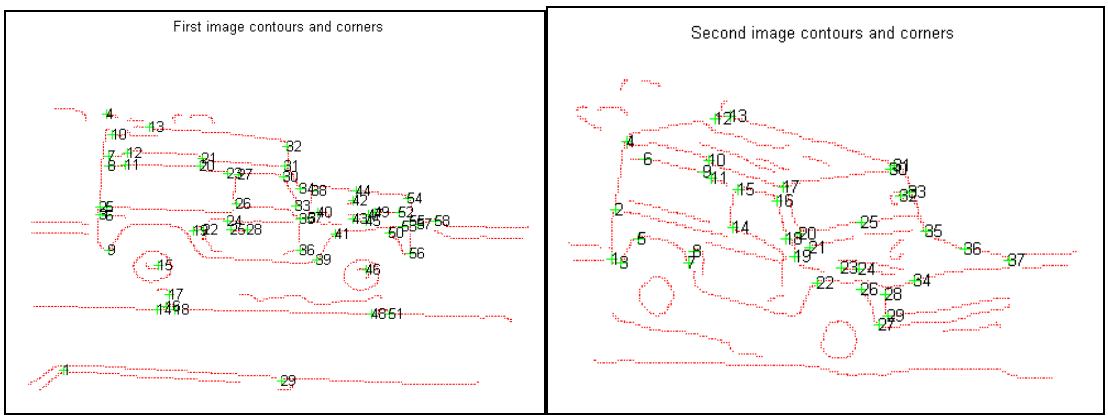


Figure 3.27. Corner points detected by the Harris-Stephens-Noble operator that lie on extended contours.

3.5 Graphs

We are proposing a combinatorial extraction of the information along the contours connecting every pair of points of interest in the form of a graph, which will indeed increment considerably the processing time, but will especially strengthen the reliability of the primitive features for our scenario. The search space is reduced by including ancillary heuristic constraints; otherwise the combinatorics could become unwieldy.

We introduce some basic definitions in graph theory [24]. A graph $G(V,A)$ is a pair of sets V and A where the elements of the set V are called *vertices* or *nodes* and the elements of A are called *arcs*. The nodes contain information about the structures and the arcs the relationships between the structures. If there exist the connections $\alpha=(v,w)$ and $\beta=(w,v)$ and $\alpha=\beta \rightarrow (u,w)=(w,v)$ the arcs are considered in both directions and the graph is called a *non-directed graph*. A node w is adjacent to another node v if and only if there exists an arc that links both nodes. A path in a graph is a sequence of nodes $p=\{v_1, v_2, ..., v_n\} \mid (v_i, v_{i+1}) \in A \ \forall i \in [1,n[$, which length is the number of arcs that the path contains or the number of vertices minus one. A path is *simple* when all its vertices are different, or at the most, only the first and last are the same. A non-directed graph is *connected* when there is a path connecting any pair of nodes of the graph, *i.e.* all the nodes are connected.

Our basic features, contours and points of interest, can be organised in the form of a graph. The search of paths between contour-connected points of interest can provide a better performance against noise and viewpoint variability. Points of interest prove that

can be a quite reliable support in the wide-base case whereas contours are exposed to partial extractions, occlusions and different labelling at junctions.

In our system, the arcs will represent contours which overlap or lie within a certain proximity to a Harris corner, and the nodes will be virtual representations of Harris corners over or in the proximity of contours, intersections as defined in section *3.4* and the endpoints of the contours represented by arcs. We differentiate between *processing* or *active* nodes (Harris corners) and *auxiliary* nodes (intersections and endpoints). The former gives rise to *processing* arcs, which are the paths from where to extract the information that will define descriptors, whereas auxiliary nodes play the role of connectivity. That choice is consistent with the fact that we consider high-curvature points more reliable than contour endpoints or projective intersections between contours.

Contours and points of interest (corners and intersections) have been extracted, and data structures contain spatial information about the contours in the proximity of each point of interest and about the closest sample in the contour to that point of interest. The information is reorganised so as to have for each contour the points of interest associated with them, that way contours with no points of interest as well as corners without contours within its vicinity are discarded. The nodes are expanded by searching for its connections, *i.e.* the equivalent of the parent and the successors in a tree structure. We consider connectivities of a node with: *a)* next and previous nodes along the contour and *b)* other nodes in other contours associated to the same point of interest. After expansion the nodes are visited using a *Depth First Search* (*DFS*) algorithm. The *DFS* algorithm returns the sequence of visit of nodes within each connected graph, providing paths between any two nodes of the graph. The shortest path is the one with minimum distance in number of nodes and where loops within the path are sieved. The process can result in a single or multiply connected graphs depending on whether all points of interest are interconnected or not.

Let us carry out a simple example to illustrate the idea. Figure *3.28* is our input image and figure *3.29* represents the corresponding graph. Four different contours, represented in different colours, have been extracted. Also, grey, orange and white-filled stars represent allegedly extracted (manually defined) Harris corners, intersections and endpoints respectively. The yellow boxes at the right hand side of the stars symbolize

the nodes associated with that point of interest. To avoid confusion nodes are listed by letters in the colour of the contour they belong to, whereas interest points by numbers. In that figure, there are a couple of particular cases. First let us examine the point of interest number *9*, a Harris corner.

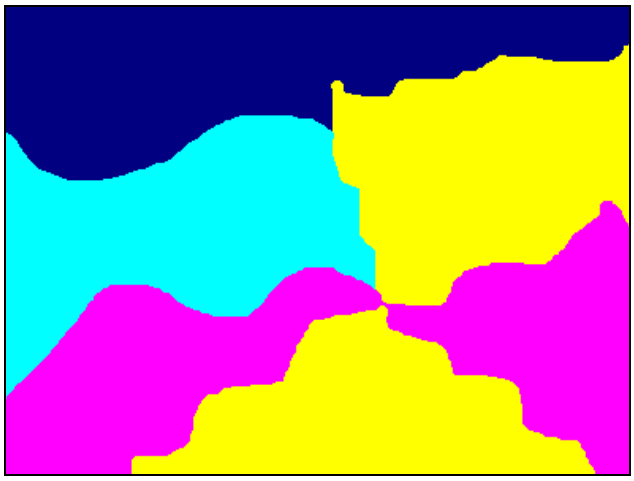


Figure 3.28. Sample image.

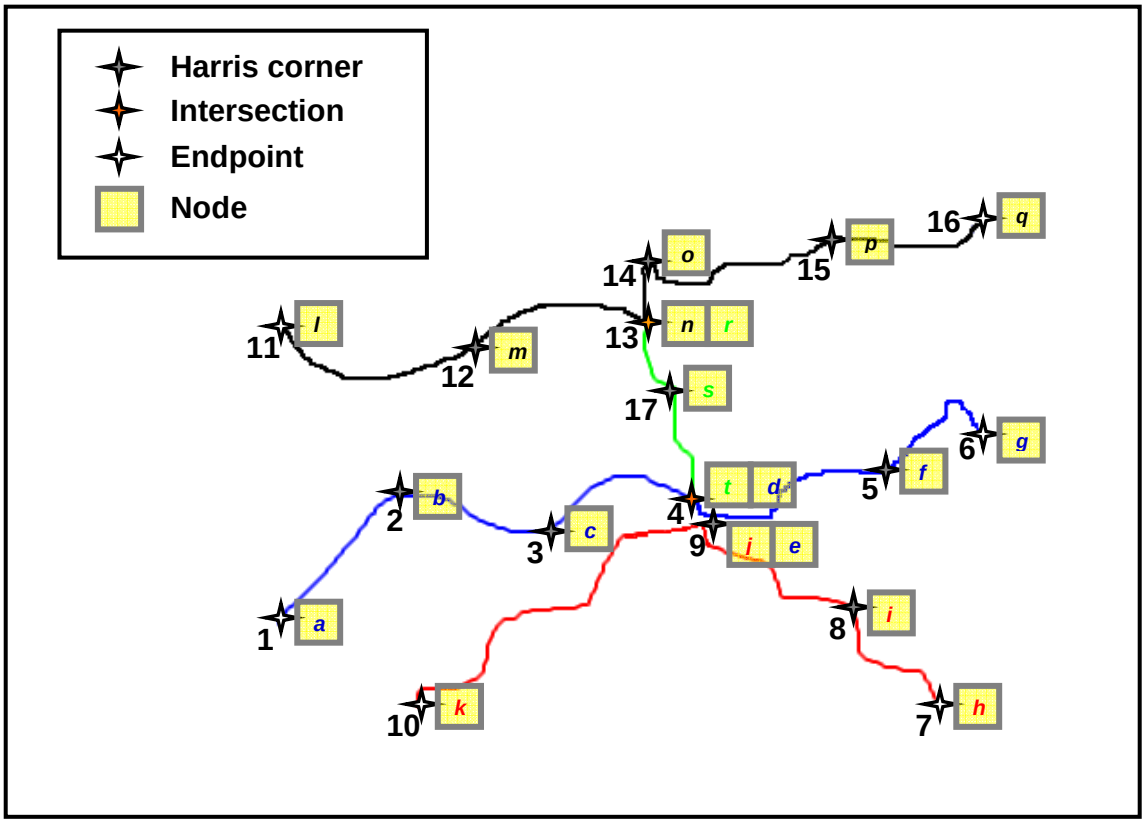


Figure 3.29. Graph.

This point of interest does not lie over a contour but it has red and blue contours within its vicinity. Nodes *j* and *e* are thus created. Node *j* will contain the closest sample to the *poi* in the red contour and node *e* will contain the closest counterpart in the blue contour. Therefore node *j* is connected to *k* and *i* and node *e* to *d* and *f*. Since nodes *j* and *e* are associated to the same *poi*, both are also connected and therefore permit a virtual path between the red and blue contours. A second case is the green contour. It is not connected to any other contour. However, by projecting its endpoints within a predefined distance (projection of end segment of a contour for intersection of contours) it gets in touch with the black and blue contours generating nodes *n* and *r* and *t* and *d*, respectively. As these are intersections, they will not be active nodes but will work only as connections between contours. Starting from node *a*, the *DFS* algorithm gives the following order of visits *D = {a, b, c, d, e, f, g, f, e, j, i, h, i, j, k, j, e, d, t, s, r, n, o, p, q, p, o, n, m, l, m, n, r, s, t, d, c, b, a}*. Sequence *D* gives all the possible paths between all these connected nodes. As an example, the two active nodes *c* and *i* can be linked by the path *p1={i, j, k, j, e, d, t, s, r, n, o, p, q, p, o, n, m, l, m, n, r, s, t, d, c}*, but the shorter path *p2={ c, d, e, f, g, f, e, j, i}* is the one naturally preferred. Still, notice that this path is not simple and the shortest path is the one resulting after removing the loop that exists inside the *p2*: *p3={c, d, e, j, i}*. The procedure that builds the graph and processes the information between points of interest to define descriptors is shown in pseudo-code in the next page.

A demonstration of the method over real stereo images is shown in figures *3.30* to *3.34*. This image dataset is comprised of indoor and outdoor scenes. Images in figures *3.33* and *3.34* are a reference for many authors [83]. Contours and points of interest are detected as explained in previous sections. The information is organised in the graphs displayed in the figures, where yellow nodes are active nodes (*POIs*) and white nodes are auxiliary nodes (intersections and contour endpoints) that can interconnect nodes in different contours. The spatial coordinates of the path in between active nodes defines the ground information to build an invariant descriptor in the next chapter. Therefore, there exist as many descriptors as combinatorics among active nodes.

Input:

- Landmarks:
  - Corners over/nearby contours (*POIs* – active nodes)
  - Intersections (auxiliary nodes)
- Contour spatial information (arcs)

Output:

- Descriptors for pairs of *POIs*

Process:

FOR every contour
    Find the landmarks in their proximity
    IF no landmark
        Continue
    END
    FOR every landmark
        Find the closest sample to that landmark in the contour
        IF *POI*
            Node is active
        ELSE
            Node is auxiliary
        END
        Store landmark as a node, function (active/auxiliary), closest contour sample and contour number
    END
    Store endpoints of the contour as auxiliary nodes
    Store landmarks, function (active/auxiliary), closest sample and contour number
END

%Expansion of nodes
FOR each node *u*
    Search for consecutive and previous nodes in the same contour

    %Search for neighbouring nodes in other contours around the landmark
    IF the landmark has neighbour contours
        FOR every neighbour contour
            Find sample in the other contour associated to the landmark

Identify to which node it corresponds and store the node as a connection

END

END

END

%Build graph

FOR each node *u*

IF already visited

Continue

END

Do DFS visit and store graph

Search for redundant nodes that correspond to the same landmark

Delete these nodes from the list of nodes to process in the graph

Delete nodes that are intersections from the list of nodes to process but maintain them in the list of connections between nodes

END

%Compute descriptor

FOR each graph

Compute all combinations between active nodes taken two at a time

FOR each combination

Find shortest path

- Find minimum distance (in number of nodes)

  e.g.: *c d e f g f e j i h*

- Sieve loops in the path

  e.g.: *c d ~~e f g f~~ e j i h*

Do not consider intermediate contours along the same contour

Extract spatial information along the path (to compute invariant descriptor in the next chapter)

Save descriptor

END

END

RETURN

Figure 3.30. Book stereo scene. a-b) Contours and corners; c-d) Output graph.

Figure 3.31. Antenna stereo scene. a-b) Contours and corners; c-d) Output graph.

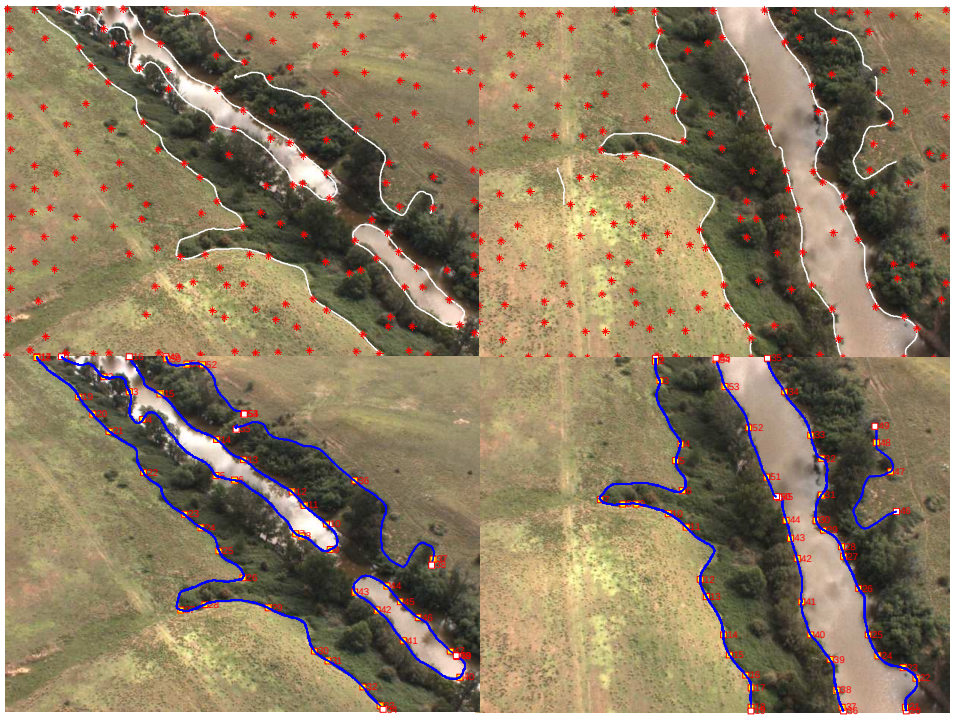


Figure 3.32. Countryside stereo scene. a-b) Contours and corners; c-d) Output graph

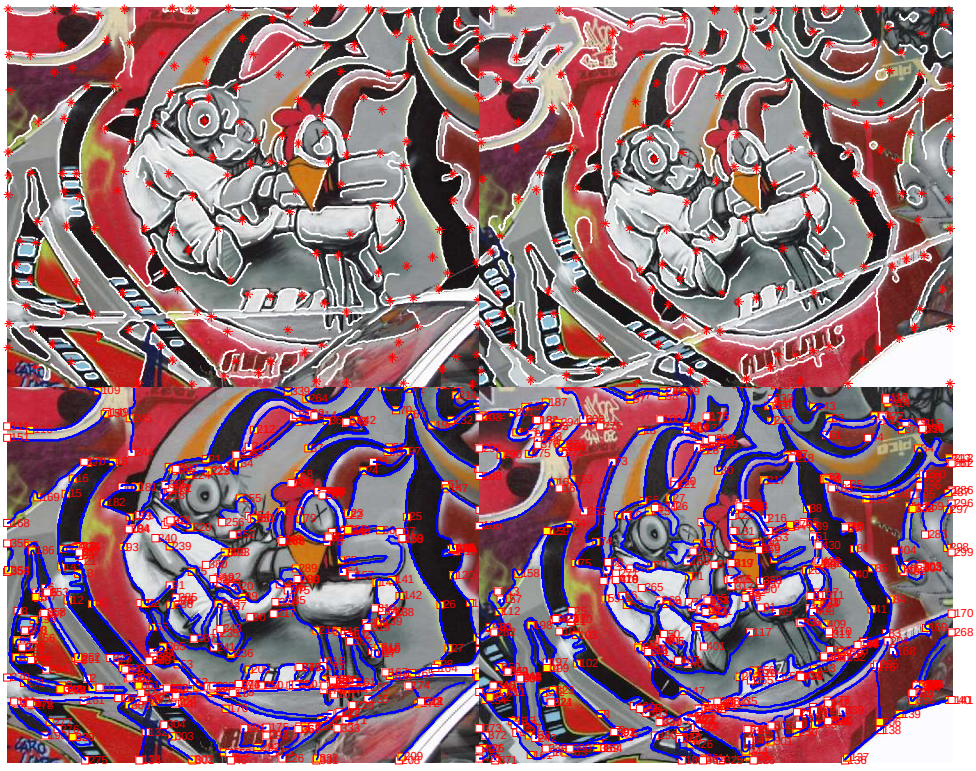


Figure 3.33. Graffiti stereo scene. a-b) Contours and corners; c-d) Output graph.

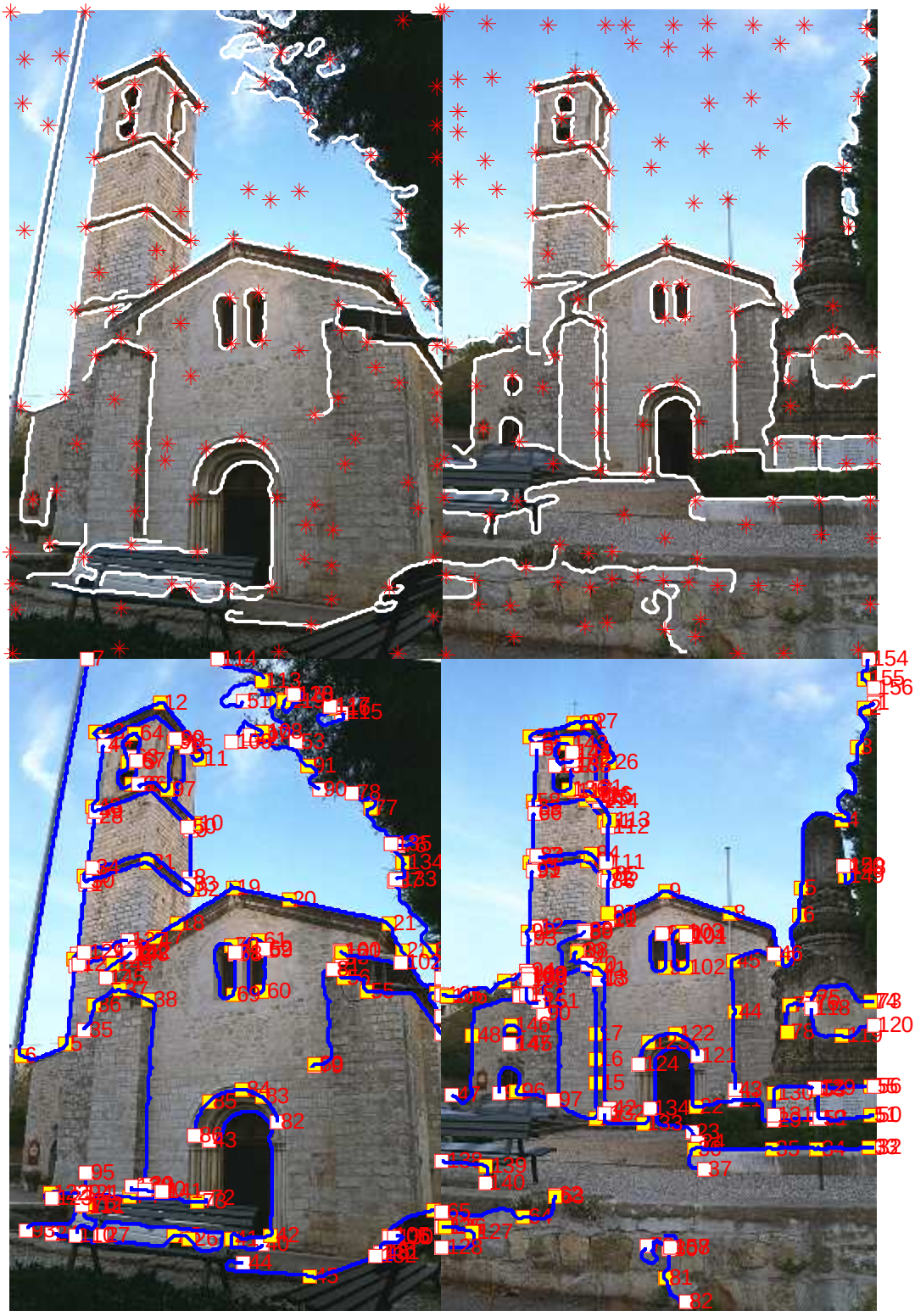


Figure 3.34. Valbonne stereo scene. a-b) Contours and corners; c-d) Output graph.

3.6 Summary

We have presented different methods for the extraction of morphological and photometric information from images. The extraction of features is a preliminary step of principal importance for the success of further scene analysis. The accuracy views during the extraction, the amount of features and their consistency across views will define the complexity and the feasibility of the method.

Extended contours were found in the images by using maps of magnitude and direction gradients. The resultant edges were extended to contours by assembling edges within a neighbourhood given proximity, continuation and similarity. We have also presented a contour segmentation technique functional for finding projective intersection between contours and for articulating flank regions at both sides of the contours where to analyse the photometry. These regions have a high dependence on the extraction and intersections of contours and on the photometric nature of the image to define homogeneous photometric regions at both sides of the contours. Two alternatives were anticipated: regions along whole contours and regions emanating from points of interest and delimited by contour intersections. Any of the proposed methods proved reliable enough for the wide-baseline case. A better and also more expensive choice was the rearrangement of points of interest and contours in the form of a graph. The combinatorics of all the paths delimited by points of interest interconnected by contours are preferred as signatures and regions to extract photometric information alongside. Since corners prove good repeatability and good behaviour under viewpoint and photometric changes proclaim the employ of a graph articulated by corners and intersections as an interesting alternative to tackle the unreliability in the extraction of contours given noise, breaks and the tracing of other contours at intersections.

Chapter 4 – The affine invariant descriptor

4.1 Introduction

The variability of the objects viewed under different viewpoints and illumination conditions can be solved in three ways: a) by searching from an a-priori camera model the whole space of transformations and align the transformed and reference image; b) by using image normalization of scale, rotation, contrast, etc. or c) by constructing invariant functions. The first approach is obviously not viable due to computational burden. The second alternative is sometimes included as a pre-processing stage inside a more-efficient process when there is information from a model that permits normalization. However, invariance is a better solution that is achievable for planar objects and there exists a large literature.

Methods that use invariant descriptors characterise features which do not change under a given photometric or geometric image deformation with the purpose of finding counterpart landmarks (points of interest, regions…) in both images to solve the correspondence problem. Simple examples of geometric invariance can be a segment line, which length does not change under a translation or rotation in the plane but it does under other *2D* transformations; or a circle, that under an affine transformation will be distorted in to an ellipse. In the photometric case, the transformation will rely on extrinsic and intrinsic parameters of the cameras and the lighting conditions. Therefore, it is fundamental to know the kind of transformation that the images will undergo and the set of features to work with in order to find descriptors invariant to this geometric transformation, which is usually affine or projective.

Projective invariants from points and lines have been developed from the theory of geometric algebra [8]: the *1D* cross-ratio as the basic projective invariant (from four points in a line) and its bi- and tri-dimensional generalisations (five points in a plane and six points in *3D* space, respectively); as well as *3D* invariants for the stereo case (six non-coplanar corresponding points, given the fundamental matrix *F*) and for the three-view scenario (components of the trilinear tensor from lines and planes).

Shape descriptors such as Fourier descriptors and Elliptic Fourier descriptors have been widely used in the literature, but are generally restricted to close contours. Other popular feature descriptors over two dimensional functions are Fourier Mellin descriptors [1,23], Zernike moments [106,60] and pseudo-Zernike moments [14]. However, the pioneering work on moment invariants (absolute orthogonal moment invariants) by Hu [53] has been used extensively over throughout the years. These moments based on algebraic invariants were invariant to similarity transformations. The concerned reader can find individual modifications and improvements of Hu's moments in [77,6,67,90]. Also Flusser and Suk [37] presented complex moment invariants to affine transformations and, lately, Flusser and Zitová [39] combined and expanded the invariance to contrast and to convolution with a centrally symmetric point-spread function (blur effect). Mindru *et al.* [82] presented generalised colour moments, descriptors that compute affine invariant moments on shape and colour bands.

Excluding the last method, most of the bibliography aforementioned is related to intensity images. Doubtless the use of colour [95] can contribute with further information but at the same time colour is very sensible to the scene illuminant. Therefore, raw colour features are not reliable *per se* in image recognition. This dependency on the illumination should be removed and some other stronger to illumination models such as CIE LUV can be preferred rather than the traditional RGB model. Although out of our scope, other colour representations are based on histogramming. Nevertheless, this option has the drawback of losing the spatial information of the patterns.

4.2 Affine geometric invariance

4.2.1 The affine frame

An area-preserving affine transformation $\vec{y} = A\vec{x} + \vec{b}$, is characterized by a translation vector $\vec{b}$ and a matrix *A* being *SL(2,R), i.e.* the group of all real *2x2* matrices with determinant one that preserve oriented area [44,85]. Starting from a *Frenet* frame where the area enclosed by two vectors $\{e_1 \;\; e_2\}$ is the unit area, we search for an oblique system of coordinates defined by two vectors $\{a_1 \;\; a_2\}$. These vectors delimit a parallelogram of unit area, thus having an area preserving frame under affinities. This frame can be defined over every point of a curve $\Gamma(t)=(x(t),y(t))^T \in R^2$, which is at least

a two times differentiable planar curve. The vector a_1 can be the tangent vector at a given point of the curve, whereas a_2 should be defined so as to enclose an oblique frame of unit area. Therefore, the determinant $\|a_1(t), a_2(t)\|$ should be one. The setting is:

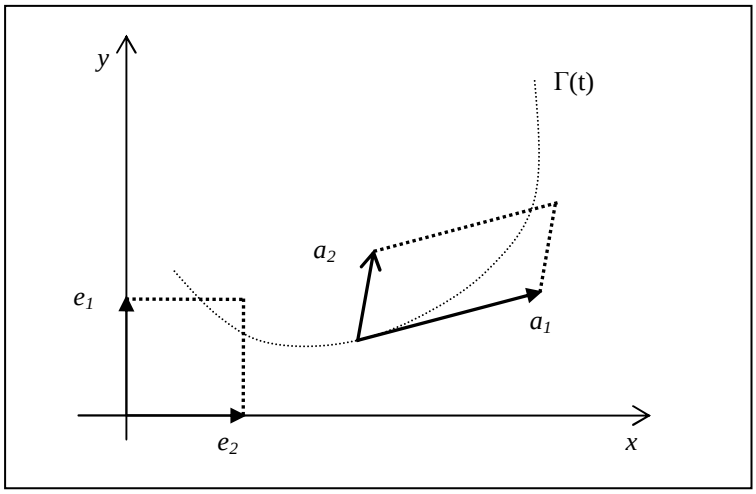


Figure 4.1. Euclidean and affine frames (from [44])

If $\dot{\Gamma}(t)$ and $\ddot{\Gamma}(t)$ are respectively the first and second derivatives of the curve Γ at the parameter t, these vectors $\{a_1\ a_2\}$ that determine a unit area are given by:

$$\begin{aligned} a_1(t) &= \left\|\dot{\Gamma}(t), \ddot{\Gamma}(t)\right\|^{-\frac{1}{2}} \dot{\Gamma}(t) \\ a_2(t) &= \left\|\dot{\Gamma}(t), \ddot{\Gamma}(t)\right\|^{-\frac{1}{2}} \ddot{\Gamma}(t) \end{aligned} \tag{4.1}$$

4.2.2 The affine arc-length metric

The basic concepts on affine differential geometry introduced above lead us to the definition of the affine arc-length expression.

As we need the parallelogram created by the oblique frame $\{a_1\ a_2\}$ to be of unit area, the curve Γ is reparameterised to a new parameter σ – always assuming the condition $\left\|\dot{\Gamma}(t), \ddot{\Gamma}(t)\right\| \neq 0$.

$$\left\|\dot{\Gamma}(x(t),\ y(t)),\ \ddot{\Gamma}(x(t),\ y(t))\right\| = \frac{dx}{dt}\frac{d^2y}{dt^2} - \frac{d^2x}{dt^2}\frac{dy}{dt}$$
$$= \frac{dx}{d\sigma}\frac{d\sigma}{dt}\cdot\frac{d^2y}{d\sigma^2}\left(\frac{d\sigma}{dt}\right)^2 - \frac{dy}{d\sigma}\frac{d\sigma}{dt}\cdot\frac{d^2x}{d\sigma^2}\left(\frac{d\sigma}{dt}\right)^2 \tag{4.2}$$
$$= \underbrace{\left\|\frac{d\Gamma}{d\sigma}, \frac{d^2\Gamma}{d\sigma^2}\right\|}_{1}\cdot\left(\frac{d\sigma}{dt}\right)^3 = \left(\frac{d\sigma}{dt}\right)^3$$

Therefore the expression of the arc-length parameterisation σ is as follows:

$$\sigma(t) = \int_{0_0}^{b}\sqrt[3]{\left\|\Gamma'(x(t),y(t)),\Gamma''(x(t),y(t))\right\|}dx(t)dy(t) \tag{4.3}$$

and the normalized version:

$$\sigma_N(t) = \frac{\sigma(t)}{\max(\sigma)} \tag{4.4}$$

which is an absolute invariant. However it needs both endpoints of the curve to be known.

By performing several affine transformations to the original contour we can compute a parameter analysis of the affine invariant metric as shown in the next figure. However, although the normalized affine arc-length is an absolute invariant, it cannot cope with partial contour matching, *i.e.* the contours should correspond exactly to each other, unless a partial (and exact) segmentation of corresponding parts of the contours is known.

In order to evaluate the performance of the affine arc-length the next sequences (figure 4.2) show a real image and its affinely transformed counterpart. The contour maps are extracted in both images and a few contours highlighted as examples. Four synthetic curves have also been superimposed on the images to add to the test: a circle, a parabola, a ellipse and a *sine-exponential* function described by *z(x(t),y(t))* with *x(t)=a*sin(t)* and *y(t)=b*exp(t)*. Figure 4.3 presents the affine arc-length *σ(t)* of these curves, where *t* is the centripetal distance along the curve. The distance along the curve is normalized to *1*. At a first glance, the affine arc-length could be used to distinguish between some corresponding curves. However, despite the reduced set of curves of this example we see that some of them have similar behaviour.

We will see in the next sub-section that there exists a linear relation between the affine arc-length of two affinely transformed curves. This linear relation, which gives an estimation of the transformation undertaken, is not evident in figure 4.3. That is due to the fact that the *x*-coordinate of the plot represents the centripetal distance along the curve. This metric is not invariant under affine transformations, thus the property of geometric invariance up to a linear relation of the affine arc-length is not evident in this representation. Figure 4.4 represents the ratio of corresponding affine arc-length curves to the third power, which is a measure of the transformation between the curves. Notice that the ratios for the ground truth (synthetic curves) overlap each other giving a single measure of the transformation between all them. For the real contours, which are extracted independently, the results are less satisfactory since there is no sample-to-sample contour correspondence between views.

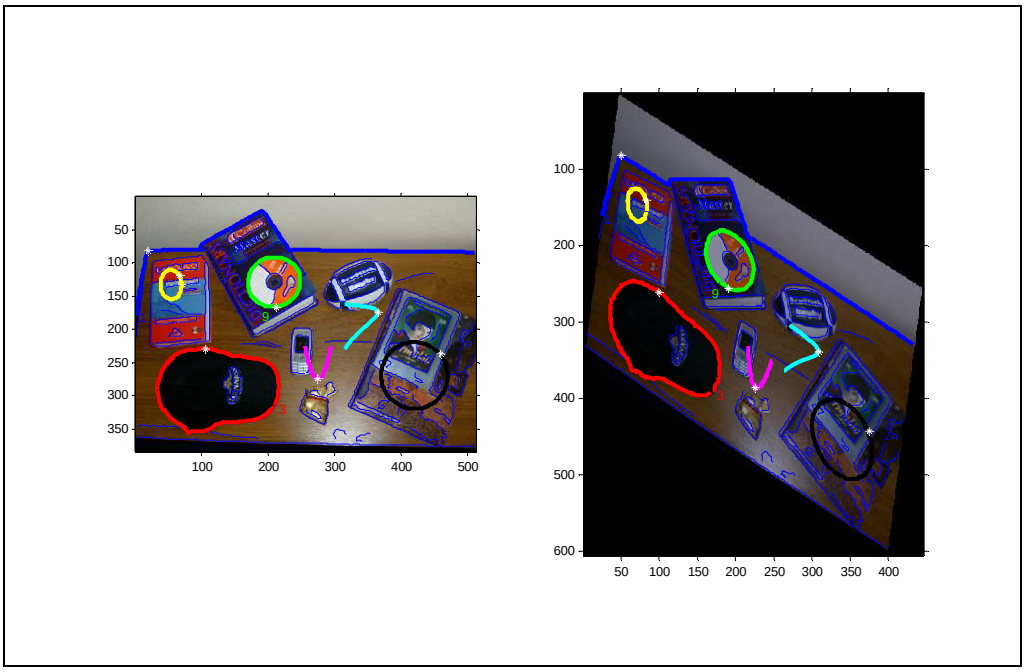


Figure 4.2. Original and affinely transformed images with contour map. Highlighted, selection of real contours and synthetic curves under study.

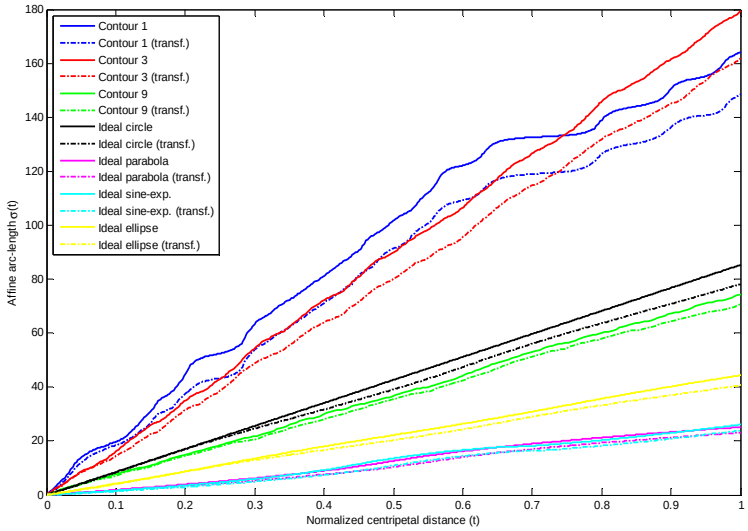


Figure 4.3. Affine arc-length

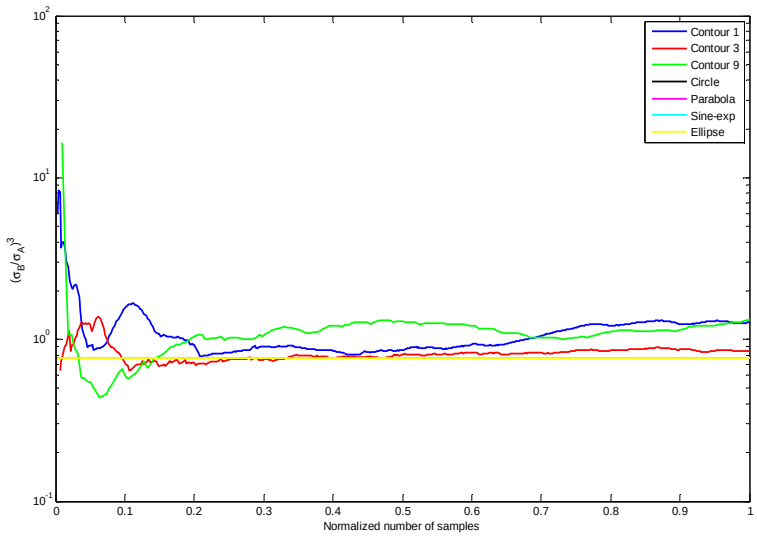


Figure 4.4. Ratios of affine arc-length.

Figure 4.5a presents an instance of a synthetic image where a contour and a point of interest has been extracted. Figure 4.6b shows an instance of the same image transformed by an affinity. The other figures show the affine arc length and normalised affine arc length under a wide range of transformations. Notice how the affine arc length is invariant up to scale, whereas the normalised affine arc length is an absolute invariant for the whole range of affine transformations.

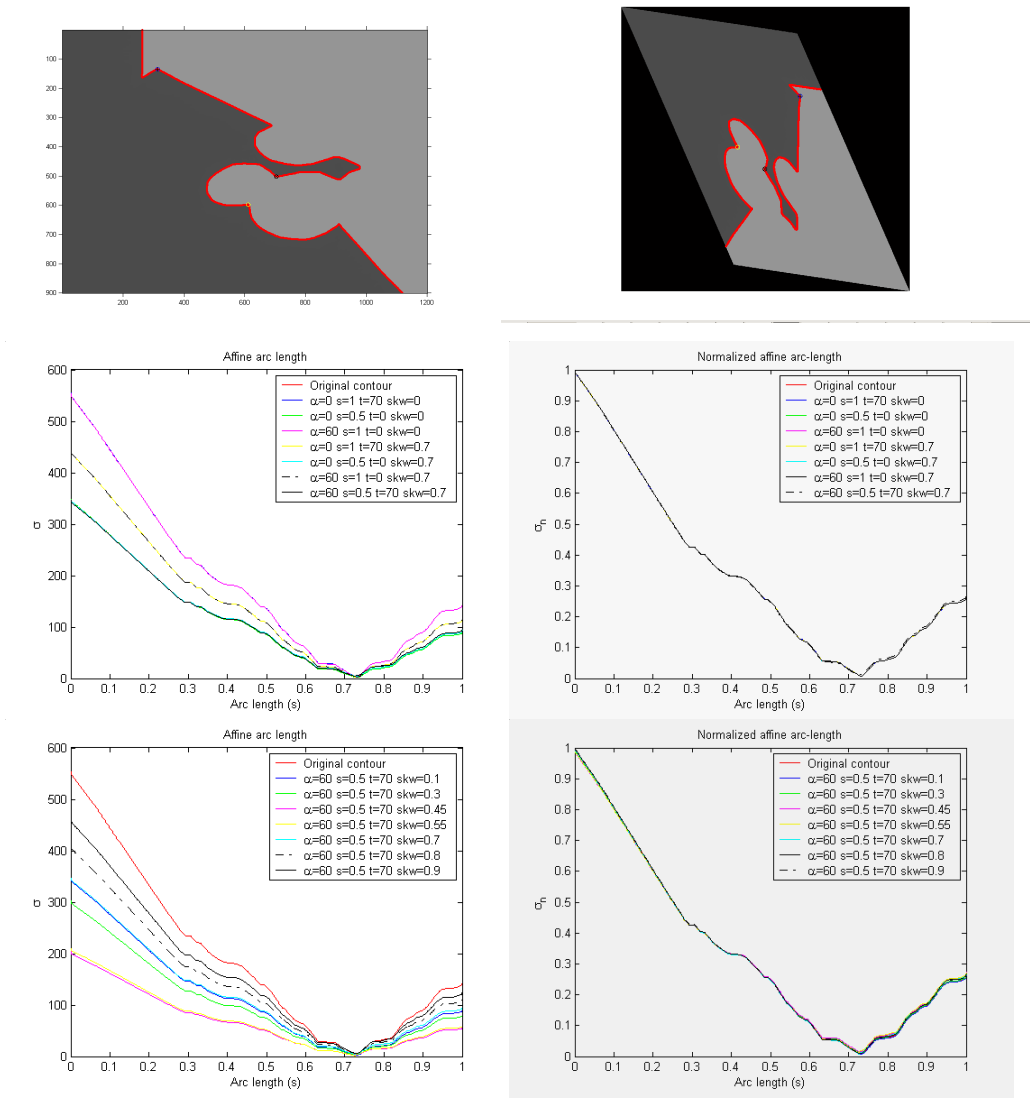


Figure 4.5. Affine and normalized arc-length parameter analysis. a) and b) Input contour and affine-transformed contour, point of interest marked with a black circle and endpoints of the contour in blue and yellow circles c) and e) affine arc-length for the original and transformed contours. d) and f) normalized affine arc-length for the original and transformed contour.

4.2.3 The affine invariant area

We assume again that we have a curve $\Gamma_A(t)$ that is transformed to a curve $\Gamma_B(t)$ by an affine transformation M. Then $\Gamma_A(t)$ and $\Gamma_B(t)$ are reparameterised as $\Gamma_A(\sigma_A(t))$ and $\Gamma_B(\sigma_B(t))$, respectively. Recall that the parameter σ defines an oblique frame of unit area at every point of the curve. We pursue that these parallelogram instead of covering a unit area in the second image it should enclose the area that corresponds to the unit

frame in the first image. Evidently, the effect of the affine transformation *M* is reflected in the transformed image by a scaling of the corresponding area [49]:

$$\frac{\text{area after transformation}}{\text{area before transformation}} = \|\mathrm{M}\| \tag{4.5}$$

with *M* a *3x3* matrix:

$$M = \begin{bmatrix} m_{11} & m_{12} & t_x \\ m_{21} & m_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix} \tag{4.6}$$

where the determinant is $m_{11}m_{22}$-$m_{12}m_{21}$, as m_{11} and m_{22} correspond to scaling in *xy* coordinates and m_{12} and m_{21} to shear. Consequently, scaling does shape the new area of the parallelogram whereas shearing can only affect it.

From equations (4.1) and (4.5), two corresponding areas can be extracted by scaling the parallelogram defined by the vector a_{1B} and a_{2B} to $\tilde{a}_{1B}$ and $\tilde{a}_{2B}$:

$$\|\tilde{a}_{1B}, \tilde{a}_{2B}\| = \|a_{1B}, a_{2B}\| \cdot \|M\| \tag{4.7}$$

The relation between the two affine arc length metrics in both corresponding curves Γ_A and Γ_B is as follows:

$$\frac{\sigma_B(t)}{\sigma_A(t)} = \frac{\int_{t_0}^{t} \sqrt[3]{\|a_{1B}, a_{2B}\| \cdot \|M\|}\, dx(t)dy(t)}{\int_{t_0}^{t} \sqrt[3]{\|a_{1B}, a_{2B}\|}\, dx(t)dy(t)} = \sqrt[3]{\|M\|} \tag{4.8}$$

Therefore we show that there is a linear relationship between the affine arc-length of two corresponding curves. So far, by computing the affine arc length of a curve and its transformed version we can estimate the transformation undergone *M*. However, we approach that fact the other way around: instead of extracting the transformation between the two curves, we can scale the vectors $\{a_1 \;\; a_2\}$ in the second image by the relation given in equation (4.8) and extract corresponding patches in both images.

Consequently, from the two equations above:

$$\left\|\tilde{a}_{1B}(t), \tilde{a}_{2B}(t)\right\| = \left\|a_{1B}(t), a_{2B}(t)\right\| \cdot \left(\frac{\sigma_B(t)}{\sigma_A(t)}\right)^3 \tag{4.9}$$

Example. The following example illustrates the idea. For easiness we choose a circle. A curve Γ_A that describes a circle is expressed in parametric form:

$$\begin{aligned} x &= r \cdot \cos(\theta) \\ y &= r \cdot \sin(\theta) \end{aligned} \tag{4.10}$$

where:

$$\theta = \frac{l}{r} \tag{4.11}$$

being l the length of the circle and r its radius.

From equation (4.1):

$$\begin{aligned} a_1 &= x'(\theta) = \begin{bmatrix} -\sin(\theta) \\ \cos(\theta) \end{bmatrix} \\ a_2 &= x''(\theta) = \begin{bmatrix} -\cos(\theta) \\ -\sin(\theta) \end{bmatrix} \end{aligned} \tag{4.12}$$

and the area defined by these two vectors is one:

$$\left\|a_1, a_2\right\| = \sin^2(\theta) + \cos^2(\theta) = 1 \tag{4.13}$$

If we apply now an affine transformation M to Γ_A, we have the ellipse Γ_B

$$\begin{aligned} x &= a \cdot \cos(\theta) \\ y &= b \cdot \sin(\theta) \end{aligned} \tag{4.14}$$

where a and b are the semi-major and semi–minor axis, respectively. The determinant of the area defined by a_1 and a_2 over Γ_B is also *1*.

As the ellipse is obtained by applying an affine transformation M with expression:

$$M = \begin{bmatrix} k_x & s_x & t_x \\ s_y & k_y & t_y \\ 0 & 0 & 1 \end{bmatrix} \tag{4.15}$$

with k_x, k_y representing the scale in xy coordinates; s_x, s_y the shear and t_x, t_y the translation, although not relevant. The new area in the second image is given by:

$$\left\| \tilde{a}_{1B}, \tilde{a}_{2B} \right\| = \left\| \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \cdot \begin{bmatrix} k_x & s_x \\ s_y & k_y \end{bmatrix} \right\| = k_x k_y - s_x s_y \tag{4.16}$$

However, the parameters of the transformation (k_x, k_y, s_x and s_y) are unknown. The computation of the affine arc-length over the circle and the (transformed) ellipse gives an estimation of the transformation M which scales a_{1B} a_{2B} according to equation *4.8*.

Figure 4.6 shows the example of the input circle and transformed ellipse. The top plots show the extraction of a unit area from a given point t of the circle and the corresponding one in the ellipse. The central plots are the affine arc length along both curves (see the linearity between them) and the normalized version, which is an absolute invariant. The bottom plots are the extraction of the unit area in the input curve and the affine-arc-length scaled extraction of equivalent region of ratio M.

Figure 4.7 illustrates better the same example by overlapping circles of different radius over a background, input image. Both the background image and the circles are affinely transformed. The affine arc length is computed over both curves and the invariant area defined by vectors a_1 and a_2 and the counterpart given by equation (4.8) are shown for the first sample of the contour. In figure 4.7a) vectors a_1 and a_2 are coloured in red and green, respectively. Figure 4.7b) shows the effect of computing the affine invariant vectors over every sample of the circle. The tips of vectors a_1 and a_2 are linked resulting in affine invariant regions at both sides of the contour.

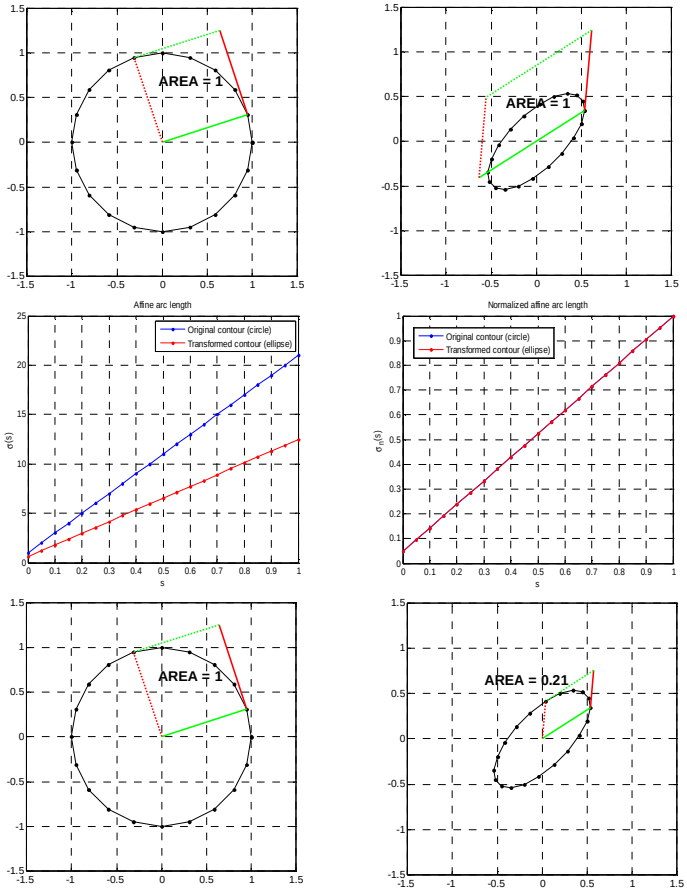


Figure 4.6. Affine-arclength-based method to extract corresponding areas. a) original curve (circle) and extraction of unit area by vectors a_1 and a_2 ; b) affinely transformed circle (ellipse) and extraction of unit area by vectors a_1 and a_2; c) affine arclength of the circle and ellipse; d) normalized affine arclength of the circle and ellipse e) same as a) and f) corresponding area defined by $\left\|\tilde{a}_{1B}, \tilde{a}_{2B}\right\|$.

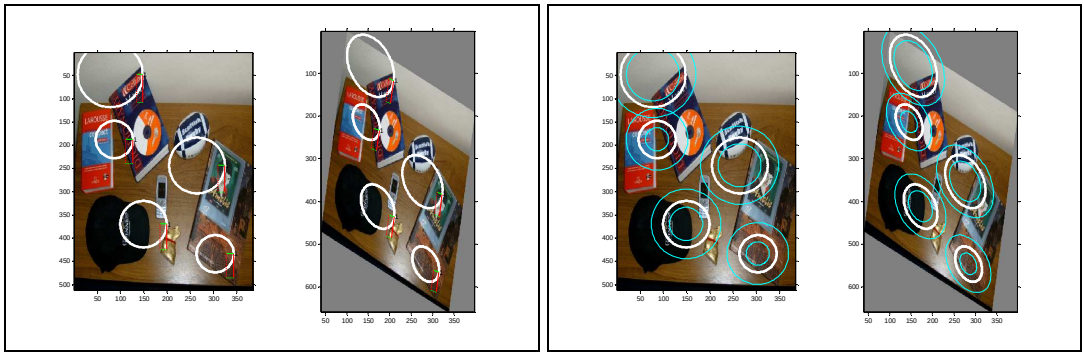


Figure 4.7. Affine invariant regions over two affinely transformed background images. a) Affine arc-length vectors enclosing corresponding areas and b) affine invariant regions by linking tips of invariant vectors.

4.3 Affine photometric invariance

4.3.1 Hu's moment invariants

Hu [53] presented a set of moments invariant to rotation, translation and changes in scale for planar geometry based in algebraic invariants. The ordinary moments of order $p+q$ of a continuous function *f(x,y)* are defined by:

$$m_{pq}^{(f)} = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} x^p y^q f(x,y)dxdy \qquad \forall f(x,y) \Rightarrow \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x,y)dxdy > 0 \qquad (4.9)$$

The central moments $\mu_{pq}^{(f)}$ are expressions of the ordinary moments that can deal with translation in the image:

$$\mu_{pq}^{(f)} = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} (x-\bar{x})^p (y-\bar{y})^q f(x,y)d(x-\bar{x})d(y-\bar{y}) \qquad (4.10)$$

being

$$\bar{x} = \frac{m_{10}^{(f)}}{m_{00}^{(f)}} \qquad \bar{y} = \frac{m_{01}^{(f)}}{m_{00}^{(f)}} \qquad (4.11)$$

the geometric centre of gravity of the function *f(x,y)* that define the central moments.

The normalized central moments, which can be invariant to changes in scale, are defined from the central moments:

$$\eta_{pq}^{(f)} = \frac{\mu_{pq}^{(f)}}{\mu_{00}^{\gamma}} \qquad \gamma = \frac{p+q+2}{2} \qquad (4.12)$$

By combining orders of normalized central moments Hu generated six absolute orthogonal invariants and one shear orthogonal invariant of the second and third order. We do not present the expressions of the moments but refer to [53].

4.3.2 Generalised colour moments

Mindru *et al*. [82] present a set of moments that preserves invariance up to affine geometric and photometric transformations of the image. These are the generalised colour moments, which are computed from a slight variation of the ordinary moments of Hu by incorporating the three (*RGB*) colour bands:

$$M_{pq}^{abc} = \int\int x^p y^q \left[R(x,y)\right]^a \left[G(x,y)\right]^b \left[B(x,y)\right]^c dxdy \qquad (4.13)$$

In this expression, $p+q$ denote again the order of the moment and *abc* indicates the degree of the moment, *i.e.* each of the powers applied to the colour bands individually. As a matter of robustness, moments are computed for low orders and degrees. Hence, considering moments up to the first order and second degree, the possible generalised colour moments and their descriptive features are: a) moments of order *pq* and degree *0* (*[a,b,c]*=000) represent the *pq*-shape moments, b) moments of degree *1* only consider one band and exclude the two others being the descriptor computed over intensities of the selected band, c) likewise moments of degree *2* combine two bands and reject the one left, and d) finally moments of order 0 (p=q=0) neglect pixel spatial information.

The basic invariant moments are devised as solutions of systems of partial differential equations by means of Lie group methods [115]. These cover affine geometric invariance combined with scale photometric invariance (Type *1*), and with scaling and offset photometric variations in the image (Type *2*). Types *3* and *4* are related to scaling plus offset illumination changes and affine changes, respectively, but no affine geometric distortion permitted. We will focus our attention on the first type, as scaling photometric invariance can suffice to model the intensity variations of indoor images. For the case of outdoor scenes, affine models describe better the changes of illumination. However, since the wide baseline case is strongly constrained by geometric distortion thus the scaling plus offset photometric model can only be used in detriment of affine photometric, model- based invariants.

$$S_{02} = \frac{M_{00}^{2} M_{00}^{0}}{(M_{00}^{1})^{2}} \quad D_{02} = \frac{M_{00}^{11} M_{00}^{00}}{M_{00}^{10} M_{00}^{01}}$$

$$S_{12} = \frac{M_{10}^{2} M_{01}^{0} M_{00}^{1} + M_{10}^{1} M_{01}^{2} M_{00}^{0} + M_{10}^{0} M_{01}^{1} M_{00}^{2} - M_{10}^{2} M_{01}^{1} M_{00}^{0} - M_{10}^{1} M_{01}^{0} M_{00}^{2} - M_{10}^{0} M_{01}^{2} M_{00}^{1}}{M_{00}^{2} M_{00}^{1} M_{00}^{0}}$$

$$D_{11} = \frac{M_{10}^{10} M_{01}^{01} M_{00}^{00} + M_{10}^{01} M_{01}^{00} M_{00}^{10} + M_{10}^{00} M_{01}^{10} M_{00}^{01} - M_{10}^{10} M_{01}^{00} M_{00}^{01} - M_{10}^{01} M_{01}^{10} M_{00}^{00} - M_{10}^{00} M_{01}^{01} M_{00}^{10}}{M_{00}^{10} M_{00}^{01} M_{00}^{00}}$$

$$D_{12}^{1} = \frac{M_{10}^{11} M_{01}^{00} M_{00}^{10} + M_{10}^{10} M_{01}^{11} M_{00}^{00} + M_{10}^{00} M_{01}^{10} M_{00}^{11} - M_{10}^{11} M_{01}^{10} M_{00}^{00} - M_{10}^{10} M_{01}^{00} M_{00}^{11} - M_{10}^{00} M_{01}^{11} M_{00}^{10}}{M_{00}^{11} M_{00}^{10} M_{00}^{00}}$$

$$D_{12}^{2} = \frac{M_{10}^{11} M_{01}^{00} M_{00}^{01} + M_{10}^{01} M_{01}^{11} M_{00}^{00} + M_{10}^{00} M_{01}^{01} M_{00}^{11} - M_{10}^{11} M_{01}^{01} M_{00}^{00} - M_{10}^{01} M_{01}^{00} M_{00}^{11} - M_{10}^{00} M_{01}^{11} M_{00}^{01}}{M_{00}^{11} M_{00}^{01} M_{00}^{00}}$$

$$D_{12}^{3} = \frac{M_{10}^{02} M_{01}^{00} M_{00}^{10} + M_{10}^{10} M_{01}^{02} M_{00}^{00} + M_{10}^{00} M_{01}^{10} M_{00}^{02} - M_{10}^{02} M_{01}^{10} M_{00}^{00} - M_{10}^{10} M_{01}^{00} M_{00}^{02} - M_{10}^{00} M_{01}^{02} M_{00}^{10}}{M_{00}^{02} M_{00}^{10} M_{00}^{00}}$$

$$D_{12}^{4} = \frac{M_{10}^{02} M_{01}^{01} M_{00}^{00} + M_{10}^{01} M_{01}^{00} M_{00}^{20} + M_{10}^{00} M_{01}^{20} M_{00}^{01} - M_{10}^{20} M_{01}^{00} M_{00}^{01} - M_{10}^{01} M_{01}^{20} M_{00}^{00} - M_{10}^{00} M_{01}^{01} M_{00}^{20}}{M_{00}^{20} M_{00}^{01} M_{00}^{00}}$$

(4.14)

S_{pq} invariants are related to single band analysis, while D_{pq} are for combinations of two out of three bands. The superscript indicates the power(s) to use for the band(s) under consideration. Therefore, there should be computed 6 *S*-invariants (S_{02} and S_{12} in *R*,*G* and *B*) plus *18 D*-invariants as a result of the three possible combinations *RG*, *RB* and *GB*. In total, *24* invariants can be reduced to a basis set of 21 invariants. The elements discarded are $D_{12}^{3(RB)}$, $D_{12}^{4(RG)}$ and $D_{12}^{4(GB)}$.

Figure 4.8 shows an input image and combinations of geometrical and photometrical patches – some extracted over the same area, others not. The basic set of 21 invariant moments is computed. Figure 4.9 shows the result for corresponding and non-corresponding regions.

4.4 Descriptor and matching

In Chapter 3 we have discussed the way of extracting ribbons around contours. The drawback is that we are not extracting invariant regions and the approach is ad-hoc. However, considering that the contour map is accurately detected, patches are extracted with an acceptable photometric homogeneity. Starting from a set of points of interest, a Harris corner lying on one contour at least, the descriptor is a vector containing the *21* photometric invariant moments extracted over the patches defined by a ribbon. Thus for a point of interest there are two descriptors, one for the brighter ribbon and another for the darker side. These ribbons emanate from the point of interest. The Euclidean distance among descriptors in both images are computed and that way we can have an initial estimation of corresponding contours.

However, the descriptor should combine geometric and photometric properties. In this chapter we have defined invariant regions from the affine arc-length distance of corresponding contours but it needs to know the counterpart contour. That is sorted out by a combinatorial search over features extracted from the graph structure. Affine arc length frames are extracted along the contour that links two points of interest. These frames define the regions where to analyse the photometry by the generalised colour moments descriptor.

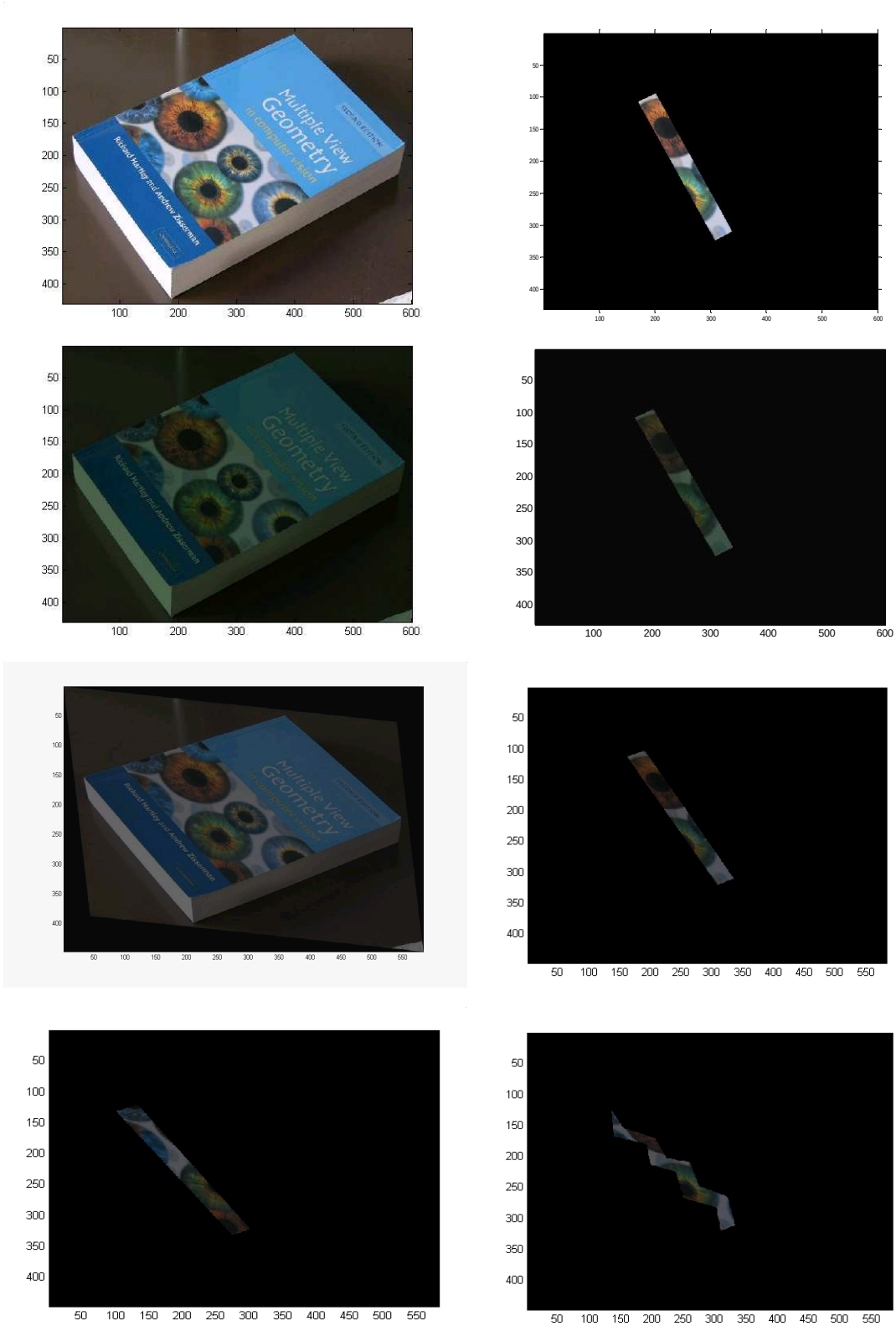


Figure 4.8. Extraction of patterns to compute the invariant moment signature. a) and b) Original image and patch extraction; c) and d) Affinely transformed photometry; e) and f) Affinely transformed geometry and scaled transformed photometry; g) non corresponding patch; h) non-corresponding patch with different morphology.

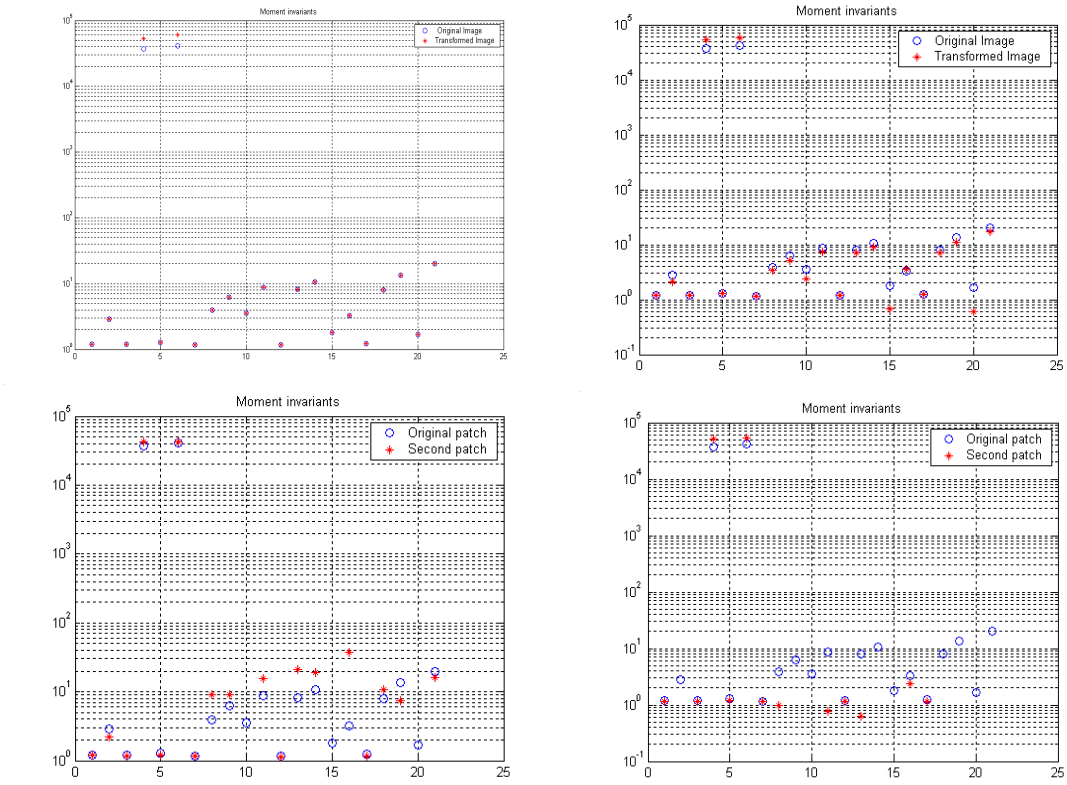


Figure 4.9. Comparison of two sets of 21 moment invariants resulting from a) two corresponding geometric patches with photometry affinely transformed (figure 4.8b) and d)); b) two corresponding patches geometrically transformed and scaling of RGB bands (figure 4.8b) and f)); c) two non-corresponding patches geometrically and photometrically transformed (figure 4.8b) and h)); and d) two non-corresponding patches with different morphology. Notice that the values of some moment invariants are missing, that is due to the non representation of negative values in logarithmic axis.

4.5 Experimental results

Regions along the contour. The proposed algorithm was tested on the real contours and synthetic curves over an indoor image already presented (see figure 4.2). The image is an RGB image with a resolution of *384x512* pixels. A contour map is created by extracting Canny edges. The Gaussian filter σ is *1.35* and hysteresis thresholds are set to *0.0312* and *0.0781*. There were *450* edges found that after linking according to gradient and direction maps, proximity, continuity and distance constraints were reduced to *96* contours of a minimum distance of *30* pixels. Four synthetic curves were overlapped over the image, resulting in a set of *100* contours in total. Seven landmark points were

selected manually on the reference image. They could have been extracted by looking for Harris pixels over contours or in the proximity of contours but for the purpose of the experiments this suffices. The original image is applied an affine transformation *M*, a *20º* rotation, *100* pixels translation in the *x*-axis, and *0.9* and *0.2* scaling and shear in both *x*- and *y*-axis respectively. A scale plus offset photometric transformation was also applied, scale *[0.6 0.6 0.7]* and offset *[-0.2 -0.2 0.1]* in the *RGB* bands.

From the points of interest we extract homogeneous photometric regions at both sides of the contours. These ribbons are delimited by the contour map as explained in Chapter 3. Figure 4.10 shows the regions extracted for the points of interest lying over/by the synthetic curves. Notice that the extraction of regions is also expanded to the neighbouring contours within a certain distance from the point of interest. That is, the algorithm starts from the point of interest, opens a small window and search for neighbour contours. Ribbons are extracted around the contours where there exists a point of interest in the vicinity. The rest of the contour map is only taken under consideration for delimiting homogeneous regions. Regions emanating from points of interest close to real contours are shown in figure 4.11. Every pair of region is classified as ‘darker’ and ‘brighter’ side of the contour, by averaging grey levels. The affine photometric invariant moments are computed over these regions for every point of interest. Therefore, the point of interest is defined by two vectors of *21* moments for each side of the contour(s). We use Euclidean distance to find the proximity among descriptors in both images. Tables 4.1 and 4.2 present the results for the points of interest lying over the curves under study. We present in green correct matches, in red mismatches and in orange the corresponding match. We are analysing here the performance of the extraction of homogeneous photometric regions as well as the invariance of the descriptor towards geometry and photometry changes. The distance matrices allow us to have an initial estimation of corresponding points over contours.

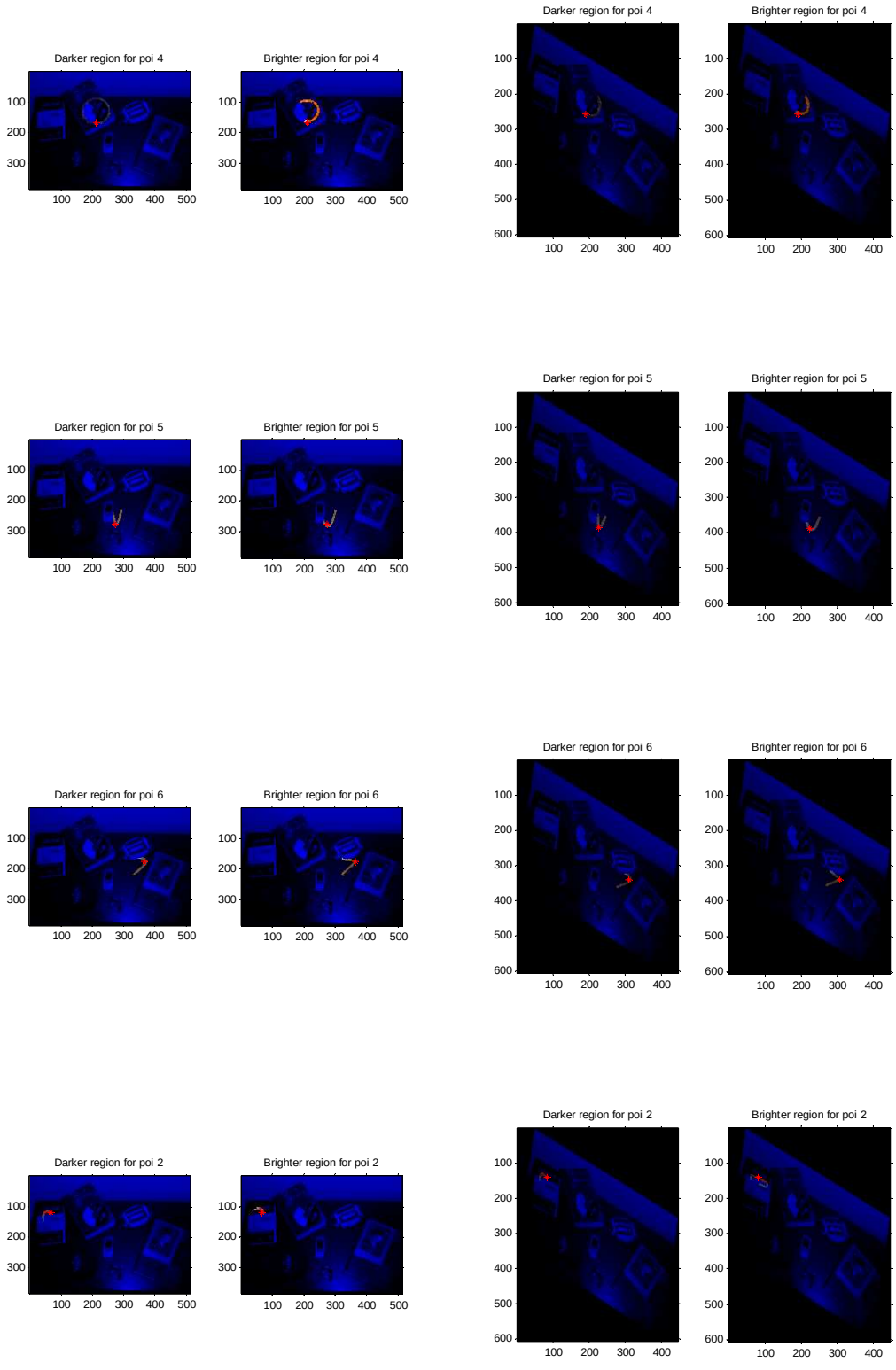


Figure 4.10. Homogeneous photometric regions from points of interest lying over synthetic curves. Left) original image and right) transformed image.

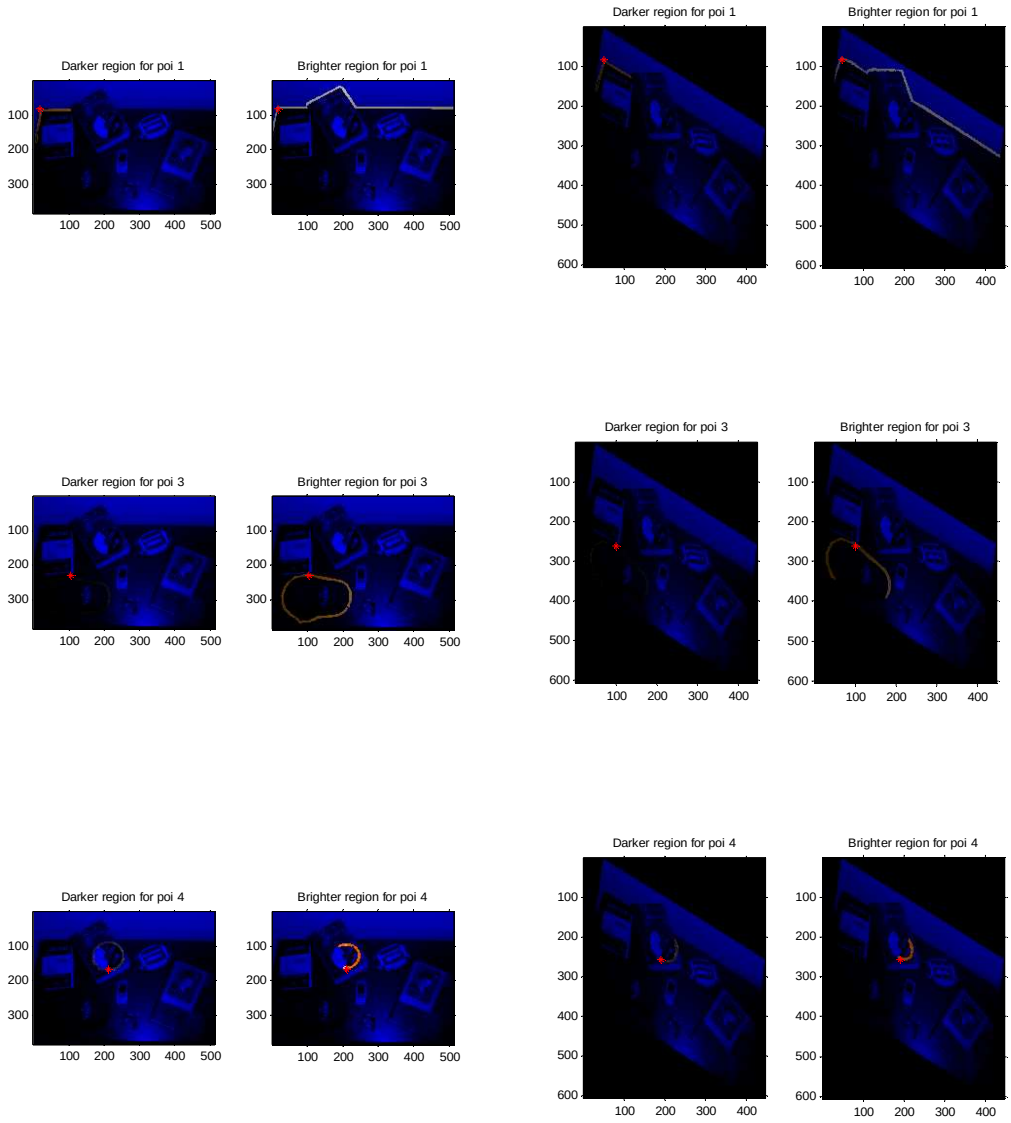


Figure 4.11. Homogeneous photometric regions from points of interest lying over real contours. Left) original image and right) transformed image.

Distance matrix for darker side							
	Circle	**Parabola**	**Sin-exp**	**Ellipse**	**C1**	**C2**	**C3**
Circle	1.7688	3.8602	9.5013	2.0559	3.4642	2.8978	3.9281
Parabola	3.8940	1.9928	9.5730	4.3539	5.2931	4.8164	5.3342
Sin-exp	8.6256	8.5970	9.0480	8.6658	9.3747	9.0764	9.8345
Ellipse	2.4610	4.2999	9.4541	1.5100	2.8824	2.5515	3.6552
C1	4.4571	4.8553	10.2390	3.0956	1.8963	2.2314	3.7357
C2	3.3103	4.8553	9.7509	2.3659	2.0661	1.8382	3.4422
C3	5.4207	6.6489	12.9295	5.5166	5.4581	5.3388	3.8689

Table 4.1. Distance matrix among descriptors based on invariant photometric moments for the darker side of the contour. Rows, original image. Columns, transformed image.

Distance matrix for brighter side							
	Circle	**Parabola**	**Sin-exp**	**Ellipse**	**C1**	**C2**	**C3**
Circle	2.7144	8.3006	8.9197	4.7941	3.7586	3.6892	6.2086
Parabola	6.7288	5.9632	6.1622	4.2438	4.8347	4.2472	6.1573
Sin-exp	5.2813	9.5286	9.7746	6.2698	5.7025	5.5429	8.0059
Ellipse	7.4692	5.6224	7.6261	3.5196	5.3378	5.0944	6.2336
C1	5.8041	6.7267	6.6712	4.7079	2.6492	2.9260	3.4681
C2	5.6476	6.5232	5.5123	3.8103	2.4245	2.1897	3.7451
C3	8.9412	7.2558	7.8222	7.8230	6.5349	6.6196	3.4455

Table 4.2. Distance matrix among descriptors based on invariant photometric moments for the brighter side of the contour. Rows, original image. Columns, transformed image.

Affine invariant frames. We compute the affine arc-length frames over the sample desk scene with synthetic and real contours independently detected to show with these examples how the affine invariant regions (parallelograms) are extracted. We can set up a relation between affine arc-length of potential corresponding contours and extract invariant regions based on affine arc-length. The contours are reparameterised from centripetal to affine arc-length distance. The terms of the affine arc-length in equation (4.3), *i.e.* up to second order derivatives, determinant and integral, are computed in the domain of splines. The computations of derivatives in finite differences introduce considerable errors. Therefore, a least-median of squares cubic spline approximation is a better solution [99]. The ratios of affine arc-lengths of corresponding contours to the third power, equation (7), give an estimation of the determinant of the fundamental matrix. Table 4.3, shows the results in an exhaustive search for corresponding contours.

The determinant of our transformation M is *0.77*, which is accurately obtained for synthetic curves. For the three real contours, the results are also very accurate (*0.7419*, *0.7369* and *0.8542*). Figures 4.12 to 4.18 depict extraction of the invariant regions for synthetic and real contours. The figures on the left represent the extraction of patches defined by the affine invariant vectors over a few contour samples, for a better visualisation. We find acceptable performance of the system over the synthetic curves, whereas real contours do not show satisfactory results. Recall that second order derivatives are sensitive to noise. Affine curvature could have been a useful invariant but its expression contains fourth order derivatives, which rules it out of any practical consideration for us.

Affine arc-length ratios							
	Circle	**Parabola**	**Sin-exp**	**Ellipse**	**C1**	**C2**	**C3**
Circle	**0.7700**	0.0220	0.0199	0.1084	5.2935	6.8597	0.5679
Parabola	26.9154	**0.7700**	0.6945	3.7897	185.0361	239.7812	19.8511
Sin-exp	29.8435	0.8538	**0.7700**	4.2020	205.1662	265.8672	22.0107
Ellipse	5.4687	0.1565	0.1411	**0.7700**	37.5961	48.7194	4.0334
C1	0.1079	0.0031	0.0028	0.0152	**0.7419**	0.9614	0.0796
C2	0.0827	0.0024	0.0021	0.0116	0.5687	**0.7369**	0.0610
C3	1.1582	0.0331	0.0299	0.1631	7.9622	10.3179	**0.8542**

Table 4.3. Affine arc-length ratios of curves in the proximity of points of interest. Rows, original image. Columns, transformed image.

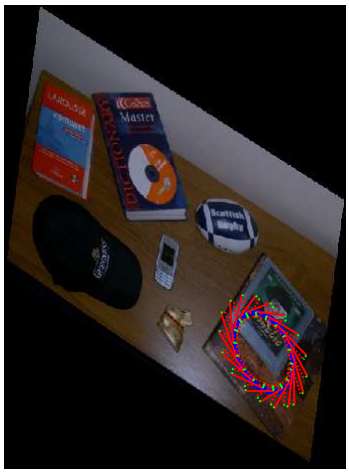


Figure 4.12. Affine invariant arc-length frames over synthetic curve. Circle.

Figure 4.13. Affine invariant arc-length frames over synthetic curve. Parabola.

Figure 4.14. Affine invariant arc-length frames over synthetic curve. Sine-exponential.

Figure 4.15. Affine invariant arc-length frames over synthetic curve. Ellipse.

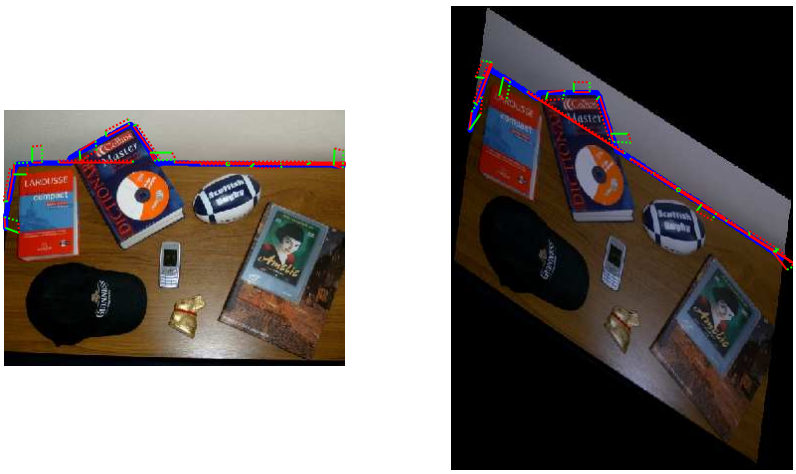


Figure 4.16. Affine invariant arc-length frames over synthetic a real contour. C1.

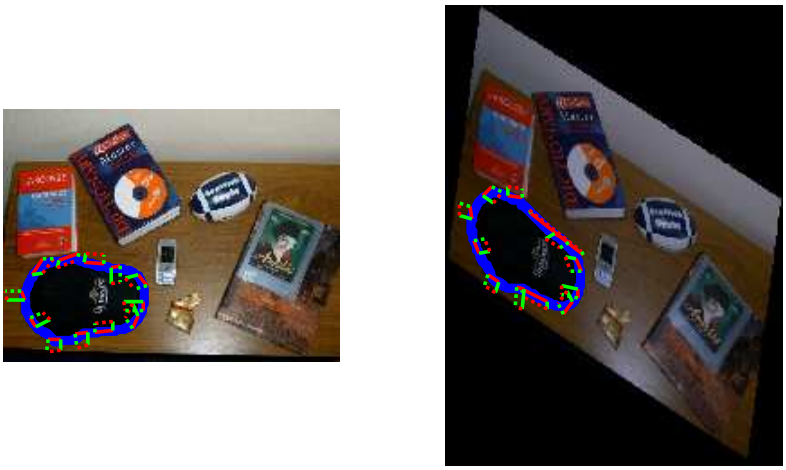


Figure 4.17. Affine invariant arc-length frames over a real contour. C2.

Figure 4.18. Affine invariant arc-length frames over a real contour. C3.

In figure 4.18 we can notice that the length of the vector a_1 for some of the samples is of considerable magnitude. The inconvenience of this is that the descriptor extract regions that are not local. In figure 4.19 we present a simple experiment. In the left figure we

plot the initial sequences of the affine invariant frame for samples along the curve. Notice how the vectors a1 in red and a2 in green lie at each side of the contour. In the right-hand side figure we show the whole sequence along the contour. There is a sample, marked by the arrow, where the vectors swing. That occurs at an inflection point. At that inflection point the determinant of equation (4.2) is null, the vectors overlap and go to infinity in order to describe a planar parallelogram of unit area. Figure 4.20 shows the value of the determinant of the derivatives along the samples of the contour. The original curve is overlap by a rotated version, which means that the determinant is invariant to rotations. The determinant of other affine versions of the input curve are also displayed. We can see that the zero crossing point of all the curves corresponds to a point of null determinant, or null affine curvature. We can discern from that that these inflection points are invariant to affine transformations.

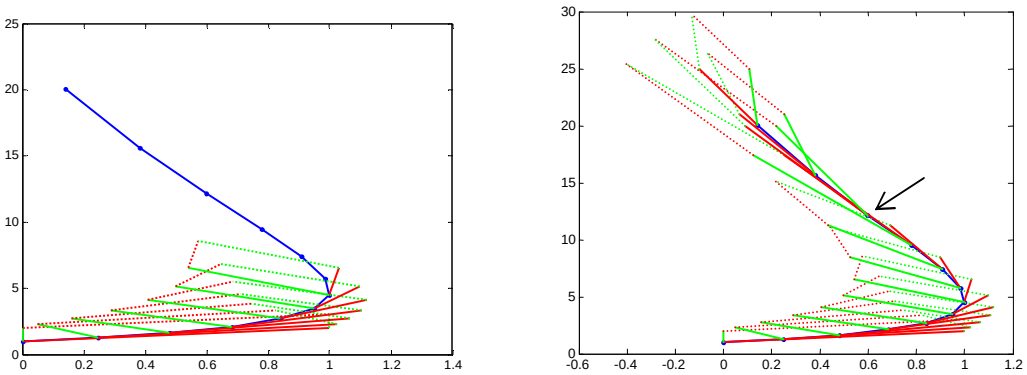


Figure 4.19. Effect of inflection over the affine invariant frame..

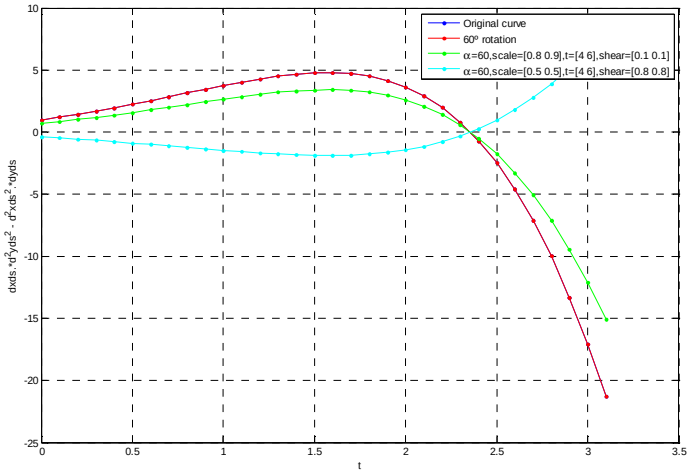


Figure 4.20. Determinant of the derivatives for different transformations.

Geometric and Photometric affine invariant approach. We combine the affine geometric invariant frame with the generalised colour moments in this section. The algorithm is presented in pseudo-code in figure 4.21. Basically, it consists of an exhaustive search over the space of spatial descriptors generated in Chapter 3 from the graph structure. The search space is reduced by setting certain constraints. For instance, we do not take samples when the magnitude of one of the vectors of the affine frame exceeds a certain magnitude, since we would not be extracting local regions (see figure 4.18a). That is caused by the samples where the determinant in equation (4.2) is close to zero, points of inflection of null affine curvature. Therefore, we do not consider these regions along the contour. Another constraint is to delimit the transformation the system can cope with. From equation (4.8), the determinant of the transformation M is a function of the ratio of two corresponding affine arc-length distances. If the determinant of M is too high or too low, both descriptors could only correspond each other when that strong transformation occurs. If we bound the space of possible transformations we are also reducing the search space. The re-scaling of the affine frame in image B is given by equation (4.9), with the assumption that these two spatial descriptors correspond. We define a grid over this re-scaled affine frame and interpolate the photometry of the image and apply the generalised colour moments descriptor. We also store the normalised affine arc-length of both spatial descriptors. However, despite that this measure is expected to be an absolute invariant, results are not so good when incorporating this measure in real applications due to the sensitivity to noise. Therefore, the only metric used to measure the distance between the two descriptors is the Euclidean distance of the natural logarithm of the generalised colour moment vectors. The voting algorithm casts votes row- and column-wise over the distance matrix of the descriptors. We cast votes only to the best 8 matches along each column of the descriptor matrix (votes v = [10 8 6 5 4 3 2 1]). We do the same row-wise and after multiply both matrices. The result matrix is weighted by the inverse of the Euclidean distance matrix. The potential correspondences are the ones with higher scores. We take as a match the pair with maximum score across its column- and row-wise location in the confusion matrix. In [5] and previous works referred there, a match is assigned when the distance between the pair is lower than 0.7 times the distance of the second best pair. However, this measure did not obtain more successful results for our setting. As another strategy, the Munkres algorithm has also been used for optimization in the assignment process.

In:

- Descriptors with spatial information from graph structures from both images

Out:

- Set of correspondences

Procedure:

1. FOR every descriptor from image A
    a. Extract affine arc-length σ_A
    b. Discard samples when the magnitude of the affine vectors exceed a predefined threshold
    c. FOR every descriptor from image B
        i. Extract affine arc-length σ_B
        ii. Estimate the determinant of the fundamental matrix |M| between the pair of contour segments (equation (4.8))
        iii. IF (|M|>maxoffset OR |M|<minoffset)
            CONTINUE – The descriptors can only correspond if a strong transformation that is out of consideration occurs
            END
        iv. Define affine invariant regions in image B (equation (4.7))
        v. Discard samples when the magnitude of the affine vectors exceed a predefined threshold
        vi. Set grid over affine invariant frames in image B
        vii. Extract photometry over samples in the grid
        viii. Compute the generalised colour moments
        ix. Compute distance between normalized affine arc-length of both descriptors
        END
    d. Set grid over affine invariant frames in image A (equation (4.1))
    e. Extract photometry over samples in the grid
    f. Compute the generalised colour moments
    g. Compute Euclidean distance between both descriptors
    END
2. Voting algorithm
3. Return set of correspondences

Figure 4.21. Geometric and photometric affine invariant algorithm.

We perform now experiments on the extraction of affine invariant arc-length frames over the images used in Chapter 3 (figures 3.30 to 3.34), *i.e.* the output to the affine invariant system is the graph structure in previous chapter. Since the results in the previous experiments were not satisfactory for real contours, we do not use the pair of stereo images but a original one and its affinely transformed (homography). The ground truth permits us visualise how accurate the matching is. The transformation applied for each experiment is summarised in Table 4.4. The results are displayed in the confusion matrices of figures 4.22 to 4.26 and the measure of recall, precision and number of corresponding regions in figure 4.27 to 4.29.

Homography	Rotation	Scale	Shear	Phot_offset	Phot_scale
1	20	1	0	0	1
2	0	2	0	0	1
3	0	2	0.1	0	1
4	0	2	0	0.2	0.7
5	20	0.75	0.1	0	1
6	20	0.75	0.3	0	1
7	20	1.25	0.1	0	1
8	20	1.25	0.3	0	1
9	40	0.75	0.1	0	1
10	40	0.75	0.3	0	1
11	40	1.25	0.1	0	1
12	40	1.25	0.3	0	1

Table 4.4. Space of transformations

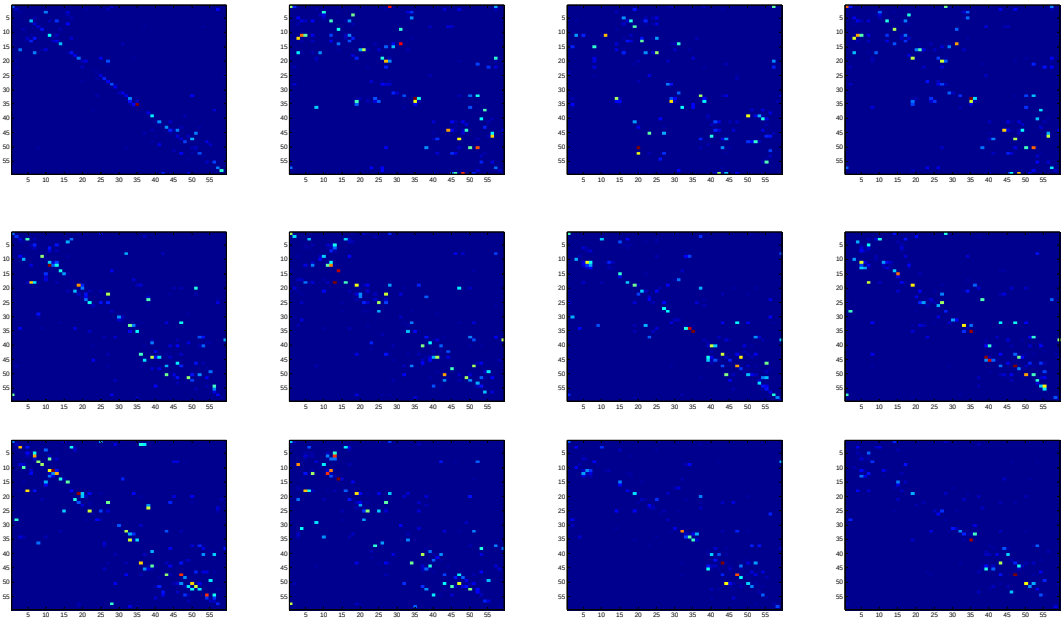


Figure 4.22.Confusion matrices. Book scene.

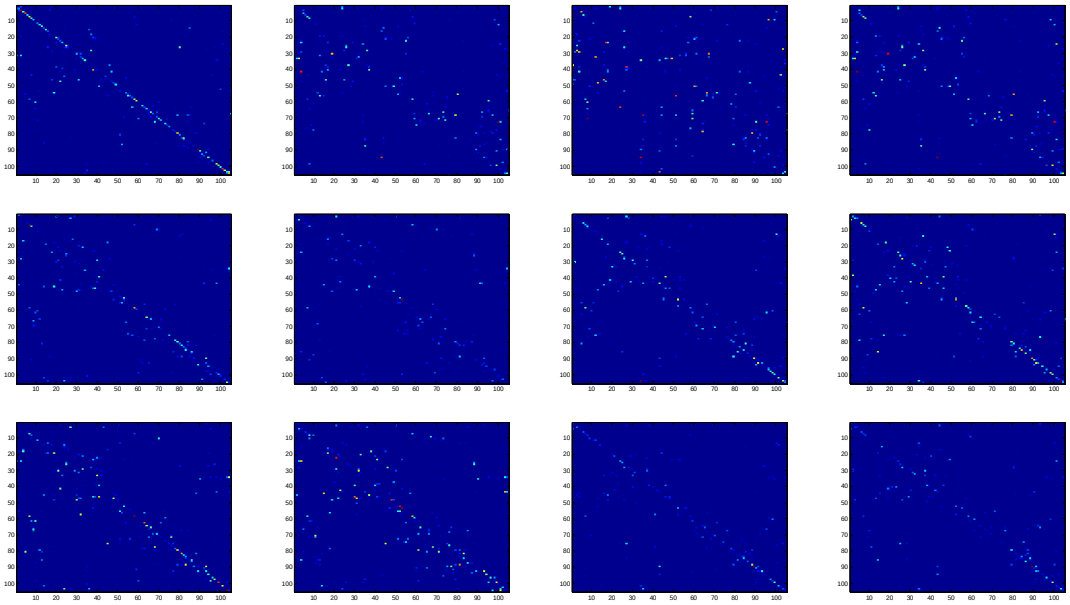


Figure 4.23.Confusion matrices. Antenna scene.

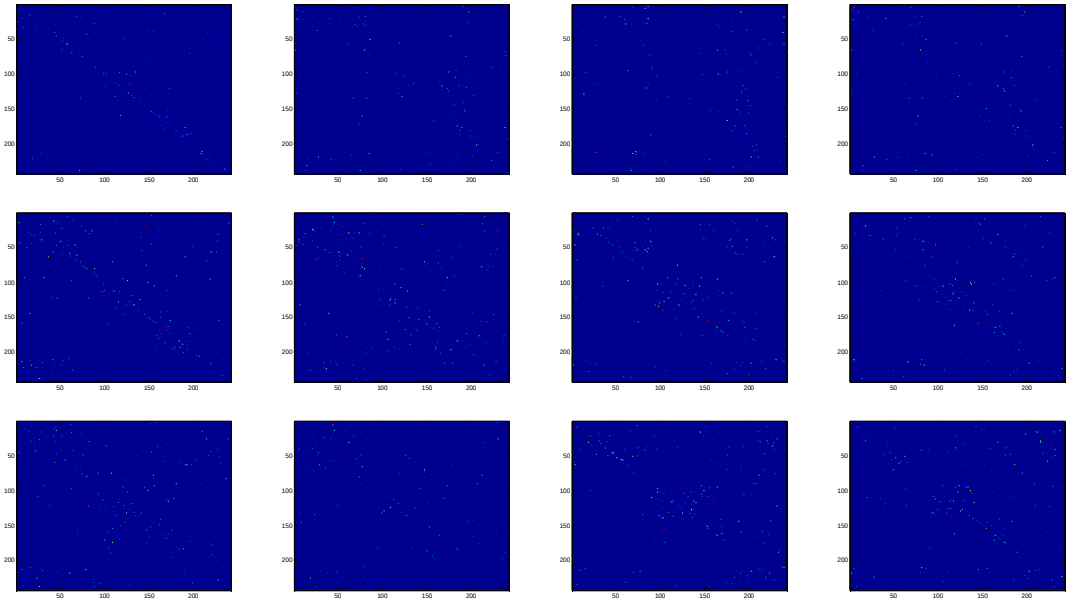


Figure 4.24.Confusion matrices. Countryside scene.

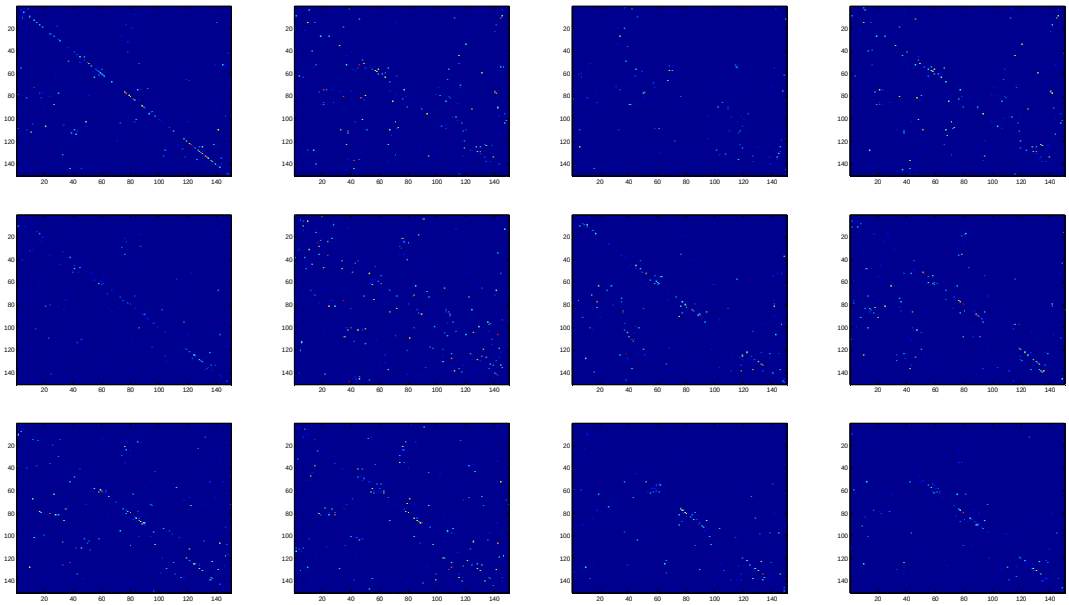


Figure 4.25.Confusion matrices. Graffiti scene.

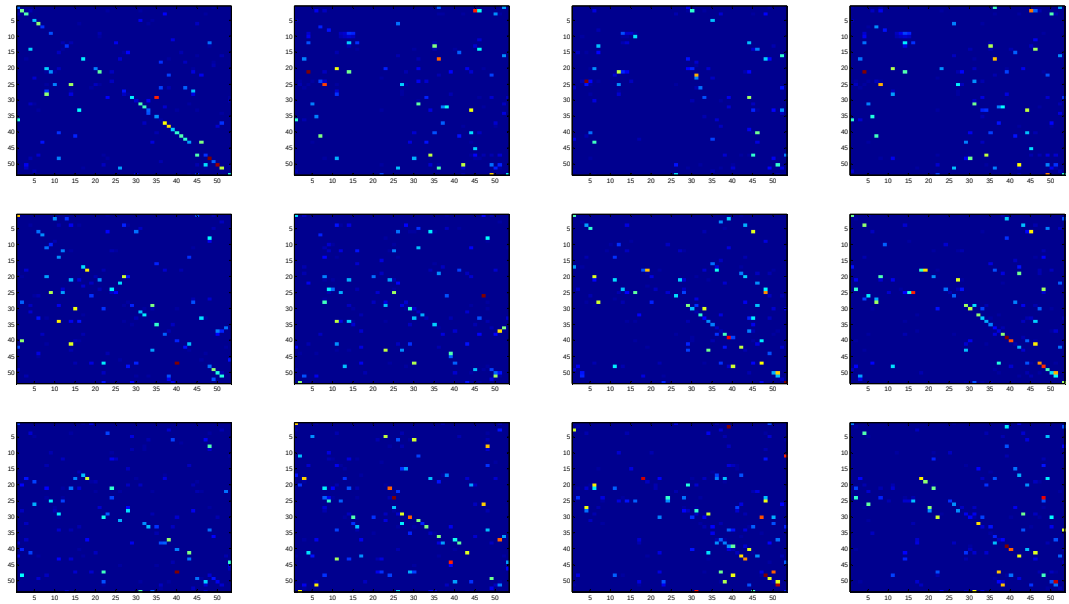


Figure 4.26.Confusion matrices. Valbonne scene.

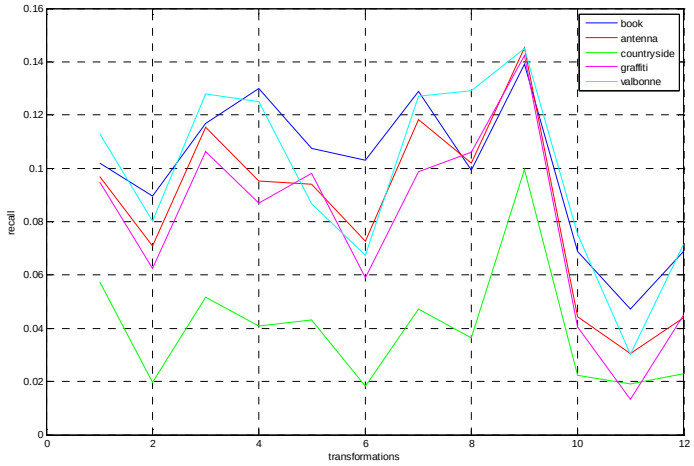


Figure 4.27.Recall

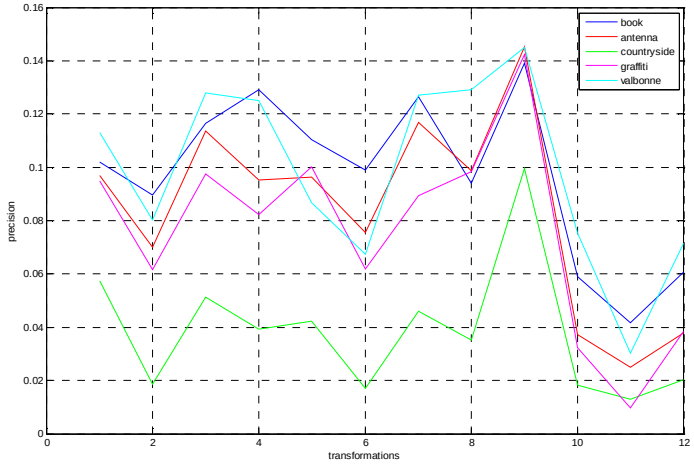


Figure 4.28. Precision

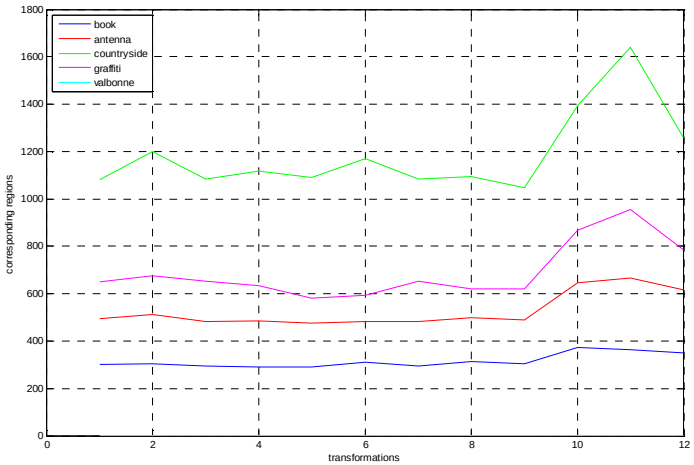


Figure 4.29.Number of corresponding regions

4.6 Error analysis

4.6.1 Propagation of errors

The affine arc-length method for the extraction of affine invariant regions was strongly dependent on the nature of the curves. The results proved to be satisfactory for synthetic curves, whereas rather the opposite for contours from real images. Splines based upon least-mean of squares fairly approximate our synthetic curves but present a little, practically insignificant errors over real contours. We analyse how these small errors propagate through the experimental procedure to a bigger error in the final result. The final error is as a by-product of the combination of the uncertainty for each single step that leads to the affinely invariant arc-length vectors.

Figure 4.30 shows a diagram with the main steps and how the error propagates. The *xy* coordinates of the curve in image *A* are applied an affine transformation *T* to generate the *xy* coordinates of the curve in image *B*. That curve is approximated by splines, introducing an error that propagates throughout the next blocs highlighted in red. In the other hand, we also transform the approximation by splines from image *A* into image *B* by using the same *T*. The error that propagates in further steps is null. That is due to the fact that the input error is zero and no more approximations happen in further steps. Therefore, we can consider the blocks highlighted in red as ground truth for the evaluation of the propagating error. Next we introduce some basics on the theory of error propagation [15].

If x is a function of two variables u and v, $\tilde{x}$ its expected value based on ground truth variables $\tilde{u}$ and $\tilde{v}$, and x_i the consequence of each individual measurement u_i and v_i, therefore the variance of x is given by:

$$s_x^2 = \lim_{N\to\infty} \frac{1}{N} \sum \left(x_i - \tilde{x}\right)^2 \tag{4.15}$$

By expressing the deviations of x as a function of its variables u and v:

$$x_i - \tilde{x} \approx \left(u_i - \tilde{u}\right)\left(\frac{\partial x}{\partial u}\right) + \left(v_i - \tilde{v}\right)\left(\frac{\partial x}{\partial v}\right) \tag{4.16}$$

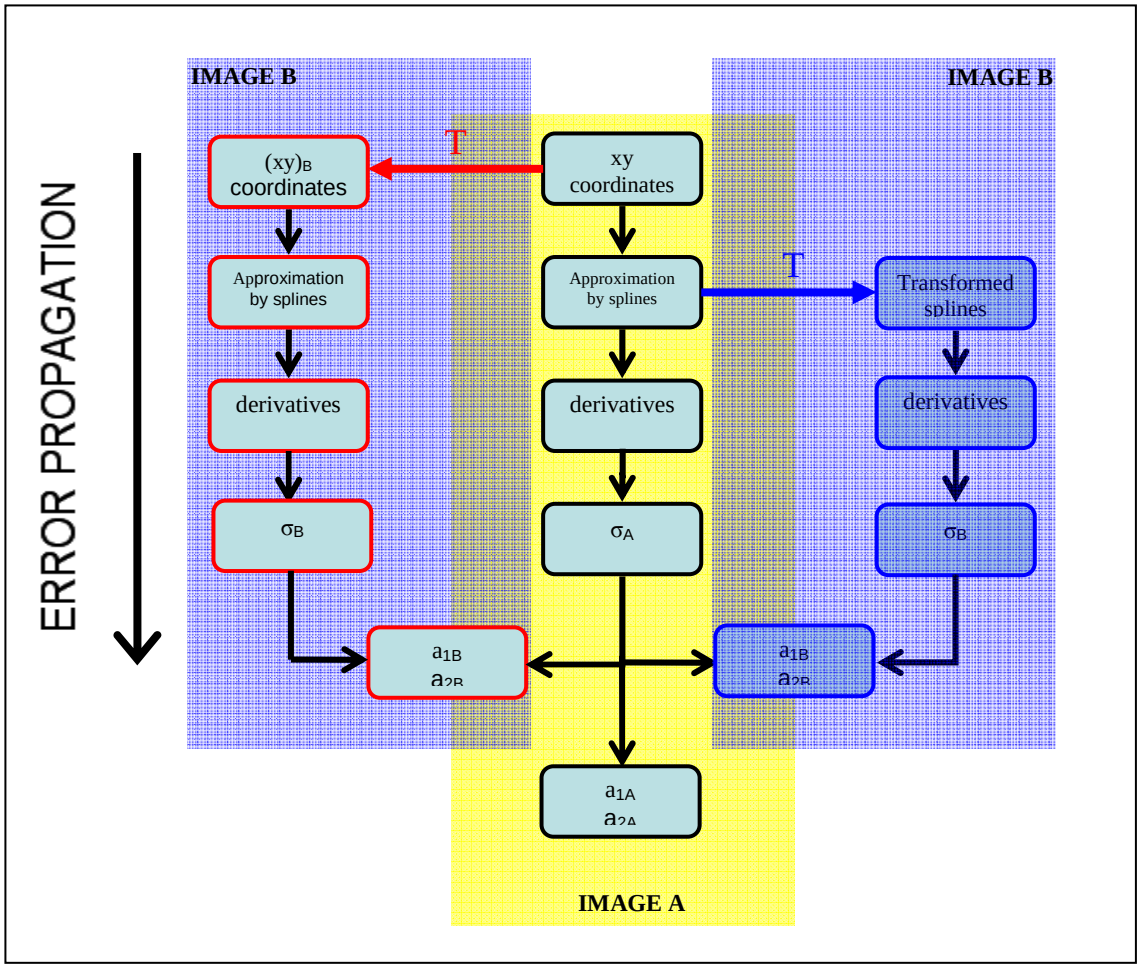


Figure 4.30. Error propagation across the calculus of affine invariant frames.

the variance s_x^2 can be expressed in terms of the deviation of the variables *u* and *v*:

$$\lim_{N\to\infty}\frac{1}{N}\sum\left[(u_i-\tilde{u})^2\left(\frac{\partial x}{\partial u}\right)^2+(v_i-\tilde{v})^2\left(\frac{\partial x}{\partial v}\right)^2+2(u_i-\tilde{u})(v_i-\tilde{v})\left(\frac{\partial x}{\partial u}\right)\left(\frac{\partial x}{\partial v}\right)\right] \quad (4.17)$$

and that way it can also be expressed as a function of the variance and covariance of *u* and *v*:

$$s_x^2 \approx s_u^2\left(\frac{\partial x}{\partial u}\right)^2+s_v^2\left(\frac{\partial x}{\partial v}\right)^2+2s_{(u)(v)}^2\left(\frac{\partial x}{\partial u}\right)\left(\frac{\partial x}{\partial v}\right)^2 \quad (4.18)$$

In our particular case our functions under study are the affine arc-length expression $\sigma(t)$ in equation (4.3), and the affine invariant vectors in the transformed image, equations (4.1) and (4.9).

If we dissect hierarchically the expression of the affine arc-length, there is a summation, a third root and a determinant, which is a subtraction of products of first and second order derivatives of the *x* and *y* components of the curve. Likewise, the final vectors in image *B* (we do not consider vectors in image *A* since we assume no error propagation in the original image) is the result of a determinant, a product by its derivatives and another product with the division of affine arc-lengths to the cube. We analyse the propagation of the uncertainties throughout the expressions in figure 4.31.

4.6.2 Experimental results

In figure 4.32 we present an example of the propagation of errors to the affine invariant frame. The first image corresponds to the original image, in the second image the splines have been transformed from image A and in the bottom image the contour coordinates were transformed and these were the seed to the frames. Notice that the affine frames in the central image covers corresponding areas to the ones in the first image; whereas in the bottom image the parallelograms do not correspond exactly. Figures 4.33 to 4.35 show the propagation of errors across the expressions in figure 4.31. These are calculated for an affine frame of unit area, *i.e.* r=[1 1]. See in expressions S11 and S12 in figure 4.28 that the effect of scaling the affine vectors by r implies a multiplication of the error by r^2. Figure 4.36 shows the distribution of the errors in the affine frame as a function of the determinant of the derivatives of the contour for the ground truth (from transformation of splines) and for the measured data (from the transformation of xy coordinates). Notice again how samples where the value of the determinant is low tend to have higher errors.

$$S_1 \equiv s^2_{x'y''} = (\tilde{y}'')^2 s^2_{x'} + (\tilde{x}')^2 s^2_{y''} + 2\tilde{x}'\tilde{y}'' s^2_{(x')(y'')}$$

$$S_2 \equiv s^2_{x''y'} = (\tilde{y}')^2 s^2_{x''} + (\tilde{x}'')^2 s^2_{y'} + 2\tilde{x}''\tilde{y}' s^2_{(x'')(y')}$$

$$S_3 \equiv s^2_{x'y''-x''y'} = s^2_{x'y''} + s^2_{x''y'} + 2s^2_{(x'y'')(x''y')}$$

$$S_4 \equiv s_{(x'y''-x''y')^{\frac{1}{3}}} = (\tilde{x}'\tilde{y}''-\tilde{x}''\tilde{y}')^{\frac{1}{3}} \cdot \frac{1}{3} \cdot \frac{s_{(x'y''-x''y')}}{(\tilde{x}'\tilde{y}''-\tilde{x}''\tilde{y}')}$$

$$S_5 = s^2_{\sigma_B} \equiv s^2_{\sum_{i=0}^{b}(x'y''-x''y')^{\frac{1}{3}}} = \sum_{i=0}^{b} s^2_{(x'y''-x''y')^{\frac{1}{3}}}$$

$$S_6 \equiv s_{(x'y''-x''y')^{-\frac{1}{2}}} = (\tilde{x}'\tilde{y}''-\tilde{x}''\tilde{y}')^{-\frac{1}{2}} \cdot \frac{1}{2} \cdot \frac{s_{(x'y''-x''y')}}{(\tilde{x}'\tilde{y}''-\tilde{x}''\tilde{y}')}$$

$$S_7 = s^2_{a_{1B}} \equiv s^2_{(x'y''-x''y')^{-\frac{1}{2}} \cdot \begin{pmatrix} x' \\ y' \end{pmatrix}} = \begin{pmatrix} \tilde{x}' \\ \tilde{y}' \end{pmatrix}^2 \cdot s^2_{(x'y''-x''y')^{-\frac{1}{2}}} + \left((\tilde{x}'\tilde{y}''-\tilde{x}''\tilde{y}')^{-\frac{1}{2}}\right)^2 \begin{pmatrix} s^2_{x'} \\ s^2_{y'} \end{pmatrix} + 2 \cdot (\tilde{x}'\tilde{y}''-\tilde{x}''\tilde{y}')^{-\frac{1}{2}} \cdot \begin{pmatrix} \tilde{x}' \\ \tilde{y}' \end{pmatrix} \cdot s^2_{\left((x'y''-x''y')^{-\frac{1}{2}}\right)\begin{pmatrix} x' \\ y' \end{pmatrix}}$$

$$S_8 = s^2_{a_{2B}} \equiv s^2_{(x'y''-x''y')^{-\frac{1}{2}} \cdot \begin{pmatrix} x'' \\ y'' \end{pmatrix}} = \begin{pmatrix} \tilde{x}'' \\ \tilde{y}'' \end{pmatrix}^2 \cdot s^2_{(x'y''-x''y')^{-\frac{1}{2}}} + \left((\tilde{x}'\tilde{y}''-\tilde{x}''\tilde{y}')^{-\frac{1}{2}}\right)^2 \begin{pmatrix} s^2_{x'} \\ s^2_{y'} \end{pmatrix} + 2 \cdot (\tilde{x}'\tilde{y}''-\tilde{x}''\tilde{y}')^{-\frac{1}{2}} \cdot \begin{pmatrix} \tilde{x}'' \\ \tilde{y}'' \end{pmatrix} \cdot s^2_{\left((x'y''-x''y')^{-\frac{1}{2}}\right)\begin{pmatrix} x'' \\ y'' \end{pmatrix}}$$

$$S_9 \equiv s^2_{\frac{\sigma_B}{\sigma_A}} = \tilde{\sigma}^2_A \cdot s^2_{\sigma^2_B} + \tilde{\sigma}^2_B \cdot s^2_{\sigma^2_A} - 2 \cdot \tilde{\sigma}_B \cdot \tilde{\sigma}_A \cdot s^2_{(\sigma_B)(\sigma_A)}$$

$$S_{10} \equiv s_{\left(\frac{\sigma_B}{\sigma_A}\right)^3} = \left(\frac{\tilde{\sigma}_B}{\tilde{\sigma}_A}\right)^3 \cdot 3 \cdot \frac{s_{\frac{\sigma_B}{\sigma_A}}}{\left(\frac{\tilde{\sigma}_B}{\tilde{\sigma}_A}\right)}$$

$$S_{11} \equiv s^2_{ad\_a1b} = r^2(1) \cdot \tilde{k}^2 \cdot s^2_{a1B} + r^2(1) \cdot \left(\frac{\tilde{a}_1(x)}{\tilde{a}_1(y)}\right)^2 \cdot s^2_k + 2 \cdot r^2(1) \cdot \left(\frac{\tilde{a}_1(x)}{\tilde{a}_1(y)}\right) \cdot \tilde{k} \cdot \left[(a_1 - \tilde{a}_1) \cdot (k - \tilde{k})\right]$$

$$S_{12} \equiv s^2_{ad\_a2b} = r^2(2) \cdot \tilde{k}^2 \cdot s^2_{a2B} + r^2(2) \cdot \left(\frac{\tilde{a}_2(x)}{\tilde{a}_2(y)}\right)^2 \cdot s^2_k + 2 \cdot r^2(2) \cdot \left(\frac{\tilde{a}_2(x)}{\tilde{a}_2(y)}\right) \cdot \tilde{k} \cdot \left[(a_2 - \tilde{a}_2) \cdot (k - \tilde{k})\right]$$

Figure 4.31. Error propagation across the calculus of affine invariant frames.

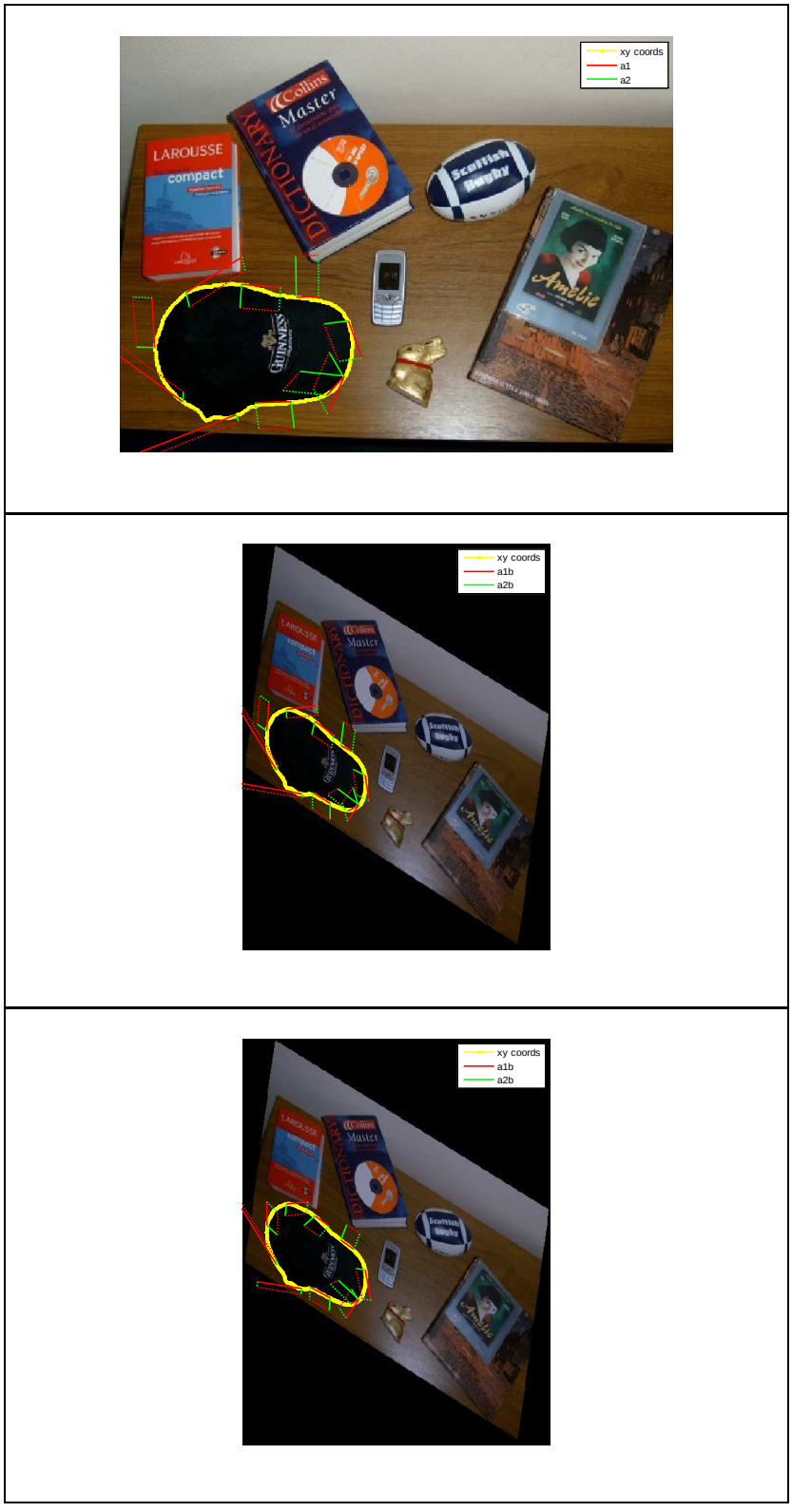


Figure 4.32. Propagation of error over affine invariant frames (*r*=[10 80]). a) Original image, b)transformed image with spline approximations transformed 1:1 and c)transformed image with transformed *xy* contour coordinates.

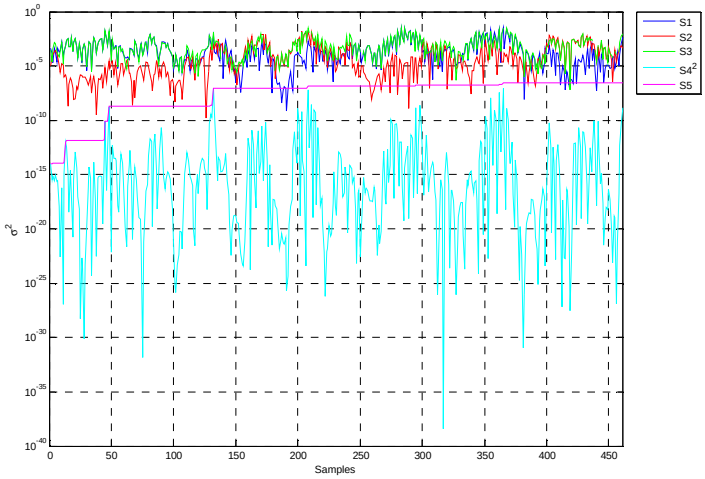


Figure 4.33. Propagation of the error along the contour for the affine arc length.

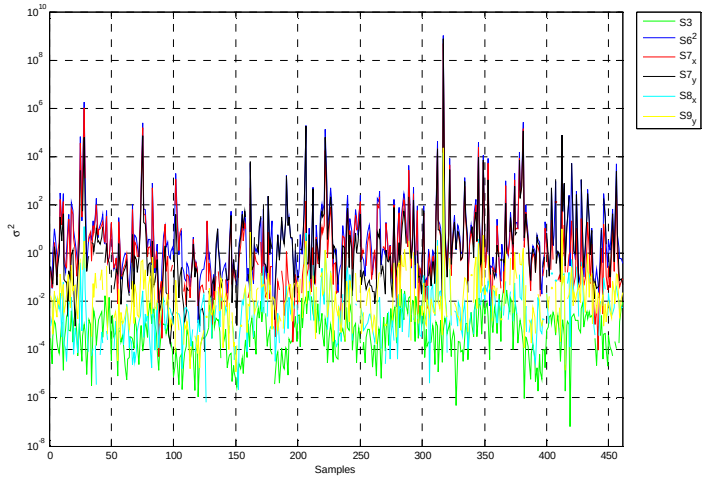


Figure 4.34. Propagation of the error along the contour for the affine arc-length ratios.

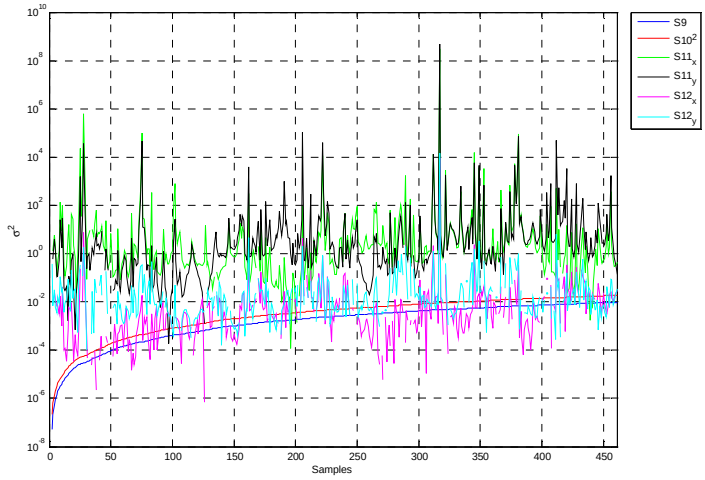


Figure 4.35. Propagation of the error along the contour for the affine arc length frames.

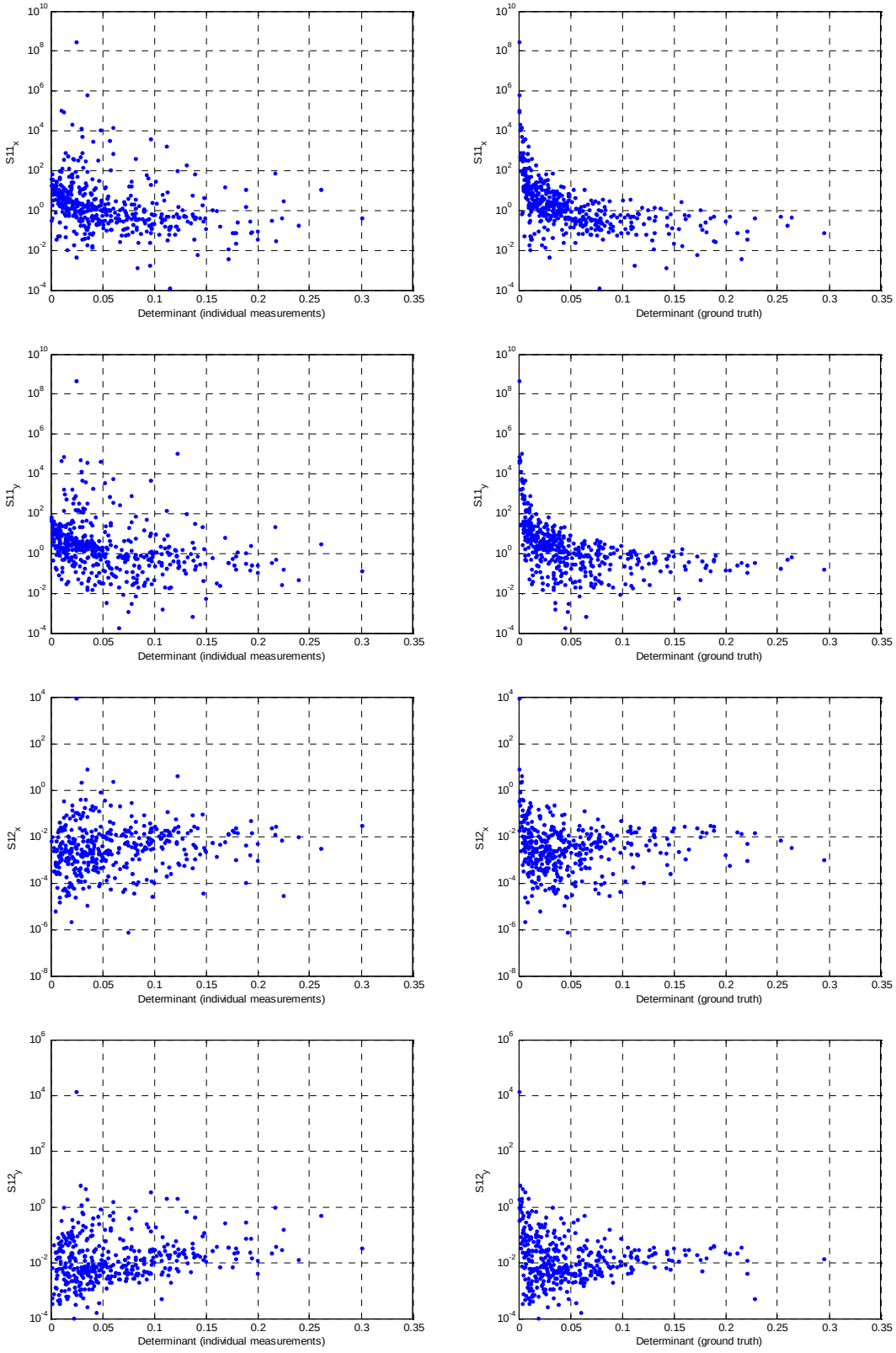


Figure 4.36. Error in the affine frame as a function of the determinant of the derivatives.

4.7 Summary

We have presented two methods for solving the correspondence problem over affinely transformed images. The first one consists of the extraction of regions of homogeneous photometry around contours. The generalised colour moments are used to describe the photometry in the regions and the matching is performed by Euclidean distance between descriptors and a voting algorithm. For the sample image containing synthetic and real contours the system works well. But this method is ad-hoc, only invariant to rotation and translations, and highly dependent on the ability of the contours to extract homogeneous photometric regions. So we have performed some tests but we do not consider the method valid for our system.

The second method plays a main role in this thesis. We have described affine geometric invariant frames along segmented contours from a graph structure that are theoretically absolute affine invariants. We have shown how these regions are extracted for synthetic and real contours and run experiments in combination with the generalised colour moments descriptor over the images used in Chapter 3. We have worked with ground truth data, *i.e.* we have taken one of the images and have affinely transformed it together with the high-curvature points and contours. The tests undertaken consisted of several single rotations and shears together with translations, and then combinations of all those (affinities) including also changes in photometry. We have displayed the results of the matchings after applying a voting algorithm in the form of confusion matrices and number of true correspondences. The percentages of correct matches are around the 10%, which implies that the system needs some further support to discern outliers. That also suggests that for a viewpoint change scene where the features are also independently extracted, the chances of a successful matching even decrease further.

We have analysed the reasons why the affine arc-length frames do not perform in practical applications as they do in theory. We have performed an error analysis throughout the steps involved in the calculation of the affine invariant frames. The small error introduced by the approximation by splines – needed for finding the spatial derivatives – propagates throughout the levels being the reason of the malfunction. These errors in the affine frame are more significant for the instances where the determinant of the first and second order derivatives is lower.

That is due to the high magnitude that one of the vectors that define the frame reaches against the magnitude of the other. Therefore, the system is endemic to the accuracy in the computation of the derivatives and at the inflection points where the affine curvature is null. The latter problem can be easily solved by discarding the samples in the contour where the affine curvature (or determinant) is below a certain threshold. The solution to the former problem is more complicated, since it implies the propagation of an input error that is always inherent to any real-world application.

Chapter 5 – Robust estimation from correspondences

5.1 Introduction

In chapter 4 we defined an affine invariant descriptor which embedded an affine arc-length distance and photometric moments. The descriptor was defined along contours' spatial coordinates delimited by points of interest. Consequently, each descriptor paired two points of interest and due to the combinatorics of the graph's approach every point of interest was encoded in at least one descriptor. An initial set of putative correspondences was computed from a confusion matrix. Now in chapter 5 we recapitulate the search of correspondences by strengthening the matching with a robust algorithm. It would be desirable that the data residuals in the sample space are approximately normally distributed. However, that is not what happens in practice since generally there exists outliers or mismatches that cannot be approximated by a normal distribution. If these outliers are considered, the transformation between the two images will not be estimated correctly. It is necessary to use robust algorithms to identifying and discarding the corrupted data.

Some of the robust algorithms are based in non-iterative methods [70,47] but we will centre our attention towards iterative methods. There are two options for the estimation of the parameters of the transformation between the images: either the minimisation of a cost function based on a certain distance metric or the use of the Gold Standard algorithm. The chapter starts with these two approaches and follows with the presentation of classical methods for the rejection of large sets of outliers. Next the whole robust algorithmic approach is presented and we finish with experimental results.

5.2 Cost functions

The projective transformation between two images (either the fundamental matrix or a projectivity) and its nature (perspective or affine) will define the number of degrees of freedom of the transformation and therefore, determine the minimum number of correspondences needed to compute that transformation. That is called the *minimal solution*. In the case that a bigger number of samples is considered (over-determined

system), if the samples in a real application are disturbed by noise, the projective transformation that maps these correspondences may not exist. The problem is reduced to be content with the best possible approximation or *optimal solution* by minimising a cost function which parameters are each pair of correspondences x_i and x_i' and the fundamental matrix F or homography H, *i.e.* the minimisation of the distance between the measured and estimated location of pairs of correspondences. Some examples of cost functions are presented in this thesis as defined in [54]. The cost functions are classified in two groups according to the minimisation of: *a)* an algebraic error, and *b)* a geometric or statistical error. For simplicity the notation is related to the case of computing a homography H $(x'=Hx)$, but it is also the same for the fundamental matrix F $(x'^{T}Fx=0)$ with the difference of computing the distance from the measured correspondence to the estimated epipolar line.

5.2.1 Algebraic distance

If we express each pair of correspondences x_i and x_i' in homogeneous coordinates, *i.e.*: $x_i = (u_i\ v_i\ w_i)^T$ and $x_i' = (u_i'\ v_i'\ w_i')^T$, x_i' and Hx_i will have the same orientation but may have different magnitude up to a scaling factor. The expression can be rearranged in the form of the cross product:

$$x_i' \times Hx_i = 0 \tag{5.1}$$

The term Hx_i can be written as:

$$Hx_i = \begin{pmatrix} h^{1T} x_i \\ h^{2T} x_i \\ h^{3T} x_i \end{pmatrix} \tag{5.2}$$

Being h^{jT} the j-th row of the homography H. Therefore, the cross product is:

$$x_i' \times Hx_i = \begin{pmatrix} v_i' h^{3T} x_i - w_i' h^{2T} x_i \\ w_i' h^{1T} x_i - u_i' h^{3T} x_i \\ u_i' h^{2T} x_i - v_i' h^{1T} x_i \end{pmatrix} \tag{5.3}$$

Taking into account that $h^{jT} x_i = x_i^T h^j$, and from equations *5.1* and *5.3*:

$$\begin{bmatrix} 0^T & -w_i^{'}x_i^T & v_i^{'}x_i^T \\ w_i^{'}x_i^T & 0^T & -u_i^{'}x_i^T \\ -v_i^{'}x_i^T & u_i^{'}x_i^T & 0^T \end{bmatrix} \begin{pmatrix} h^1 \\ h^2 \\ h^3 \end{pmatrix} = 0 \tag{5.4}$$

which can be expressed as:

$$Ah = 0 \tag{5.5}$$

We are interested in finding a non-trivial solution for *h* that minimizes the error vector $\varepsilon = Ah$. The error vector ε is also given by:

$$\varepsilon = \sum_i \varepsilon_i \tag{5.6}$$

With ε_i each of the single partial errors from each pair of correspondences and homography *H*. The vector ε_i is called the *algebraic error* and its norm is the *algebraic distance*:

$$d_{a\lg}(x_i^{'}, Hx_i)^2 = \left\|\varepsilon_i\right\|^2 = \left\| \begin{bmatrix} 0^T & -w_i^{'}x_i^T & v_i^{'}x_i^T \\ w_i^{'}x_i^T & 0^T & -u_i^{'}x_i^T \end{bmatrix} \begin{pmatrix} h^1 \\ h^2 \end{pmatrix} \right\|^2 \tag{5.7}$$

The advantage of the use of the algebraic distance is that it results in a linear solution to the problem and therefore, lower computational cost. The disadvantage is that it does not have any geometric meaning and for the case of an affine transformation, the algebraic and geometric distance are the same [54].

5.2.2 Geometric distance

The objective is finding the homography $\hat{H}$ that minimises the Euclidean distance *d(·)* between measured (*x*) and estimated ($\hat{x}$) locations of correspondences. The errors can be computed in three different ways, depending on the degree of accuracy or objectivity desired. These are the instances in ascending order:

Error in one image. The measurements in the first image are considered with null error (or true value, $\bar{x}$). Therefore, the estimated image coordinates are $\hat{x}_i = H\bar{x}_i$. The error function to minimise the geometric distance is the following square of differences:

$$\varepsilon^2 = \sum_i d\left(x_i^{'}, H\bar{x}_i\right)^2 \tag{5.8}$$

Symmetric transfer error. In most applications it is more sensible to consider that the errors occur in both images. Taking into account the backward transformation (H^{-1}), the function to minimise is given by:

$$\varepsilon^2 = \sum_i \left(d\left(x_i, H^{-1}x_i^{'}\right)^2 + d\left(x_i^{'}, Hx_i\right)^2 \right) \tag{5.9}$$

Reprojection error. The correspondences in both images are adjusted in order to minimise the error. That entails the computation of the estimated true correspondences ($\hat{x}_i$ and $\hat{x}_i^{'}$, notice that $\hat{x}_i^{'}$ is not needed since $\hat{x}_i^{'} = \hat{H}\hat{x}_i$) by means of the maximum likelihood estimation of the correspondences and the homography, as will be explained in section *5.2.3*. The cost function for the reprojection error is:

$$\varepsilon^2 = \sum_i \left(d\left(x_i, \hat{x}_i\right)^2 + d\left(x_i^{'}, \hat{x}_i^{'}\right)^2 \right) \tag{5.10}$$

Contrary to the error in one image and the symmetric transfer error, the reprojection error adds the *2n* parameters of the *n* correspondences to the parameters of the transformation *H* that are needed to optimise the cost function. The Sampson error [94] reduces the parameter space of the reprojection error to the parameters of *H*.

5.2.3 Statistical error

Probabilistic model

With absence of outliers, it can be assumed that the correspondences are affected by noise that follows a Gaussian probability distribution with zero mean and variance σ^2. Hence, the probability density function of each measurement x_i is given by:

$$P(x_i) = \left(\frac{1}{2\pi\sigma^2}\right) e^{\frac{-d(x_i,\bar{x}_i)^2}{2\sigma^2}} \tag{5.11}$$

For the case of error in both images, the probability of obtaining the set of measurements x and x' given the true homography H and measurements $\bar{x}$ is:

$$P(\{x, x'\} | H, \bar{x}) = \prod_i \left(\frac{1}{2\pi\sigma^2}\right) e^{-\frac{d(x_i,\bar{x}_i)^2 + d(x_i',H\bar{x}_i)^2}{2\sigma^2}} \tag{5.12}$$

And the log-likelihood is of the form:

$$\log P(\{x, x'\} | H, \bar{x}) = -\frac{1}{2\sigma^2} \sum_i d(x_i, \bar{x}_i)^2 + d(x_i', H\bar{x}_i)^2 + \text{constant} \tag{5.13}$$

and minimises the error function

$$\varepsilon^2 = \sum_i d(x_i, \bar{x}_i)^2 + d(x_i', H\bar{x}_i)^2 \tag{5.14}$$

The true values $\bar{x}_i$ and $H\bar{x}_i$ in the equations above must be estimated ($\hat{x}_i$ and $\hat{x}_i'$) by means of a Maximum Likelihood Estimate (MLE) of the true correspondences.

If we assume now that the errors are not only Gaussian, but there exist outliers, the error distribution can be modelled as a mixture distribution of a Gaussian and a uniform distribution [111]:

$$P(\varepsilon) = \prod_i \left(\gamma \left(\frac{1}{2\pi\sigma^2}\right) e^{-\frac{\varepsilon^2}{2\sigma^2}} + (1-\gamma)\frac{1}{\upsilon} \right) \tag{5.15}$$

where γ is a mixing parameter indicating the expected proportion of inliers and υ a constant providing some knowledge about the distribution of mismatches.

Equation (5.16) yields the negative log-likelihood for the mixture model:

$$-\mathrm{L} = -\sum_i \log\left(\gamma \frac{1}{2\pi\sigma^2} e^{\frac{\varepsilon^2}{2\sigma^2}+(1-\gamma)\frac{1}{v}}\right) \tag{5.16}$$

The maximisation of L minimises the error function in equation *5.14*.

Maximum Likelihood estimation of true correspondences

The Maximum Likelihood of true correspondences in both images ($\hat{x}$ and $\hat{x}'$) can be obtained from the measured correspondences (*x* and *x'*) and the homography (*H*) or fundamental matrix (*F*) consistent with these correspondences under the assumption that the errors follow only a Gaussian distribution. We will restrict to our more practical case of computing the fundamental matrix. The two measurements (x_i and x'_i) and the fundamental matrix (*F*) define via triangulation a hyperplane that passes through both correspondences and the two camera centres. The intersection of the beams passing through each camera centre and respective image correspondence provides the location of the point X_i in the 3D-space, whenever X_i does not lie over the baseline linking the two camera centres (epipolar geometry, Appendix A).

Therefore, the requirements of the true correspondences are twofold: they should satisfy the epipolar constraint $\hat{x}'F\hat{x}^T = 0$ and they should minimise the sum of squared differences in equation (5.10). The geometrical interpretation is straightforward, the function to be minimised is the distance between the measurements and the true correspondences lying over the epipolar lines. Thus, the solution is reduced to finding the closest distance from a point to a line.

Expectation Maximization

The Expectation Maximization (EM) algorithm [25] yields maximum likelihood estimates of parameters of models with missing data, *i.e.* there exist the (complete) data space X with observed variables X and the (incomplete) data space Y with variables Y that can only be observed indirectly through X.

The EM algorithm consists of two basic steps: the Expectation (E-) step computes the expectation of the maximum likelihood values of the complete data ($\overline{X}$) given only the

incomplete data (Y) and the current parameter values of the distribution ($\Phi^{(p)}$). Note that if all the variables could be directly observed, the log-likelihood of the complete data would solve the problem. However, it does exist an incomplete data space Y - herein the problem! The Maximization (M-) step exploits the maximum likelihood estimate of the complete data ($\overline{X}$, from the E-step) to compute the log-likelihood of the complete data, whose maximization updates the values of the parameters of the distri bution ($\Phi^{(p+1)}$). The algorithm needs an initial estimate of the incomplete variables and, after, both steps iterate until the algorithm converges. The choice of the initial estimate, the sort of distributions that models the data and the size of the parameter space will affect both the accuracy and time of convergence of the algorithm.

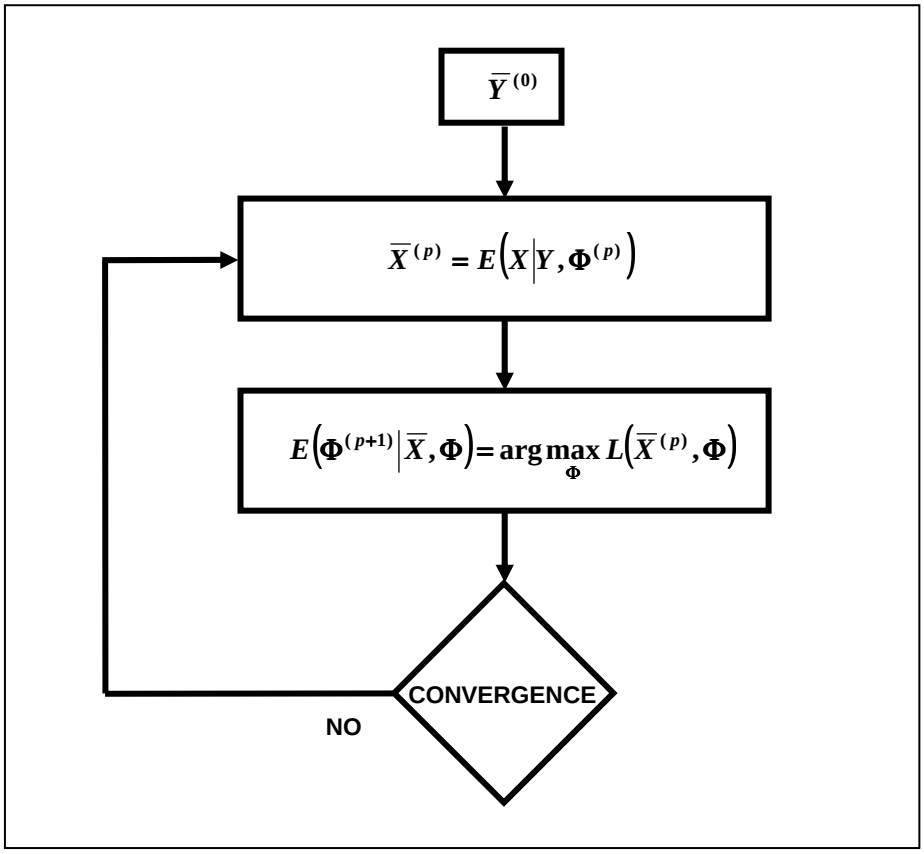


Figure 5.1. The EM algorithm.

5.2.4 Minimisation

We could dedicate a whole section about minimisation of cost function. However that is beyond our scope. We will only mention the most commonly used methods for iterative optimisation over the parameters of a function [40].

Direct search methods do not rely on the computation of the gradient of the function to minimise. Therefore, they are used when the cost function cannot be differentiated since their performance is not the most desirable. Examples are the downhill simplex and the amoeba method. Gradient-based methods can be of first (gradient descent) or second order (Gauss-Newton). The former does not usually present a good convergence while the latter depends on the approximation of a Taylor polynomial to the searching surface. Least-squares methods minimise the sum of squared residuals. Gradient based-methods can be used in the minimisation. For first order gradient descent the Jacobian is used and for Gauss-Newton the Hessian. An intermediate approach is the Levenberg-Marquardt algorithm [42]. It is considered as the best optimisation method for least-squares approaches. Levenberg-Marquardt alternates Gradient Descent and Gauss-Newton depending on the trade-off between the speed of convergence and reliability: it uses the Gauss-Newton when the Hessian is robust enough to converge fast to the minimum but uses gradient descent when Gauss-Newton finds troubles to converge. We use the Levenberg-Marquardt algorithm as a non-linear minimiser of our cost function within the Gold Standard algorithm (see next section) for scenes we assume that the projection is perspective.

5.3 The Gold Standard algorithm

The Gold Standard algorithm serves as a reference of excellence for other algorithms in the minimisation of the maximum likelihood cost function. The algorithm varies depending on whether the application consists of estimating the homography, fundamental matrix or also affine fundamental matrix. Let us explain the procedure of the Gold Standard algorithm for the maximum likelihood estimate of the fundamental matrix (thus, through minimisation of the geometric error distance in equation *5.10*). The information available is the set of correspondences (x and x'), whose error can be modelled by a normal distribution. An initial fundamental matrix ($\hat{F}$) can be estimated

from these correspondences by using the normalized *8*-point algorithm. From $\hat{F}$, and up to a projective transformation, the *3x4* camera matrix (*P'*) of the right image can be determined (providing the camera matrix (*P*) of the left image, which does not need to be computed, only set as a *3x3* identity matrix and a null 3-vector, and be consistent thus with $\hat{F}$ and *P'*). By triangulation the 3D point X_i is computed from the measured correspondences and the estimated fundamental matrix. That 3D point is reprojected to the image plane by the two camera matrices producing the maximum likelihood estimates $\hat{x}$ and $\hat{x}'$. The geometric error distance is minimised by a non-linear method (Levenberg-Marquardt) that corrects the *n* 3D-points and the parameters of the right hand-side camera.

The number of parameters of the cost function is thus *3n+12*, *i.e.* the number of 3D points by the *3* dimensions plus the *12* parameters of the right camera matrix. Despite the fact that the parameters of one of the cameras do not need to be adjusted and that the projection cameras could be defined up to scale (thus dropping one degree of freedom[3]), the complete parameter space in the minimization is still large and implies a significant computational cost.

The Gold Standard algorithm for affine geometry is much simpler. It is reduced to a linear minimisation of a cost function which is the sum of distances from sets of correspondences to the hyperplane that would fit them according to the affine fundamental geometry ($\hat{x}_i'^T F_A \hat{x}_i = 0$), where the hyperplane is f=(a,b,c,d,e)^T (defined by the fundamental matrix, see section *5.6*). The function is linearly minimised so as to force the hyperplane to pass through the centroid of the points. Then in order to minimise the distance from the points to the hyperplane the cost function is minimised in terms of the normal to the plane. This last step is solved easily by SVD.

5.4 Robust estimation

The mismatches existing within the set of correspondences will degenerate the calculated transformation that maps both images. We present the traditional robust

[3] A minimal parameterisation is not recommended since it hardens the minimisation surface [54].

algorithms that deal with big proportions of outliers (>50%) within the putative correspondence set.

5.4.1 RANSAC

The RANSAC (RANdom Sample Consensus) algorithm [31] is an iterative algorithm that randomly selects subsets of samples, models the parameters of the projectivity for that subset and computes a disparity measure over the complete set of samples. If the number of samples, which overall disparity to the model is smaller than a distance measure *t*, is larger than a predefined threshold *T* or the maximum number of iterations *N* is reached, the algorithm stops. Otherwise, it starts steps again selecting a new set of random samples. The algorithm discards the subsets containing outliers, since a wrong model will score poorly with respect to the threshold *T*. Therefore, it basically consists of a draw of hypothesis and consequent verification.

The disparity measure of RANSAC permits the definition of a set of inliers, which is the set of correspondences that approve the *consensus* threshold *t* for each iteration:

$$\rho\left(\varepsilon^2\right)=\begin{cases}1 & \varepsilon^2 < t^2 \\ 0 & \varepsilon^2 \geq t^2\end{cases} \tag{5.17}$$

The threshold *t* is set by considering the distribution of inliers, assuming a normal distribution of the location error. The distance error is therefore the result of sums of squared Gaussian errors, which results in a χ^2 distribution. The probability that this error is lower than a certain threshold leads us to model the threshold *t* with a cumulative chi-squared distribution.

The definition of the minimum number of inliers to accept a subset, *T*, is a cautious estimate of the number of inliers. The set with largest number of inliers is stored and the parameters of the model are estimated from that set.

The maximum number of iterations *N* is set to have a probability *p* (typically *0.99*) that at least one of the randomly selected samples does not contain any outlier.

$$N = \frac{log(1 - p)}{log\left(1 - (1 - \varepsilon)^s\right)} \tag{5.18}$$

being s the number of samples drawn every time and ε the proportion of outliers. N is usually adapted iteratively, *i.e.* when a subsample which contains a lower proportion of outliers than the previous estimate is found (this corresponds to a higher γ that gives rise to a higher L in equation (5.16).

The performance of RANSAC is vulnerable to a non-appropriate selection of the threshold t. If the threshold is too high wrong samples will be accepted and all the inliers (true inliers plus false positives) will contribute with the same weight; whereas when the threshold is too low the support may not be sufficient for a good modelling.

5.4.2 MLESAC

The MLESAC (Maximum Likelihood Estimate Sample Consensus) algorithm [111] is a variation of RANSAC that improves the performance by choosing a more robust cost function. Instead of considering the number of inliers, a maximum likelihood is preferred.

Another advance with respect to RANSAC is the weighted contribution of the samples. If an error is below the threshold, the error contribution of that inlier is the error itself. Whereas if the error is above the threshold, the contribution of the error of that outlier is weighted by the threshold:

$$\rho\left(\varepsilon^2\right) = \begin{cases} \varepsilon^2 & \varepsilon^2 < t^2 \\ t^2 & \varepsilon^2 \geq t^2 \end{cases} \tag{5.19}$$

The summation of all ρ*'s* is the cost function to minimise. The value of t is also selected to assure with a *95%* of probability that an inlier with an error location following a normal distribution is not rejected, *i.e.* $t=1.96\sigma$.

The negative log-likelihood presented in equation (5.16) is minimised. The trouble is that we do not know the value of the mixing parameter γ, which is an estimation of the proportion of inliers in the distribution. This is the problem of estimating parameters of

a model where there is missing data, and the approach to solve it is Expectation Maximization. The initial value of γ is chosen as an estimate of the samples are inliers (estimate of inliers for ground truth experiments in Chapter *4*). The E-step of the algorithm defines that the probability that a sample η_i is an inlier given the expected proportion of inliers is:

$$P(\eta_i = 1|\gamma) = \frac{p_i}{p_i + p_o} \tag{5.20}$$

With p_i the likelihood that a sample is an inlier given that is an inlier and p_o the likelihood that a sample is an outlier given that is an outlier. The denominator in *5.20* represents the error in the sample space, *i.e.* the mixture distribution of a Gaussian and uniform distributions as shown in equation *5.15* for a single sample *i*.

The M-step consists of a new estimation of the γ from the estimation in equation (5.20):

$$\gamma = \frac{1}{n}\sum_i P(\eta_i = 1|\gamma) \tag{5.21}$$

The algorithm iterates until convergence, generating the set of inliers. This set of inliers produce an initial estimate of the transformation that maps both images. That initial estimate is optimised by minimising the cost function over that initial estimate and the whole sample space.

5.5 Affine epipolar geometry

The epipolar geometry for perspective cameras is presented in Chapter *6*. Here we introduce the basic expressions we need for our computations. When the scenario can be approximated by affine cameras the algorithms are less complicated due to linearity. The centres of affine cameras are at infinity and the projection from 3D to 2D is parallel. Therefore, the epipolar lines are parallel since by definition all epipolar lines meet at the epipole, and this is at infinity.

The affine fundamental matrix has the form:

$$F_A = \begin{bmatrix} 0 & 0 & a \\ 0 & 0 & b \\ c & d & e \end{bmatrix} \tag{5.22}$$

As the matrix has five non-zero entries, it has four degrees of freedom: one for each epipole and two for the affinity between the pencil of epipolar lines in each view. The epipolar lines have the expressions:

$$l' = F_A x = (a, b, cx + dy + e)^T \tag{5.23}$$

$$l = F_A^T x = (c, d, ax + by + e)^T \tag{5.24}$$

And the epipoles are:

$$e = (-d, c, 0)^T \tag{5.25}$$

$$e' = (-b, a, 0)^T \tag{5.26}$$

5.6 Automated solution to the correspondence problem

Input descriptor. Our descriptor stems from the grouping of pair of points of interest. That pairing is a significant advantage when running iterative algorithms in the RANSAC's family. Recall that the number of iterations to guarantee with a probability *p* that a subset of samples is free of outliers was a logarithmic expression (equation (5.18)). As a consequence of the pairing we are already reducing the number of samples *s* to a half. For example, for the case of affine cameras approximation we require four samples for a minimal solution. By selecting two descriptors in each image we already have the four correspondences needed to calculate the affine fundamental matrix mapping both images. But the parameter *s* in equation (5.18) will have a value of *2* rather than *4*. That is due to the fact that if two descriptors correspond (a pair of corresponding points in each descriptor that are not coplanar with a pair of points of a second descriptor in their respective images), we are assuring at once that a set of two points in one image have their correspondence at the endpoints of the counterpart descriptor in the other image. That is advantageous in the sense of an improvement of speed of processing: the system will require a smaller number of iterations as we can see in figure *5.2*. But even more interesting, it is a very-welcome enhancement in terms of the proportion of outliers that the new layout can cope with. Figure *5.3* shows the

relation between the number of inliers inside a distribution that a RANSAC algorithm can deal with after *N* iterations for *s=2* and *s=4*. We can see in the plot that the cost of dealing with a proportion of around *70%* of inliers (*30%* of outliers) when selecting *4* samples is equivalent to dealing with a distribution of *50%* of outliers when we only have *2* samples to select. The gain is even more advantageous for a greater proportion of outliers. Notice that the cost of dealing with a proportion of almost *70%* of outliers for *s=4* is the same as dealing with a proportion of outliers of 90% for *s=2*. That proves that the pairing of data points with our descriptor is especially more powerful when larger proportions of mismatches exist, which means that the algorithm can cope with more corrupted datasets for the same computational cost.

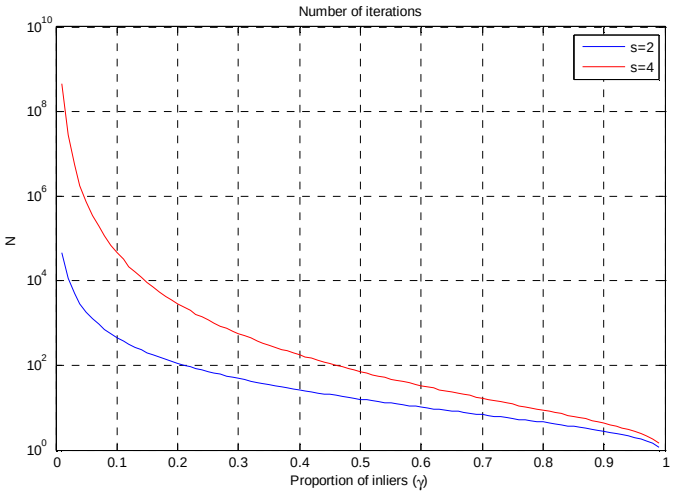


Figure 5.2. Number of iterations as a function of the proportion of inliers and number of samples.

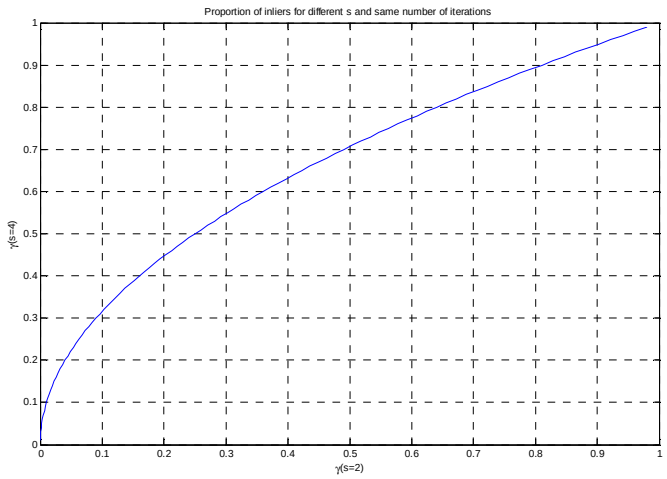


Figure 5.3. Proportion of inliers for different number of samples and fixed number of iterations.

Algorithm. Now we explain how the algorithm proceeds (see pseudo-code at the end). We subdivide the whole feature space S into the two input sub-spaces S_1 and S_2. The criterion is that S_1 encloses the *50%* of the descriptors that received the highest similarity scores and S_2 the rest. We will refer only to S_1 whenever we mention the sample space, until a new clarification arises. We first tile the image with the purpose of selecting samples (endpoints of descriptors) homogeneously distributed over the whole image - that aims at a proper estimation of the fundamental matrix. We divide the image into nine quadrants and randomly select features descriptors with the restrictions that no more than two samples can be extracted from the same quadrant and if more than one sample belongs to the same quadrant the descriptor is only accepted if the quadrant contains at least *20%* of the whole number of samples. Otherwise the descriptor is withdrawn and another one is randomly selected. Same applies if the four selected points are coplanar or three of them are collinear, since that would lead to a degenerate solution for the affine fundamental matrix.

The character of our features sets up four different combinations of correspondence of samples. Let us assume that the two selected features in the first image are f_{AB} and f_{CD}, being the sub-index the endpoints that delimit the feature. Their respective putative correspondences in the other image are $f_{A'B'}$ and $f_{C'D'}$. Therefore, the four combinations of matching are *{AA',BB',CC',DD'}*, *{AB',BA',CC',DD'}*, *{AA',BB',CD',DC'}* and *{AB',BA',CD',DC'}*. For each of these four possibilities we compute the affine minimal solution of the fundamental matrix, F_a. There can be up to three real solutions consistent with the data points and all cases should be examined. At this point we test that the epipolar lines do not overlap within a minimum distance threshold. If so, the samples are rejected and another set is chosen to avoid possible false positives since both samples would be represented by identical epipolar lines in the other image. Next we calculate the MLE of the true correspondences that minimises the algebraic error distance according to the measured correspondences and the affine epipolar geometry defined by F_a. That ML is calculated over the whole sample space, S_1. As the correspondences are arranged in pairs, we find again the dichotomy of finding which is the true point correspondence. For example, points G and H from descriptor f_{GH} in one image and G' and H' from $f_{G'H'}$ in the other image would give rise to the following two functions to minimise: $\left(G-\hat{G}\right)^2+\left(H-\hat{H}\right)^2+\left(G'-\hat{G}'\right)^2+\left(H'-\hat{H}'\right)^2$ or $\left(G-\hat{H}\right)^2+\left(H-\hat{G}\right)^2+\left(G'-\hat{H}'\right)^2+\left(H'-\hat{G}'\right)^2$, representing $\hat{\cdot}$ the maximum likelihood value or

true correspondence of that sample. Therefore, the function we need to work with is the one that minimises the distance error among the correct true correspondences, *i.e.:* $min\left(\left(G-\hat{G}\right)^2+\left(H-\hat{H}\right)^2,\left(G-\hat{H}\right)^2+\left(H-\hat{G}\right)^2\right)+min\left(\left(G'-\hat{G}'\right)^2+\left(H'-\hat{H}'\right)^2,\left(G'-\hat{H}'\right)^2+\left(H'-\hat{G}'\right)^2\right)$. The convergence by Expectation Maximization is implemented as explained in section *5.2.3*. The distance error is plugged into equation (5.15) to compute the maximum likelihood estimation of the proportion of inliers, γ. After convergence, the error distance and γ give the negative log-likelihood -L. If that –L is lower than the previous existing estimate, the set of inliers, their fundamental matrix and their errors are stored. Finally, the number of iterations *N* of the algorithm is adapted by equation (5.18) and the algorithm iterates again selecting a new set of samples. The process is repeated until the adapted maximum number of iteration is reached.

After that, another iterative procedure starts until the number of inliers obeys the minimised estimation of the fundamental matrix. We deem as inliers these samples for which the error distance is below the threshold *T*. We differ with equation *5.19* in the sense that the non-inliers, *i.e.* error equal or bigger than *T*, are not included in the minimisation process. We also sieve inliers that produce multiple matches, *i.e.* one sample has got within its vicinity more than one epipolar line, only the one with lowest error is kept. With the set of inliers we determine the Maximum Likelihood estimate of the fundamental matrix by using the Gold Standard algorithm for affine epipolar geometry. Affinities imply linearity and that eases the calculus, basically a simple SVD provides the affine fundamental matrix from the correspondences. For the case that the images we are working with have perspective effects, the process of finding the true correspondences consistent with the epipolar geometry gets more complicated as explained in section *5.3*. We perform the non-linear minimisation with the Levenberg-Marquardt algorithm. Next, with the ML estimation of the fundamental matrix we define epipolar lines and search for further correspondences within the remaining whole set of putative correspondences, *i.e.* $S_1 + S_2 - S_{inliers}$. We find the maximum likelihood of the true correspondences and calculate the error for each datum. The algorithm checks whether the number of inliers is stable and if not iterates again including new inliers, as the new correspondences that accomplish the condition that their error distance is below the threshold *T*.

Procedure: Robust estimation of the fundamental matrix

In:

- Putative correspondences from endpoints of affine invariant descriptors

Out:

- ML estimate of the fundamental matrix
- Correspondences (set of inliers)

Algorithm:

1. Tile both images for homogeneous extraction of samples
2. Repeat for N subsets of samples:
    - Select a random number of *n* correspondences
    - Check collinearity and coplanarity constraints
        - If violated, select random correspondences again
    - Four each case *{AA',BB',CC',DD'}*, *{AB',BA',CC',DD'}*, *{AA',BB',CD',DC'}* and *{AB',BA',CD',DC'}:*
        - Compute fundamental matrix
        - There can exist up to three solutions
            - ML of true correspondences over S_1
            - Calculate error for each correspondence
            - Estimate expected proportion of inliers γ
                - Until γ converges:
                    - Compute $P(\eta_i = 1|\gamma) = \frac{p_i}{p_i + p_o}$
                    - New estimation of γ from $P(\eta_i = 1|\gamma)$
            - If $\gamma > \gamma_{best}$, store parameters
        - Compute negative log-likelihood -L
        - Store best inliers, errors and fundamental matrix when $-L<-L_{best}$
    - Adapt number of iterations *N*
3. Until the number of inliers converge
    - Store all (new) correspondences deemed as inliers from S_1
        - Threshold constrain: $\rho(\varepsilon^2) = \begin{cases} \varepsilon^2 & \varepsilon^2 < t^2 \\ 0 & \varepsilon^2 \geq t^2 \end{cases}$
        - Multiple matches constrain – keep minimum error to epipolar line
    - Estimate fundamental matrix from set of inliers

- If affine, Gold Standard affine
- If perspective, Gold Standard in section 5.3

- Find further correspondences in $S_1 + S_2 - S_{inliers}$ over a strip around epipolar lines
- ML estimate of the (new) true correspondences
- Calculate error for each (new) correspondence
- Update $S_1 = S - S_{inliers}$, with $S = S_1 + S_2$

4. END

5.7 Experimental results

We show final results for the robust estimation of correspondences from the invariant descriptors between the images and their homographies in Chapter 4. Figures 5.4 to 5.6 show the recall, precision and number of regions extracted. By referring to table 4.4, we can see that the system encountered more difficulties for the instances where the transformation combined changes of scale and shear. The system should have been strong to these affinities. However the aforementioned propagated errors in the affine frames face the evidence of lower performance for strong affine changes.

We also show an example of the performance of the algorithm over the countryside scene under an affinity and changes in the illumination. The location of the 44 ground truth correspondences has been added a Gaussian noise of standard deviation 1. Figures 5.7 and 5.8 show the correspondences and the epipolar lines. Notice that epipolar lines nearby can be a source of mismatches when searching for correspondences within an epipolar line strip, since their correspondences have other epipolar lines in the proximity. Figure 5.9 shows a successful matching of correspondences.

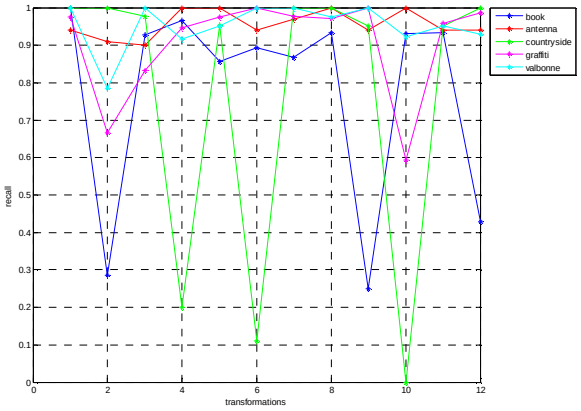


Figure 5.4 Recall

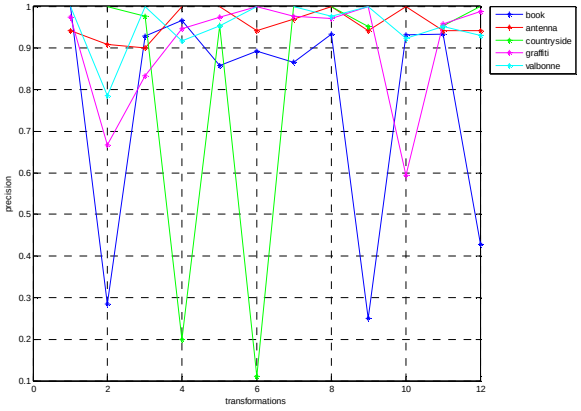


Figure 5.5 Precision

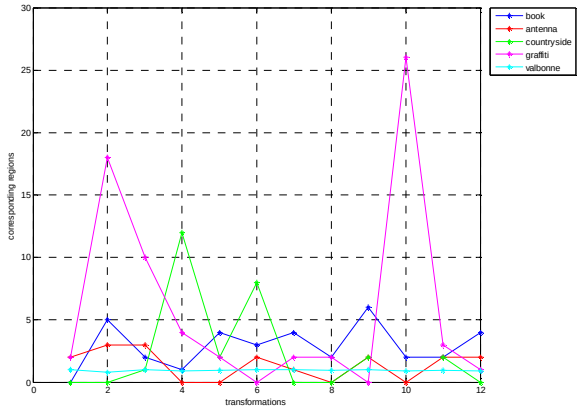


Figure 5.6 Number of corresponding regions

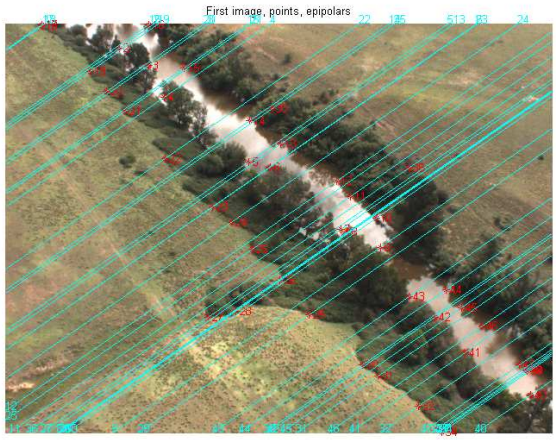


Figure 5.7 Correspondences and epipolar lines in the original image.

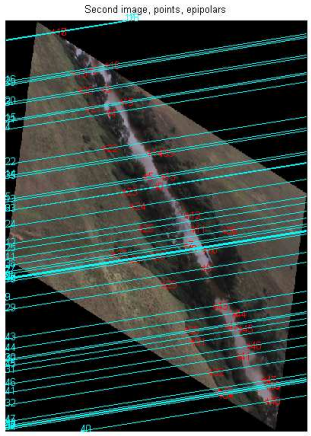


Figure 5.8 Correspondences and epipolar lines in the transformed image.

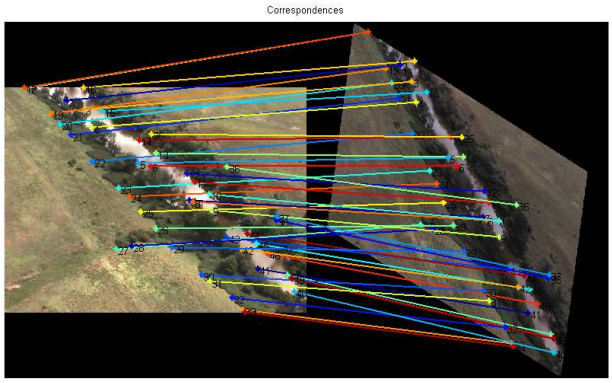


Figure 5.9 Matching of correspondences.

γ_0	# matches		Success (%)		γ_{best}		Iterations	
	avg	std	avg	std	avg	std	Avg	Std
0.1	8.400	2.998	13.49	8.696	0.250	0.024	61.74	14.82
0.15	14.300	3.457	20.06	24.849	0.259	0.022	43.45	18.45
0.2	20.100	16.535	48.82	41.064	0.268	0.045	50.01	19.38
0.25	26.600	8.947	82.14	4.810	0.332	0.043	31.23	17.49
0.3	33.800	10.304	93.41	4.364	0.377	0.053	27.77	15.53
0.4	32.700	9.129	92.94	6.291	0.458	0.059	12.99	4.02
0.5	35.100	11.070	93.69	10.640	0.536	0.050	8.70	2.91
0.6	34.300	11.585	95.30	3.378	0.659	0.049	6.72	1.46
0.7	33.700	12.266	98.88	5.778	0.740	0.057	4.49	1.07
0.8	29.500	11.335	96.44	5.639	0.886	0.058	2.95	1.12
0.9	29.800	12.726	98.42	3.759	0.963	0.047	0.52	1.10

Table 5.1. Performance of MLESAC algorithm

Table 5.1 presents the results of the performance of the MLESAC algorithm over the same scene for different values of γ. That is, the set does not contain outliers but the initial estimation of our correct potential matches within the set, previous robust estimation, is γ. The algorithm was executed 1000 times for each γ. In the table, the number of matches is the number of correct (ground truth) matches found from the initial set of 44, 'success' is the percentage of true inliers found. γ_{best} is the maximum likelihood estimation of inliers obtained by the iterative process of MLESAC previous to the search of further correspondences by the optimization and strip about epipolar lines stages.

The result of using the system over a real image is shown in figures 5.10 and 5.11. We can see that the system is not able to find the correct whole set of correspondences. In figures 5.12 and 5.13 we perform a search of a minimum number of correspondences that permits the recovery of the fundamental matrix for a minimum solution. That is extracting only the set of best inliers in the iterative MLESAC algorithm – four points (two pairs of descriptors) – and not extraction of further correspondences over a strip distance from epipolar lines. The search of correspondences is not completely fulfilled but we can observe that the correspondence problem degenerates in the search of further correspondences when the fundamental matrix of the initial set of correspondences does not satisfy the transformation in between both images.

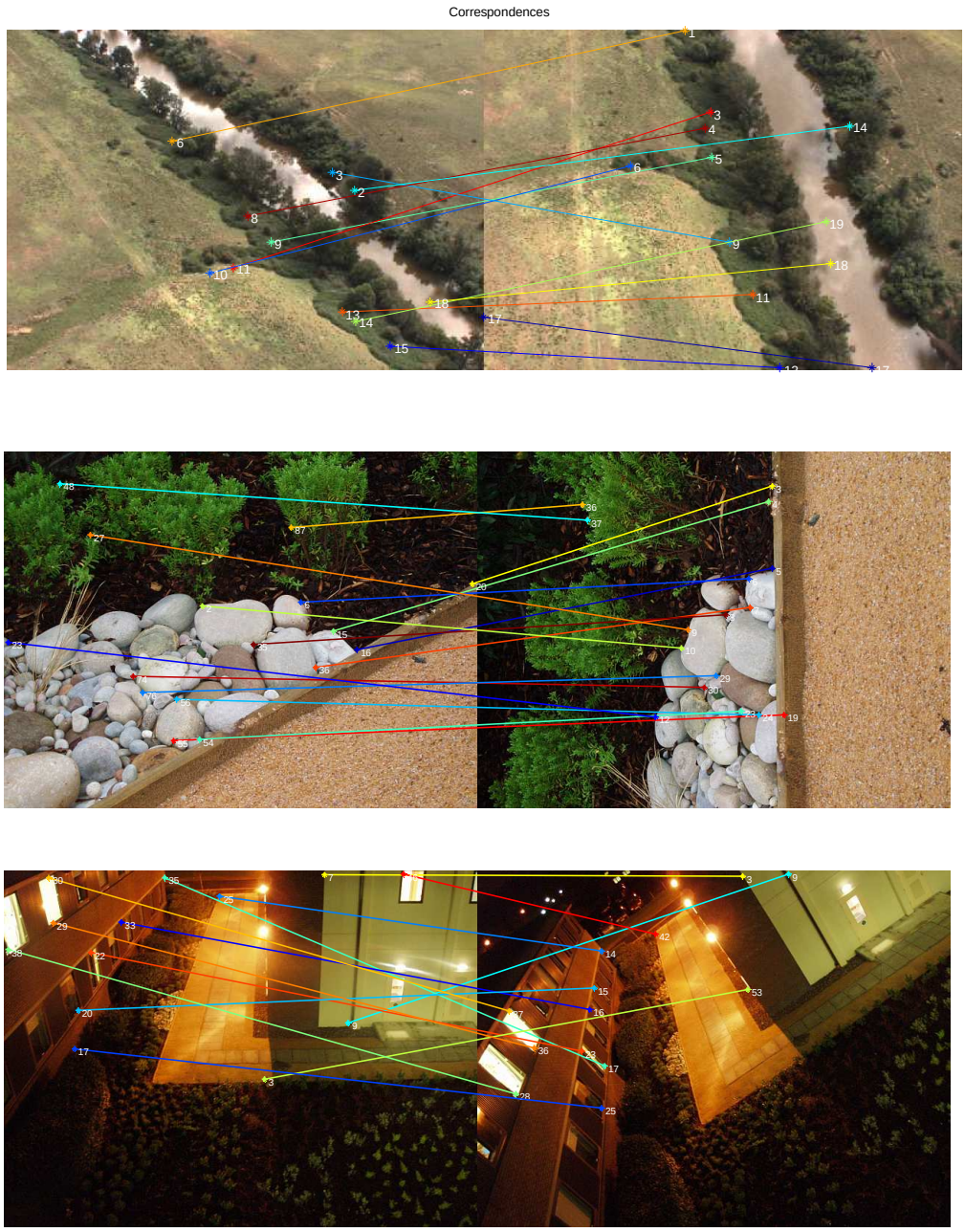


Figure 5.10. Matching of correspondences over real images.

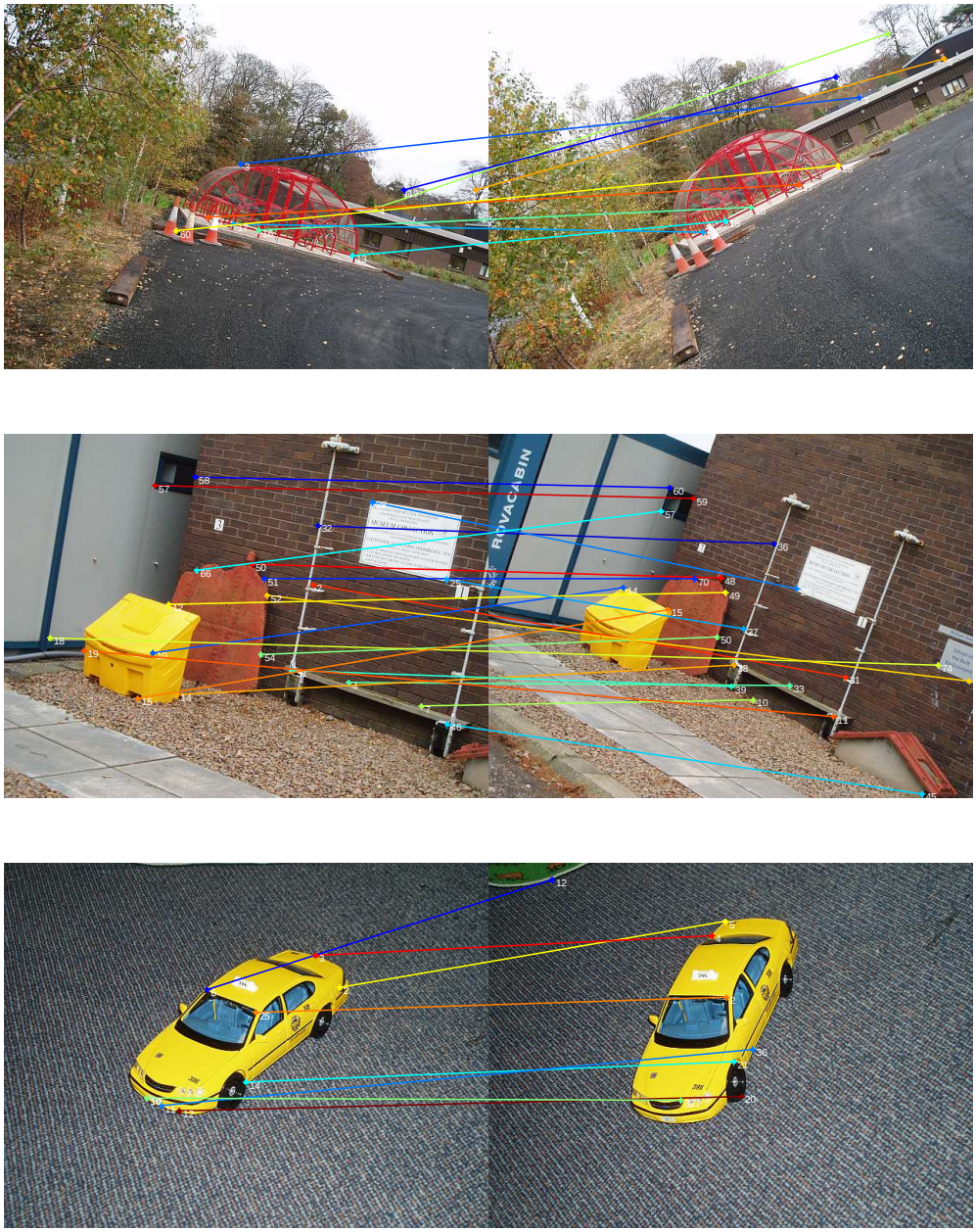


Figure 5.11. Matching of correspondences over real images.

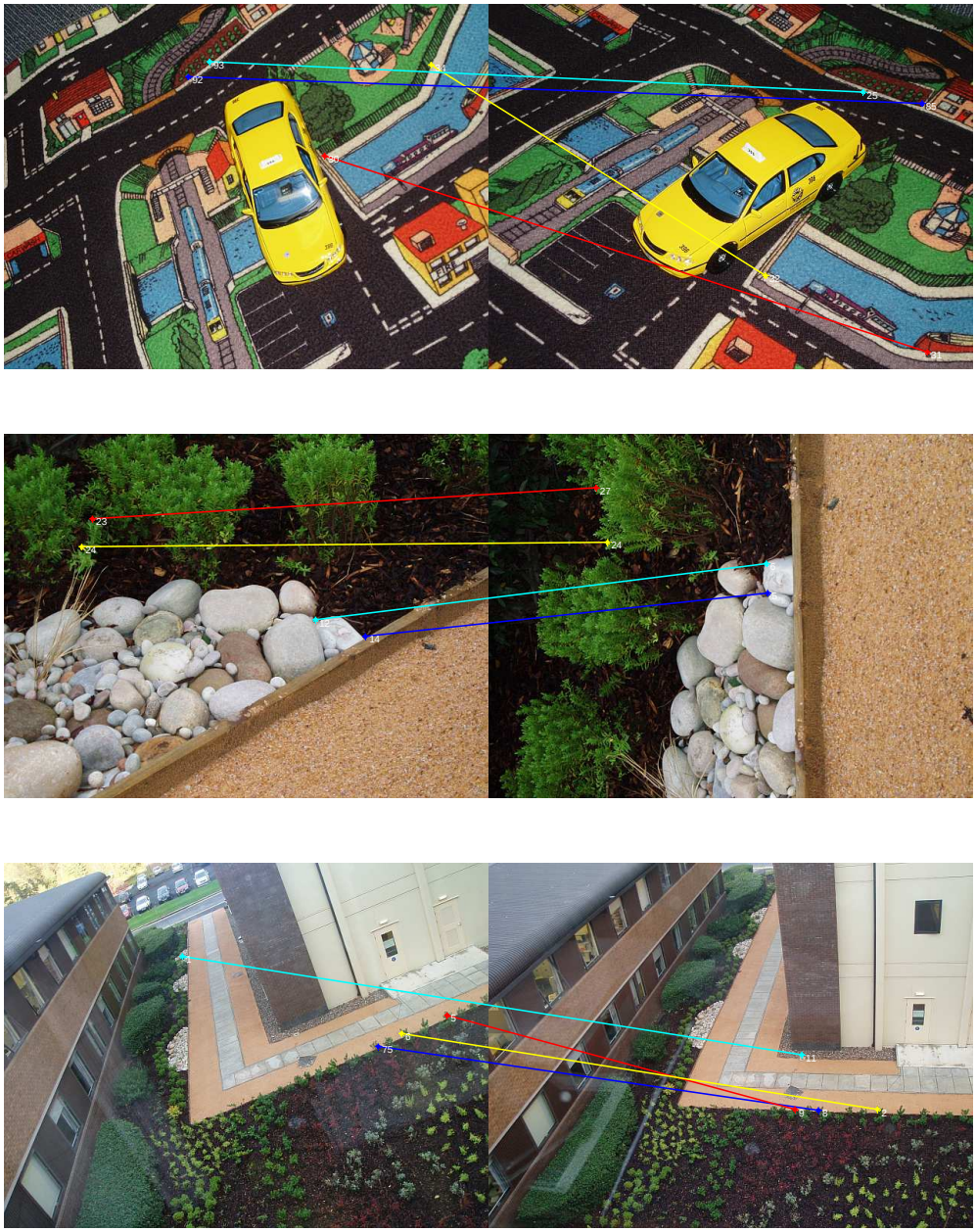


Figure 5.12. Matching of best set of inliers for a minimal solution.

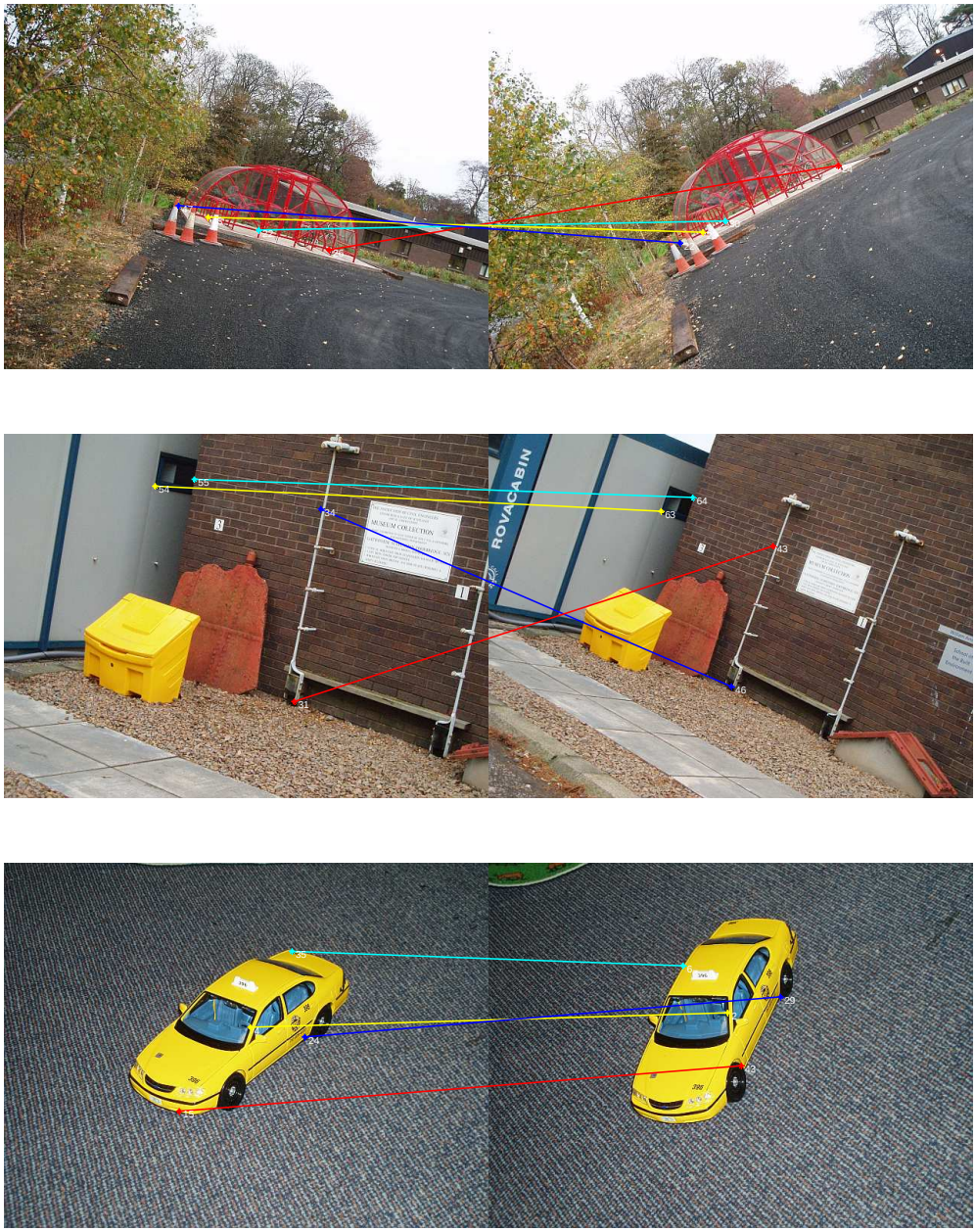


Figure 5.13. Matching of best set of inliers for a minimal solution.

5.8 Conclusions

We have implemented the MLSAC algorithm [111], which is able to deal with the high proportion of outliers in the sample set of correspondences. Part of the success of the MLESAC is the nature of the features that we input. Robust estimators are usually input with single point correspondences. Our features consist of pairs of points. If two descriptors correspond, that implies that there exist already two corresponding points. That permits faster convergence of the algorithm by reducing the number of iterations or being able to deal with higher proportions of outliers at the same computational cost. The experiments were performed with the synthetic data from the affine invariant descriptor in Chapter 4. When using real data, the robust estimator of the parameters of the fundamental matrix between the images was not able to find the correct correspondences. However, the system proved a satisfactory performance for affinely transformed images. That stems from the fact that the computation of the geometric descriptor relies on the calculus of derivatives as shown in previous chapter.

Chapter 6 – Conclusions

6.1 Discussion

We have proposed a method that combines an absolute affine geometric invariant with an affine invariant photometric descriptor based on moments developed by Mindru *et al.* [82]. The affine geometric invariant is based on the affine arc-length metric. Affine invariant parallelograms are extracted along contours. The principal difficulty of this kind of approach is that the affine arc-length is very dependent on the adequate extraction of the contours and particularly on a right detection of the endpoints of segmented contours. Contour maps are not always reliably extracted under change of view or illumination: they are sensitive to occlusion, partial detection and different labels at junctions. To ameliorate this, we implement two approaches. First, we perform perceptual contour grouping that improves the interconnections of the contour map. Second, we consider high-curvature points lying over contours that are robust to viewpoint and illumination changes. This permits segmentation of the contours into more reliable and bounded primitives from which we form the invariants. We organise the information in a graph structure: the nodes store the spatial information of the high-curvature points whereas the edges are the contour segments delimited by the high curvature points. We generate a descriptor for each pair of interconnected high-curvature points. Thus, the system can accommodate the affine arc-length based descriptor, and is robust to poorly defined contour detection, since all the possible combinatorics of interconnected high-curvature points and different labelling of contours are considered. However, the drawback is the inherent computational cost associated with a dense search space.

Experimental analyses have shown that the area defined by the affine invariant parallelogram is very susceptible to input errors. The affine arc-length requires the computation of the first and second order spatial derivatives along the contour. We approximate the contour with a least-square cubic spline approximation and compute the derivatives of the splines. That is preferred to other alternatives like computing finite differences, which are sensitive to noise. However, the small noise that stems from the approximation with splines propagates throughout the expressions that define the affine arc-length frame resulting in a considerable error at the output. Indeed, in our

tests even the computation of the affine invariant frame from one-to-one affinely transformed contours results in unsatisfactory performance. In our tests with ground truth, we compared the performance of the system when transforming the spatial coordinates of the contour with an affinity with the performance when applying the same transformation to the spline themselves (figure 4.30). In the former case, the error introduced by the splines is minimal but it is propagated, whereas for the latter there is no error propagation. We performed experiments with synthetic data where the contours were affinely transformed. The matching success of the algorithm was low, although further research has been undertaken in the improvement of the voting algorithm. In particular its substitution by an iterative Munkres algorithm. However, we have centred our efforts upon the descriptor itself rather than on the matching process. The low number of true correspondences found for synthetic images restricted the application of the system over real viewpoint scenes where the contours are independently extracted.

In Chapter 5 we have implemented a maximum likelihood estimate RANSAC algorithm, MLESAC [111], which is able to deal with the high proportion of outliers in the sample set of correspondences. Part of the success of the MLESAC is the nature of the features that we input. Robust estimators are usually input with single point correspondences. Our feature descriptors are based on pairs of points. So matching the descriptors between images implies that there exist two corresponding points. This permits faster convergence of the algorithm in comparison with single point-feature descriptors by reducing the number of iterations, or enabling the system to deal with a higher proportion of outliers at the same computational cost. We have proved that the algorithm is able to deal with percentages of outliers of around 90%. This is equivalent computationally to the cost of dealing with a proportion of outliers of less than 70% in the single-point feature case. These experiments were performed with the synthetic data from the affine invariant descriptor in Chapter 4. When using real data, the robust estimator of the parameters of the fundamental matrix between the images was not able to find correct correspondences due to errors in the extraction of the affine invariant frames.

6.2 Further work

The performance over real scenes of the geometric affine invariant frame, which is core to our application, is endemic to the propagation of errors. There is a need to investigate whether this propagation of error can be mitigated; or whether even if the parallelograms are not absolute invariant due to these errors, the descriptor can produce more reliable regions; or if not, we need to look at alternatives that are more robust.

Wide baseline matching has been applied almost exclusively to images of the same modality. The system can be expanded potentially to multi-modal applications using for instance, mutual information as a photometric descriptor. The incorporation of intensity and range data models of image formation can also be assessed for multi-modal registration and even fusion. Defining a collection of models, initially for visible and infrared imagery, with a different number of characteristic components may be helpful in constraining feature search as well as establishing complementary information and eliminate interpretation ambiguities between different modes. Therefore, changes in illumination, emissivities, reflectance, surface normal o depth would produce a different variation of intensities depending on the image mode. The approaches would be strongly dependent on the operational scenario, for instance the search for planar patches approximations is effective for aerial survey but totally inappropriate for long range IR or RF data.

Appendix A – Matching contours in image pairs using Fourier descriptors

A.1 Introduction

We study an alternative approach to solve the correspondence problem using corresponding contours in a pair of images. The basis for this study is the work by Wu and Sheu [127], which described a method to match closed contours in a pair of perspective images. As before, contour information is potentially more robust against changes in the photometry of the image, and the computational complexity may be reduced by limiting the dataset. Boundaries are higher level entities than corners, edges, etc. being able to conglomerate much more information that at the same time can be constrained by some metrics in order to have a better definition of the entity.

The method assumes a perspective projection and knowledge of the positions, orientations and focal length of the cameras. Hence, the fundamental matrix that relates a point in one image to an epipolar line in the other image is known. Contours on either image plane can be represented by Fourier series. Then Fourier descriptors are defined using the computed epipolar geometry to perform the matching based on a measure of similarity. This metric, termed the "spectral distance", between these two descriptions is a measure of the degree of matching between the contours. If this is maximised, then the contours are well matched. The authors compute an iterative procedure in the frequency domain where sets of slopes and intercepts of the epipolar lines corresponding to each contour are used as descriptors. Therefore, it is the difference between these two encodings that is minimized.

If the method assumes knowledge of the epipolar geometry then at first sight it is not a viable approach to match uncalibrated, wide-baseline images. There are two possibilities:

- The minimum spectral distance is used as a cost function in an optimization procedure. The transformation into the Fourier domain might be expected to make the computation more robust.

- It is used as a hypothesis verification step in an iterative algorithm. Hypotheses are generated by random sampling of point sets from extended contours, then the minimum spectral distance(s) of the whole contour(s) is (are) used to confirm the hypotheses in a robust manner.

A.2 Scene geometry

As depicted in figure *A.1*, the relative positions and orientations of the two cameras with respect to the world coordinate system are known, but the locations of the 3D points in the space are unknown. Three tri-dimensional and two bi-dimensional coordinate systems are involved. These are, respectively, the world coordinate system, the two coordinate systems of the two cameras and the pair of two-dimensional coordinate systems of their image projection planes. The second coordinate of the cameras' reference frame corresponds to the dimension of depth. Therefore, image planes u_i and w_i are parallel to the image planes u and w of the i^{th} camera coordinate system at a certain depth magnitude which is set by the focal length λ_i of each camera. L_1 and L_2 are the rays that go through the camera centres and project in *perspective* a 3D point of the world onto the image plane coordinates.

The displacement between the two cameras is:

$$\begin{bmatrix} x_{cd} \\ y_{cd} \\ z_{cd} \end{bmatrix} = \begin{bmatrix} x_{c1} - x_{c2} \\ y_{c1} - y_{c2} \\ z_{c1} - z_{c2} \end{bmatrix} \tag{A.1}$$

where x_{ci} y_{ci} z_{ci} are the coordinates of the camera centres in world coordinates. Therefore, a point into the coordinate system of one camera can be transformed into the coordinate system of the other camera by:

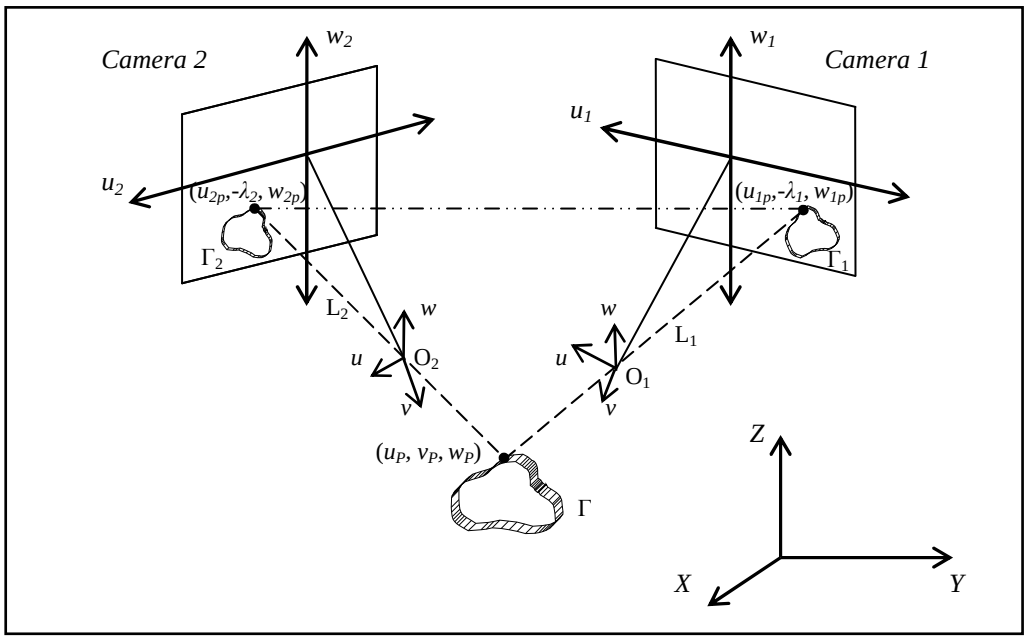


Figure A.1. The camera geometry.

$$\begin{bmatrix} u_1 \\ v_1 \\ w_1 \end{bmatrix} = R_t \begin{bmatrix} u_2 \\ v_2 \\ w_2 \end{bmatrix} + P_t \tag{A.2}$$

If R_1 and R_2 are the orientation of the each camera with respect to the world reference frame, then R_t and P_t are:

$$R_t = R_1 \cdot R_2^{-1} \tag{A.3}$$

$$P_t = \begin{bmatrix} J \\ K \\ L \end{bmatrix} = -R_1 \cdot \begin{bmatrix} x_{cd} \\ y_{cd} \\ z_{cd} \end{bmatrix} \tag{A.4}$$

The calculations can be reduced in complexity without loss of generality by simply assuming a canonical camera configuration. Hence, let us consider that the basis of all coordinates is the camera *1* coordinate system. Consequently, this camera will have its camera centre at the origin and look along its *v* axis. The resulting new coordinates for the camera centres O_i and two imaged points from object Γ, Q_1 and Q_2, are the following:

$$O_1 = (0,0,0) \qquad Q_1 = (u_{1p}, -\lambda_1, w_{1p})$$

$$\mathrm{O}_2 = (J,K,L) \qquad Q_2 = R_t \begin{bmatrix} u_{2p} \\ -\lambda_2 \\ w_{2p} \end{bmatrix} + \begin{bmatrix} J \\ K \\ L \end{bmatrix} \tag{A.5}$$

where the coefficients λ_i denote the focal length of each camera.

A.3 Epipolar Geometry

The epipolar geometry [130,54,36,109] limits the search of the position of points in one view of a single scene according to the position of their counterparts in the other view by means of epipolar lines which constraint where the points lie. The projective geometry between the two images is defined with the support of the internal parameters of the camera and the relative pose. It is independent of the scene structure.

The epipolar geometry is represented by a 3x3 matrix, the *essential matrix* when the internal parameters of the camera are available or the *fundamental matrix* when these are unknown.

Considering O_l and O_r the optical centres of two cameras and a point P in the 3D space; p_l and p_r are the images of P on the 2D image plane of each camera (figure *A.2*). The epipolar plane Π is defined by the point in the first image p_l and the two optical centres. The line which intersects Π with the plane of the second camera (π_r) is called the *epipolar line*. This constrains the location of the counterpart of p_l (p_r) to this line. Furthermore, for every point p_{lk} in the first image describing a plane Π_k, there exists a point e_r in the other image, called the *epipole*, which all the possible k epipolar lines pass through by. This is due to the line between the optical centres acts as a pencil for all epipolar lines and the epipole lies in the intersection of this joining line between O_l and O_r.

Assuming pinhole model:

$$s_l \tilde{p}_l = A_l [I\ 0] \begin{bmatrix} P \\ 1 \end{bmatrix} \qquad s_r \tilde{p}_r = A_r [R\ t] \begin{bmatrix} P \\ 1 \end{bmatrix} \tag{A.6}$$

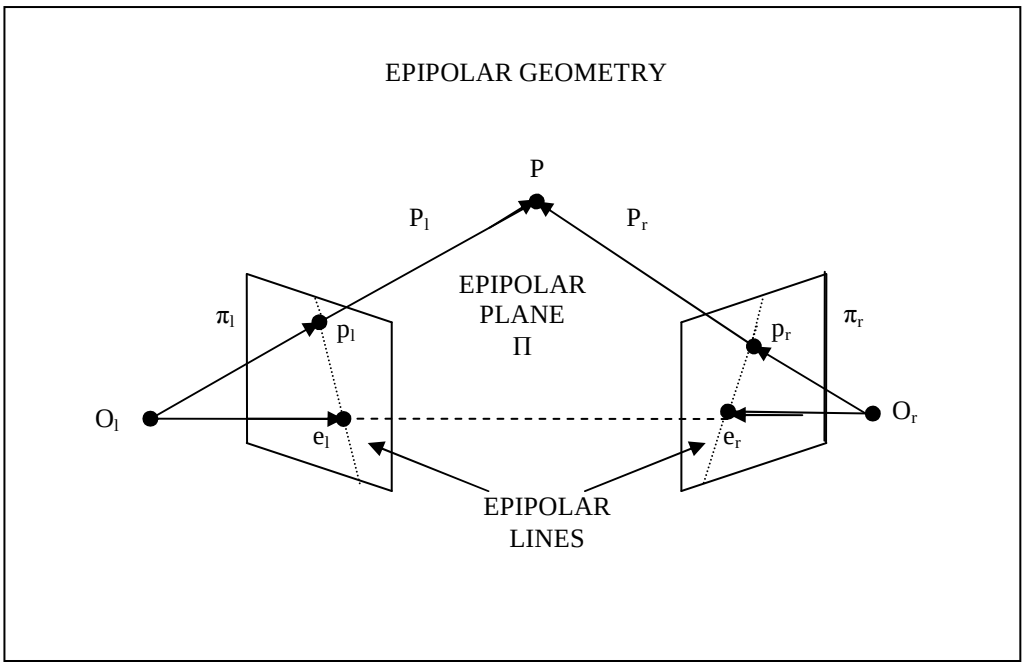


Figure A.2. The epipolar geometry.

with s_l and s_r arbitrary scales, A_l and A_r the intrinsic matrices of the cameras, I the identity matrix and R and t the rotation and translation of the second camera respect to the first. Cancelling s_l, s_r and P from equation *A.6*, gives the fundamental equation *A.7*, which says that the corresponding point in the right image lies on the epipolar line:

$$\tilde{p}_r^{\,T} A_r^{-T} T R A_l^{-1} \tilde{p}_l = 0 \qquad \text{(A.7)}$$

where T is an anti-symmetric matrix defined by t such $Tx = t \Lambda x$ for all 3D vector x, with Λ denoting the cross product.

From the previous equation, it can be extracted the expression of the *fundamental matrix* of the two images:

$$F = A_r^{-T} T R A_l^{-1} \qquad \text{(A.8)}$$

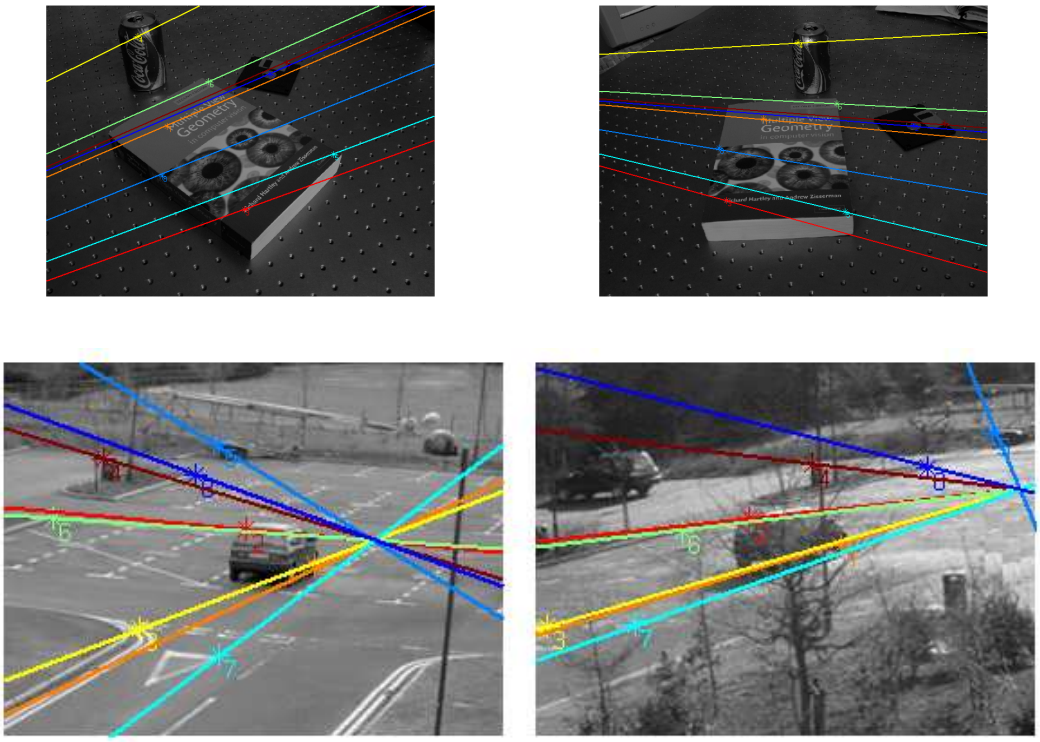


Figure A.3. Epipolar lines over indoor and outdoor wide-baseline scenes.

A.4 Slope- and intercept-based contour matching

Therefore, the search for a point Q_2 in the right image is constrained according to the given point Q_1 in the left image and the epipolar camera geometry. To find the relationship between points Q_1 and Q_2, we refer to the seminal paper by Longuet-Higgins [70]:

$$Q_2{}^T E\, Q_1 = 0 \tag{A.9}$$

where E is the essential matrix, $E = R_t\, [P_t]_x$, and $[P_t]_x$ denotes the skew-symmetric matrix:

$$[P_t]_x = \begin{bmatrix} 0 & t_3 & -t_2 \\ -t_3 & 0 & t_1 \\ t_2 & -t_1 & 0 \end{bmatrix}$$

For uncalibrated cameras, we can use the fundamental rather than the essential matrix.

Then

$$Q_2{}^T F\, Q_1 = 0 \tag{A.10}$$

and

$$l_2 = F\, Q_1, \qquad l_1 = F^T Q_2 \tag{A.11}$$

where l_1 and l_2 are the epipolar lines corresponding to the given points in image planes 1 and 2, respectively. Therefore, for two corresponding points $Q_1 = (u_{1p}\ -\lambda_1\ w_{1p})$ and $Q_2 = (u_{2p}\ -\lambda_2\ w_{2p})$, from equation (A.10) we have:

$$[u_{2p}\ -\lambda_2\ w_{2p}]\ F\ [u_{1p}\ -\lambda_1\ w_{1p}]^T = 0 \tag{A.12}$$

where F is the 3x3 fundamental matrix, which is a function of a rigid transformation:

$$F = R_t^T\,[P_t]_x = R_t^T\,K_t^T = R_t^T\,(-K_t) \tag{A.13}$$

For a given point (u_{2p} , w_{2p}) on image plane 2, the epipolar line l_1 on image plane 1 is given by equation (A.11) the parametric expression of the line is:

$$l_1 \equiv [a_1\ b_1\ c_1]^T = F^T[u_{2p}\ -\lambda_2\ w_{2p}]^T \tag{A.14}$$

and, as a line, l_1 can be expressed as:

$$w_1 = \eta_1 u_1 + \rho_1\lambda_1 \tag{A.15}$$

the slope and intercept are:

$$\eta_1 = \frac{-a_1}{c_1} \qquad \rho_1 = \frac{b_1}{c_1} \tag{A.16}$$

Similarly, for a point (u_{1p}, w_{1p}) on image plane 1 the corresponding epipolar line on image plane 2 is:

$$l_2 \equiv [a_2\ b_2\ c_2]^T = F\ [u_{1p}\ -\lambda_1\ w_{1p}]^T \tag{A.17}$$

$$w_2 = \eta_2 u_2 + \rho_2\lambda_2 \tag{A.18}$$

$$\eta_2 = \frac{-a_2}{c_2} \qquad \rho_2 = \frac{b_2}{c_2}\lambda_2 \tag{A.19}$$

At this stage, we have two epipolar lines, one in each image plane, expressed as parametric equations. The equations below show the relationship between the parameters of both epipolar lines.

Hence, let $[u_p, v_p, w_p]^T$ be the coordinates of a 3D point in the space, the expressions of the 3D-to-2D perspective projections of this point are given by:

$$\begin{bmatrix} u_P \\ v_P \\ w_P \end{bmatrix} = s_1 \begin{bmatrix} u_{1p} \\ -\lambda_1 \\ w_{1p} \end{bmatrix} \qquad \begin{bmatrix} u_P \\ v_P \\ w_P \end{bmatrix} = s_2 R_t \begin{bmatrix} u_{2p} \\ -\lambda_2 \\ w_{2p} \end{bmatrix}_t + P_t \tag{A.20}$$

where s_1 *and* s_2 *are* the scalar parameters of the perspective projection of each camera.

Inserting the term $[u_{1p} \quad -\lambda_1 \quad w_{1p}]^T$ from equation (A.17) into equation (A.14) and applying equation *A.20*:

$$l_2 \equiv \begin{bmatrix} a_2 \\ b_2 \\ c_2 \end{bmatrix} = \frac{F}{s_1} \cdot \begin{bmatrix} u_P & v_P & w_P \end{bmatrix}^T \xrightarrow{4.10} l_2 \equiv \begin{bmatrix} a_2 \\ b_2 \\ c_2 \end{bmatrix} = \frac{-R_t^T \cdot K_t}{s_1} \cdot \begin{bmatrix} u_P & v_P & w_P \end{bmatrix}^T \tag{A.21}$$

Developing analogous steps for the parametric expression of the epipolar line in image plane 1 (equation (A.14)) yields the following expression:

$$\begin{bmatrix} a_1 \\ b_1 \\ c_1 \end{bmatrix} = \frac{K_t}{s_2} \cdot \begin{bmatrix} u_P & v_P & w_P \end{bmatrix}^T \tag{A.22}$$

Thus, the relationship between the parameters is given by:

$$\begin{bmatrix} a_2 \\ b_2 \\ c_2 \end{bmatrix} = s \cdot R_T \cdot \begin{bmatrix} a_1 \\ b_1 \\ c_1 \end{bmatrix} \qquad s = -\frac{s_2}{s_1} \qquad (A.23)$$

The scalar constant *s* can be neglected if the parameters $[a_i\ b_i\ c_i]$ are expressed as a function of their respective intercepts and slopes as in equations (A.16) and (A.19).

From the relationship among the parameters of corresponding epipolar lines, it can be demonstrated that given a point (u_{2p}, w_{2p}) in image plane *2*, the epipolar line on plane *2* is:

$$\boxed{[u_2\ -\lambda_2\ w_2]\ R_t^T\ F^T\ [u_{2p}\ -\lambda_2\ w_{2p}]^T = 0} \qquad (A.24)$$

Finally, if $(u_{2p},\ -\lambda_2,\ w_{2p})$ and $(u_{1p},\ -\lambda_1,\ w_{1p})$ are corresponding points, then the expressions of their respective epipolar lines on image plane *2* should be equal, then:

$$s \cdot R_t^T \cdot F^T \cdot [u'_{2p} - \lambda_2\ w'_{2p}]^T = F \cdot [u'_{1p} - \lambda_1\ w'_{1p}]^T \qquad (A.25)$$

The approach is described above for a pair of points, one from each image. When matching contours, each contour is treated as a set of connected points, and so each contour leads to a set o epipolar lines. At this stage, we could define a procedure to match these sets of epipolar lines. This would assume pre-calibration of the cameras' intrinsic and extrinsic parameters, but the measure of similarity between closed contours does not depend on knowledge of corresponding points between the images. Further, the contours are not constrained to be planar in the 3D space.

Procedure: Match a pair of closed contours, one in each image (spatial domain)

In:

- $\Gamma_1, \Gamma_2,$; one closed contour from each image in $\{x_i,y_i\}$ form.
- R_t, P_t a rotation and translation matrix that defines the position of the second camera 2 with respect to camera 1.
- The focal lengths of cameras 1 and 2, λ_1 and λ_2.

Out:

- *DM*, a metric defining the similarity between the two contours.

Algorithm:

1. Compute the fundamental matrix from known camera extrinsic and intrinsic parameters (equation (A.13)).

2. Compute the set of epipolar lines in image plane *2*, that correspond to the set of contour points in image plane 1, using $l_2=FQ_1$. Express these in terms of their slopes and intercepts, η_2 and ρ_2.

3. Compute the set of epipolar lines in image plane *1*, that correspond to the set of contour points in image plane *2*, using $l_1=F^TQ_2$.

4. Knowing, the transformation from image plane *1* to the image plane *2* (R_t and P_t) compute the set of transformed epipolar lines in image plane *2*, $\bar{l}_2$, from the set of epipolar lines l_1. Express these in terms of their slopes and intercepts, $\overline{\eta}_2$ and $\overline{\rho}_2$.

5. Compute a distance metric between the two sets of epipolar lines in image plane *2*, using the set of slopes and intercepts $\{\eta_2, \rho_2\}$ arising from the contour in image plane *1*, and the set of slopes and intercepts $\{\overline{\eta}_2, \overline{\rho}_2\}$ arising from the contour in image plane *2*.

However, Wu and Sheu expressed the contours as Fourier series in a spectral domain. They claimed that there are two advantages of this approach. First, most of the information about shape is contained in the first few coefficients. Hence the matching process can be made more efficient than using complete point sets $\{x_i,y_i\}$. Second, the process is inherently more noise insensitive in the spectral domain, since higher frequency components can be easily truncated. Further, since the comparison is made in the spectral domain, the encoding is invariant to the choice of starting point on the

contour. For a contour Γ_1 in the left image and Γ_2 in the right image, the description through their Fourier series coefficients is as follows:

$$\Gamma_1 : \begin{bmatrix} u_1(t) \\ -\lambda_1 \\ w_1(t) \end{bmatrix} = \begin{bmatrix} a_{10} \\ -\lambda_1 \\ e_{10} \end{bmatrix} + \sum_{k=1}^{\infty} \begin{bmatrix} a_{1k} & b_{1k} \\ 0 & 0 \\ e_{1k} & f_{1k} \end{bmatrix} \cdot \begin{bmatrix} \cos\left(k\frac{2\pi t}{T}\right) \\ \sin\left(k\frac{2\pi t}{T}\right) \end{bmatrix}$$
$$\Gamma_2 : \begin{bmatrix} u_2(t) \\ -\lambda_2 \\ w_2(t) \end{bmatrix} = \begin{bmatrix} a_{20} \\ -\lambda_2 \\ e_{20} \end{bmatrix} + \sum_{k=1}^{\infty} \begin{bmatrix} a_{2k} & b_{2k} \\ 0 & 0 \\ e_{2k} & f_{2k} \end{bmatrix} \cdot \begin{bmatrix} \cos\left(k\frac{2\pi t}{T}\right) \\ \sin\left(k\frac{2\pi t}{T}\right) \end{bmatrix} \quad \text{(A.26)}$$

where t defines the sample, T the total number of samples around the contour, k the harmonic term and u_i, u_i and λ_i, the x, depth (focal) and y coordinates of the contour in the image plane i respectively. The Fourier series coefficients correspond to the a, b, e and f terms.

Thus, the spatial information of the contour is transformed into the frequency domain. The slopes (η) and intercepts (ρ) of the epipolar lines on image plane *2* from points extracted from image plane *1* are described in that domain using the approach described in equations (A.17) to (A.19), computing the same parameters of the epipolar lines on image plane *2* but from contour points of image plane *2* by applying equation (A.24). That is:

$$\eta(\omega) = \frac{\hat{a}_{t0} + \sum_{k=1}^{\infty}\left[\hat{a}_{tk} \cdot \cos(k\omega) + \hat{b}_{tk} \cdot \sin(k\omega)\right]}{\hat{e}_{t0} + \sum_{k=1}^{\infty}\left[\hat{e}_{tk} \cdot \cos(k\omega) + \hat{f}_{tk} \cdot \sin(k\omega)\right]}$$
$$\rho(\omega) = \frac{\hat{c}_{t0} + \sum_{k=1}^{\infty}\left[\hat{c}_{tk} \cdot \cos(k\omega) + \hat{d}_{tk} \cdot \sin(k\omega)\right]}{\hat{e}_{t0} + \sum_{k=1}^{\infty}\left[\hat{e}_{tk} \cdot \cos(k\omega) + \hat{f}_{tk} \cdot \sin(k\omega)\right]} \quad \text{(A.27)}$$

with:

$$\begin{bmatrix} \hat{a}_{t0} \\ \hat{c}_{t0} \\ \hat{e}_{t0} \end{bmatrix} = F \cdot \begin{bmatrix} a_{10} \\ -\lambda_1 \\ e_{10} \end{bmatrix} \qquad \begin{bmatrix} \hat{a}_{tk} & \hat{b}_{tk} \\ \hat{c}_{tk} & \hat{d}_{tk} \\ \hat{e}_{tk} & \hat{f}_{tk} \end{bmatrix} = F \cdot \begin{bmatrix} a_{1k} & b_{1k} \\ 0 & 0 \\ e_{1k} & f_{1k} \end{bmatrix} \tag{A.28}$$

for the coordinate points in the frequency domain from image plane *1* (left image), and:

$$\begin{bmatrix} \hat{a}_{t0} \\ \hat{c}_{t0} \\ \hat{e}_{t0} \end{bmatrix} = R_t^T \cdot F^T \cdot \begin{bmatrix} a_{20} \\ -\lambda_2 \\ e_{20} \end{bmatrix} \qquad \begin{bmatrix} \hat{a}_{tk} & \hat{b}_{tk} \\ \hat{c}_{tk} & \hat{d}_{tk} \\ \hat{e}_{tk} & \hat{f}_{tk} \end{bmatrix} = R_t^T \cdot F^T \cdot \begin{bmatrix} a_{1k} & b_{1k} \\ 0 & 0 \\ e_{1k} & f_{1k} \end{bmatrix} \tag{A.29}$$

for the other Fourier description of those points in image plane *2* (right image).

The sets of slopes and intercepts along a contour are periodic, thence:

$$\begin{bmatrix} \eta(\omega) \\ \rho(\omega) \end{bmatrix} = \begin{bmatrix} \eta_{a0} \\ \rho_{a0} \end{bmatrix} + \sum_{k=1}^{\infty} \begin{bmatrix} \eta_{ak} & \eta_{bk} \\ \rho_{ak} & \rho_{bk} \end{bmatrix} \cdot \begin{bmatrix} \cos(k\omega) \\ \sin(k\omega) \end{bmatrix} \tag{A.30}$$

To solve equation (A.30), in which there exist two unknowns but also four other infinite terms, an iterative solution is proposed. The algorithm iterates until an approximation error δ^j_{bound} computed from η_{ak}, η_{bk}, ρ_{ak} and $\rho_{bk,}$ converges to a minimum, which is predefined. Once this minimum has been reached the algorithm terminates and the Fourier descriptors for the set of slopes and intercepts of the epipolar lines on image plane *2* are defined. Recall that this is performed for the set of epipolar lines on image plane *2* calculated from the set of contour points on image plane *1* (equations (A.27*)* and (A.25*)*) and for the set of epipolar lines on image plane 2 computed from the set of contour points in image plane *2* (equations (A.24), (A.27) and (A.29)).

A.5 Minimum spectral distance and fuzzy logic implementation

The next step defines the measure of similarity between corresponding contours in different planes by means of a spectral distance, as both are now represented by Fourier descriptors of the same set of epipolar lines in the same plane. An additional, claimed benefit of application in the frequency domain is that it gains benefit of invariance to the position of the starting point on the contour.

The spectra of the slopes and intercepts of a contour *i* in the left image and of another contour *j* in the other image are given by η_{iak}, η_{ibk}, ρ_{iak}, ρ_{ibk} and η_{jak}, η_{jbk}, ρ_{jak}, ρ_{jbk}, respectively. Hence,

$$SD_{ij} = \alpha \cdot \left[\left(\eta_{ia0} - \eta_{ja0} \right)^2 + \left(\rho_{ia0} - \rho_{ja0} \right)^2 \right]^{\frac{1}{2}} + (1-\alpha) \cdot \left[\sum_{k=1}^{N} \left(\hat{\eta}_{ik} - \hat{\eta}_{jk} \right)^2 + \sum_{k=1}^{N} \left(\hat{\rho}_{ak} - \hat{\rho}_{jk} \right)^2 \right]^{\frac{1}{2}} \tag{A.31}$$

where

$$\hat{\eta}_{ik} = \sqrt{\eta_{iak}^2 + \eta_{ibk}^2} \qquad \hat{\rho}_{ik} = \sqrt{\rho_{iak}^2 + \rho_{ibk}^2}$$

$$\hat{\eta}_{jk} = \sqrt{\eta_{jak}^2 + \eta_{jbk}^2} \qquad \hat{\rho}_{jk} = \sqrt{\rho_{jak}^2 + \rho_{jbk}^2}$$

SD_{ij} is the spectral distance between descriptors of contours *i* and *j*, *k* is the harmonic number, *N* is the total number of harmonics and α is a factor constrained in the interval *0* to *1* that weights the relevance of the frequency terms of the descriptor. Thus, the value of this parameter α is related to the set of epipolar lines that defines each contour. For the case of similar shapes the *dc* term could acquire greater significance as this defines the position, whereas the higher frequency terms are of most interest in defining differences in shape of the contours.

An automatic method based on the principles of fuzzy logic [28] has been implemented to adjust this weighting factor. This is the degree of matching between contours $DM_{ij}(\alpha)$ for a certain α:

$$DM_{ij}(\alpha) = 1 - \frac{SD_{ij}}{\max(SD_{ij})} \tag{A.32}$$

Let *l* be a number of evenly spaced α's considered in the interval *[0…1]*. There will exist *l* different fuzzy sets $R(\alpha)$, containing the degrees of matching $DM_{ij}(\alpha)$, that will be calculated for each α and enhanced via a fuzzy AND operator ($\otimes$). The fuzzy degree of

matching, $\tilde{D}$, enhances the value for good matches and reduces the ones with a low degree of match DM_{ij}:

$$\tilde{D} = \bigcap_{m=1}^{l} R(\alpha_m) \equiv R(\alpha_1) \otimes R(\alpha_2) \otimes R(\alpha_3) \otimes \ldots \otimes R(\alpha_l) \tag{A.33}$$

$$R(\alpha_{m-1}) \otimes R(\alpha_m) \equiv \left\{ \left[(i, j), MR_{ij}(\alpha_{m-1}, \alpha_m) \right] \middle| i \in X, j \in Y \right\} \tag{A.34}$$

$$MR_{ij}(\alpha_{m-1}, \alpha_m) \equiv \begin{cases} \max\left\{0, DM_{ij}(\alpha_{m-1}) + DM_{ij}(\alpha_m) - 1\right\} \\ for \;\; DM_{ij}(\alpha_{m-1}) + DM_{ij}(\alpha_m) - 1 < U_t \\ \\ \frac{1}{2}\left[DM_{ij}(\alpha_{m-1}) + DM_{ij}(\alpha_m)\right] \\ for \;\; \mathrm{U_t} \le DM_{ij}(\alpha_{m-1}) + DM_{ij}(\alpha_m) - 1 \le 1 \end{cases}$$

where U_t is a threshold. The outcome is a table of the degree of matching between *m* contours on image the left image and *n* contours on the right image, Table *A.1.* The final algorithm is shown below. Also, figure A.4 shows a graphical representation.

pair	*1*	*2*	…	…	*N*
1	$\tilde{D}_{11}$	$\tilde{D}_{12}$	…	…	$\tilde{D}_{1n}$
2	$\tilde{D}_{21}$	$\tilde{D}_{22}$	…	…	…
…	…	…	…	…	…
m	$\tilde{D}_{m1}$	$\tilde{D}_{m2}$	…	…	$\tilde{D}_{mn}$

Table A.1. Degree of matching matrix

Procedure: Match a pair of closed contours, one in each image (spectral domain, Wu and Sheu)

In:

- Γ_1, Γ_2,; one closed contour from each image in $\{x_i, y_i\}$ form
- *N*, the number of harmonics used in a Fourier descriptor of each contour
- δ_{max}, an approximation error for the vibrating slope and intercept representations
- R_t, P_t a rotation and translation matrix that defines the position of the second camera 2 with respect to camera *1*.
- The focal lengths of cameras *1* and *2*, λ_1 and λ_2

Out:

- *DM*, a metric defining the similarity between the two contours

Algorithm:

1. Compute the fundamental matrix from known camera extrinsic and intrinsic parameters (equation (A.13)).
2. Compute the *(N+1)* Fourier series coefficients for contour, Γ_1, using equation *A.26*.
3. Compute the corresponding parameters for the epipolar lines in image plane *2* using equation (A.28).
4. Convert this into the spectral set of slope and intercept functions, $\eta_2(\omega)$ and $\rho_2(\omega)$ using equation (A.27), that exist within image plane *2*.
5. Expand $\eta_2(\omega)$ and $\rho_2(\omega)$ in Fourier series (η_{a0}, η_{ak}, η_{bk}, and ρ_{a0}, $\rho_{ak,,}$ ρ_{bk}) as expressed in equation (A.28). Use δ_{max} to determine the number of harmonics.
6. Compute the Fourier series coefficients for contour, Γ_2, using equation (A.27).
7. Compute the corresponding Fourier series coefficients for the epipolar lines in image plane *1*, then use the known transformation matrices to compute the Fourier series coefficients for the transformed epipolar lines in image plane *2* using equation (A.29).
8. Covert this into the spectral set of slope and intercept functions, $\overline{\eta}_2(\omega)$ and $\overline{\rho}_2(\omega)$, that exist within image plane *2*.
9. Expand $\overline{\eta}_2(\omega)$ and $\overline{\rho}_2(\omega)$ in Fourier series ($\overline{\eta}_{a0}, \overline{\eta}_{ak}, \overline{\eta}_{bk}$ and $\overline{\rho}_{b0}, \overline{\rho}_{ak}, \overline{\rho}_{bk}$), as expressed in equation (A.30). Use also δ_{max} to determine the number of harmonics.
10. Determine the minimum spectral distance between $\{\eta_{a0}, \eta_{ak}, \eta_{bk}, \rho_{a0}, \rho_{ak,,} \rho_{bk}\}$ and $\{\overline{\eta}_{a0}, \overline{\eta}_{ak}, \overline{\eta}_{bk}$ and $\overline{\rho}_{b0}, \overline{\rho}_{ak}, \overline{\rho}_{bk}\}$ (equation (A.31)).
11. Compute the degree of matching $DM_{ij}(\alpha)$ (equation (A.32)) and optimise the search for a set of equidistant values of the parameter α by using a fuzzy logic approach (equations (A.33) and (A.34)).

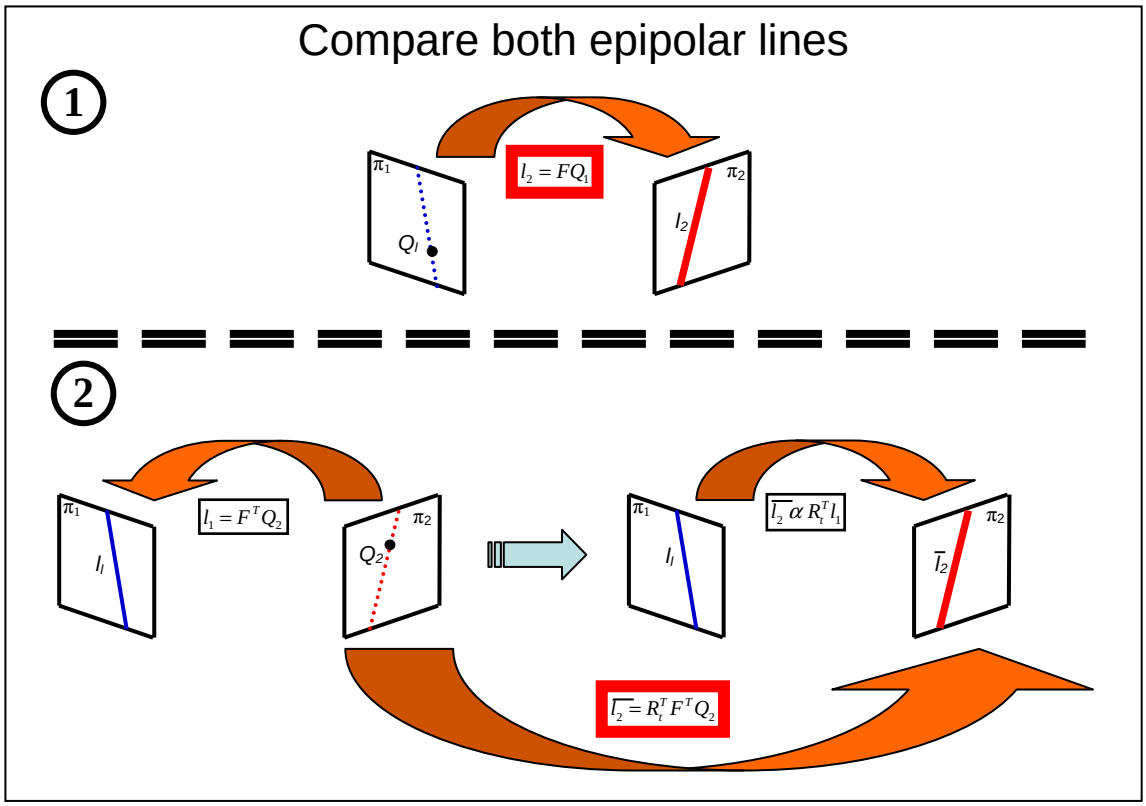


Figure A.4. Graphic representation of the epipolar lines algorithm. Top figures represents the relation between a point Q_1 on the left plane and epipolar line I_2 in the second plane. The blue dashed line is the unknown epipolar line on the left plane of the corresponding point of Q_1 on the right image (also unknown). The left bottom figure represents the relation of the epipolar line on the left plane of Q_2 (the potential corresponding point of Q_1) whereas the right bottom graph shows the relation between two epipolar lines. Therefore, the bottom biggest arrow gives a relation for extracting from a point Q_2 on the right image plane the epipolar line (also on the plane of Q_2) of the potential pairing point Q_1. Finally, the minimum spectral distance metric would compare slopes and intercepts of both epipolar lines on the right planes (solid red epipolar lines).

Using the above procedure, we obtain a minimum spectral distance (*MSD)*, normalized in the range 0<=*MSD*<=1, for each of *mn* pairs of contours in the two images, where *m* and *n* are the numbers of contours in the respective images. This can be represented in the form of a matrix. To obtain a final consistent labelling, it is necessary to find the optimum labelling between the possible pair of contours using the appropriate

constraints. Wu and Sheu used a fuzzy logic procedure in which they can prioritise the importance of position or shape of the contour using a parameter, α, that weights the first (*dc*) component of the Fourier series with respect to the higher harmonics.

A.6 Experimental results

First, the process will be demonstrated using synthetic data. A simple scene is represented by two planar contours in 3D space, instead of creating a complex setting, depicted in figure A.5. The camera *1* coordinate system is paced at the world reference frame, and camera 2 points toward the scene from a different, only slightly displaced position and orientation.

The 3D-to-2D projections of each camera for a perspective CCD projection are shown in figures A.6 and A.7. Note that the intersection of the axial rays (in cyan) with the plane of the object sets the origin of coordinates in the projected image.

Figure A.8 shows a representation of the Fourier analysis of the set of spatial coordinates corresponding to the largest contour up to an increasing number of harmonics. Note that a short number of *k* harmonics gives a fair approximation to the original signal. This is equivalent to smoothing the contour in the spatial domain. Figures A.9 and A.10 depict the values of the spectral coefficients of the expansions of slopes and intercepts $\hat{\eta}_{ik}$ $\hat{\eta}_{jk}$ $\hat{\rho}_{ik}$ $\hat{\rho}_{jk}$ (steps 5 and 9 above – equation (A.31)) with N=20 for the four possible combinations (i,j=1,2) between the two contours in the two images.

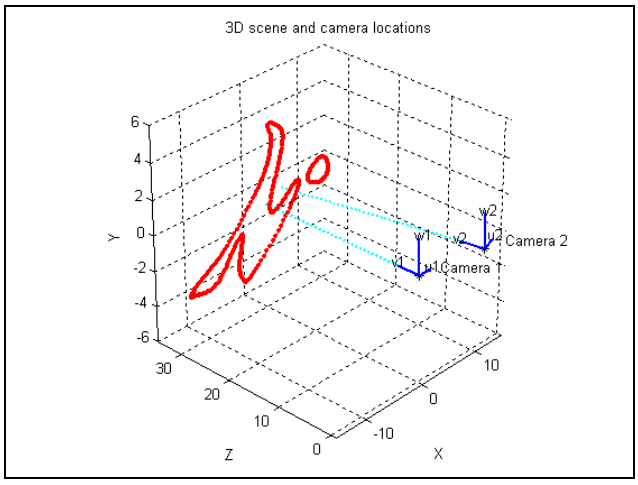


Figure A.5. 3D scene and camera geometry

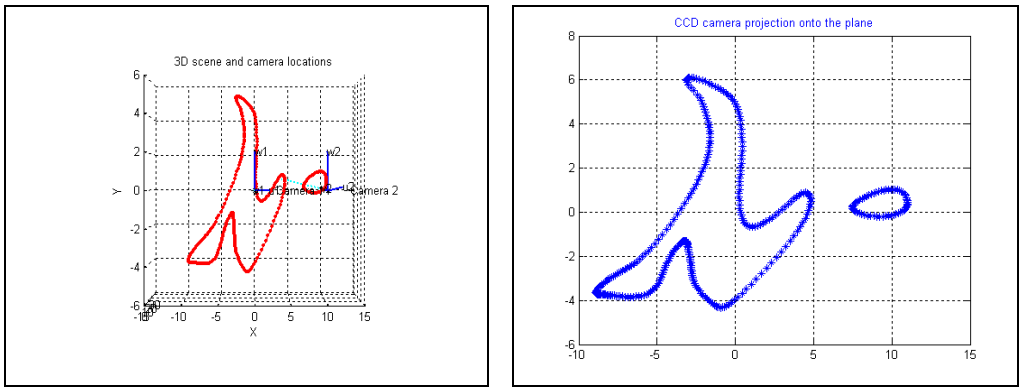


Figure A.6. 3D scene view from a viewpoint perpendicular to the axial ray of camera 1 (left), and CCD projection onto the plane (right).

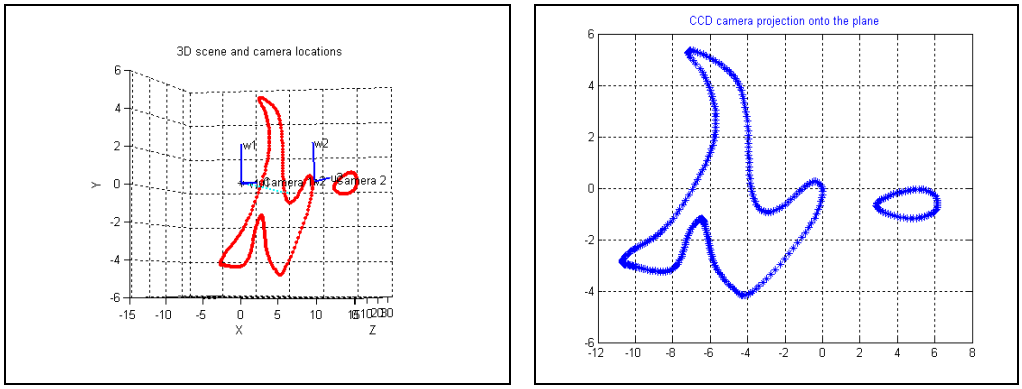


Figure A.7. 3D scene view from a viewpoint perpendicular to the axial ray of camera 2 (left) and CCD projection onto the plane (right).

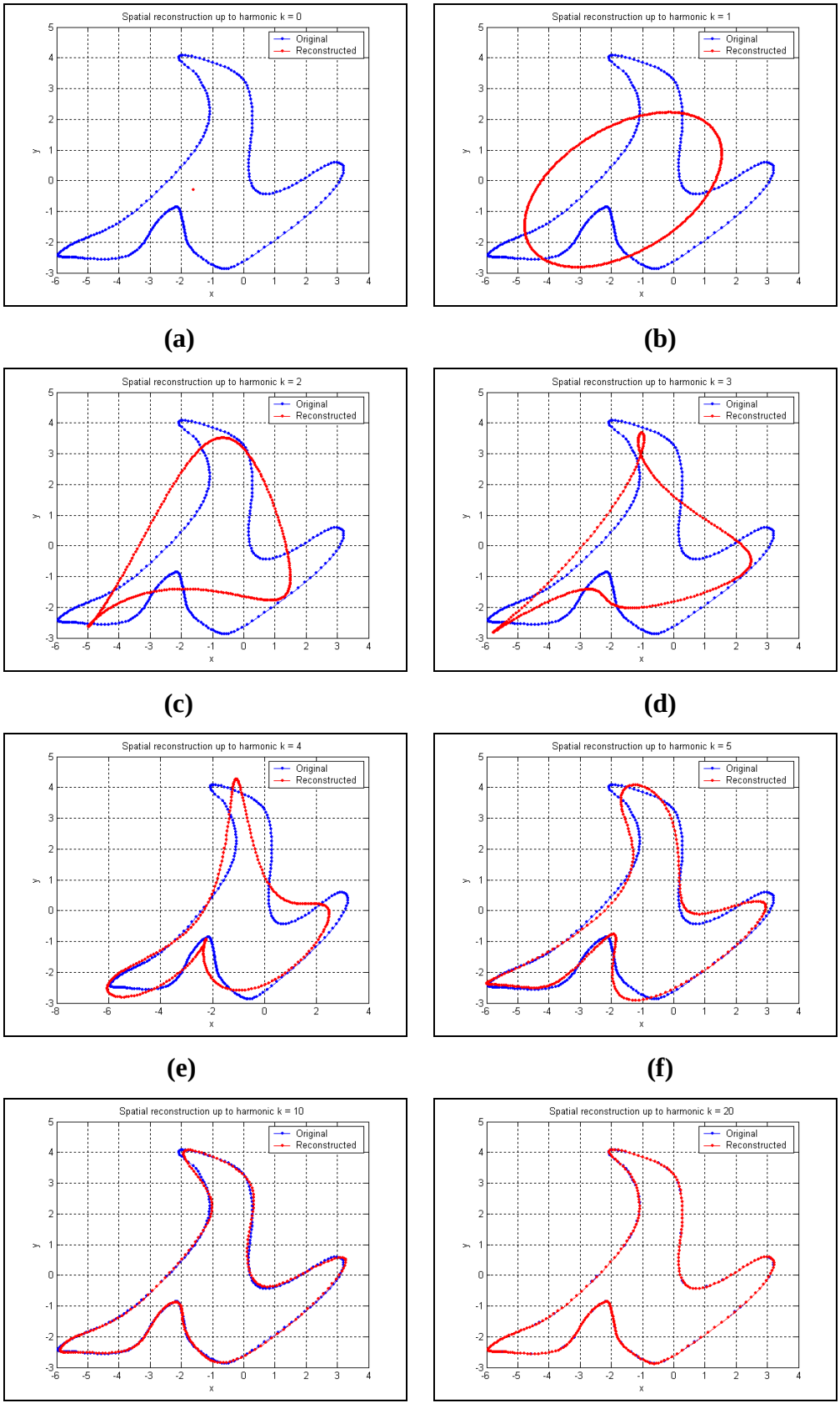


Figure A.8. Recovered contour in the spatial domain by using k harmonics. Successively, k=0, 1,2,3,4,5,10,20

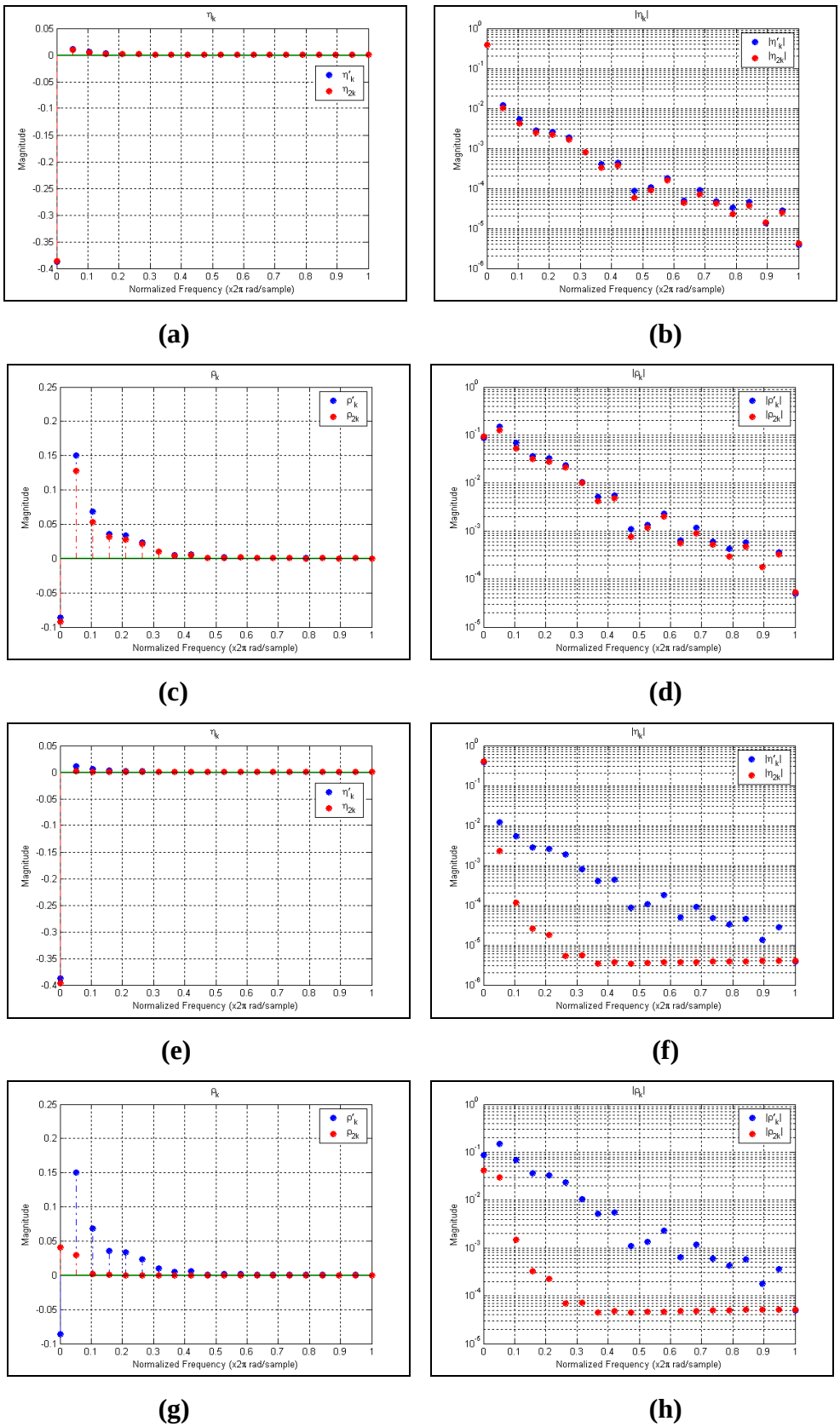


Figure A.9. Measure of slopes and intercepts of image data extracted from contour *i* in the left image and from contour *j* in the right image (equation *4.28*) [a-d] / (i,j)={1,1}; [e-h] / (i,j)={1,2}: a) and e) slopes $\hat{\eta}_{ik}$ $\hat{\eta}_{jk}$. b) and f) respective logarithmic plots. c) and g) intercepts $\hat{\rho}_{ik}$ $\hat{\rho}_{jk}$. d) and h) respective logarithmic plots.

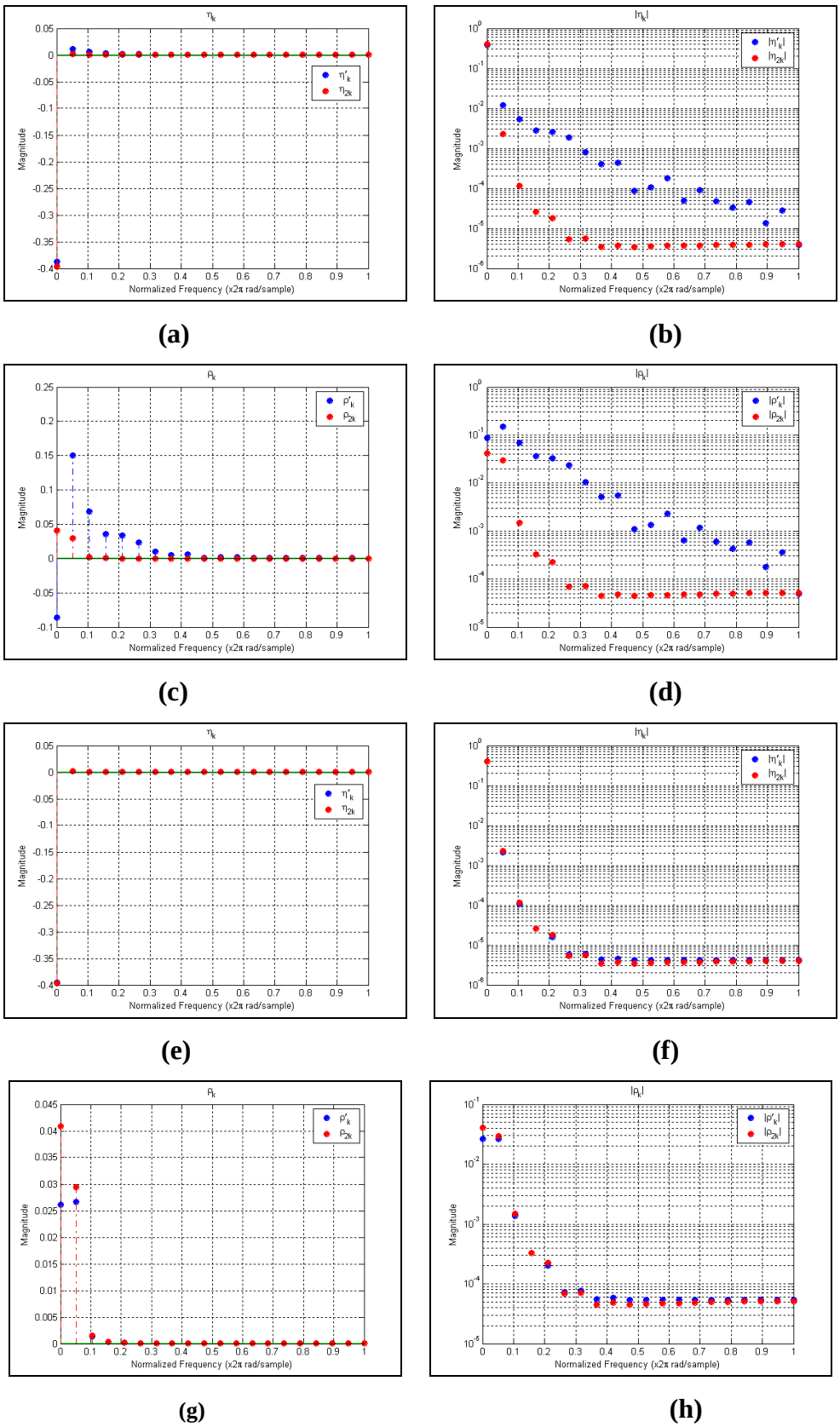


Figure A.10. Measure of slopes and intercepts of image data extracted from contour *i* in the left image and from contour *j* in the right image (equation (A.28)) [a-d] / (i,j)={2,1}; [e-h] / (i,j)={2,2}: a) and e) slopes $\hat{\eta}_{ik}$ $\hat{\eta}_{jk}$. b) and f) respective logarithmic plots. c) and g) intercepts $\hat{\rho}_{ik}$ $\hat{\rho}_{jk}$. d) and h) respective logarithmic plots.

The degree of matching performed by the current implementation of the algorithm for a threshold of $U_t = 0.6$ is:

$$\tilde{D}(0.6) = \begin{bmatrix} 0.9395 & 0 \\ 0 & 0.8947 \end{bmatrix}$$

This means that the algorithm identifies that the first contour in the first (left) image corresponds to the first contour in the second (right) image, with a degree of matching *(DM)* of *0.9395*. For the second contour in both images, notice that the *DM* is *0.8947*. The matching between the pair of first and second contours in the left image with the pair of second and first contours in the right image, respectively, are rated with 0. The matrix above yields the contour correspondence solution.

However, the satisfaction with the degree of matching obtained is a function of the parameter U_t. This parameter was set empirically. Figure A.11 shows the correspondence matrix of *DM* for different values of U_t. Values within the range [*0.1-0.6*] show similar results and good performance. However, for higher values of U_t the results are degraded:

$$\tilde{D}(0.7) = \begin{bmatrix} 0 & 0 \\ 0 & 0.8947 \end{bmatrix} \quad \tilde{D}(0.8) = \begin{bmatrix} 0 & 0 \\ 0 & 0.7893 \end{bmatrix} \quad \tilde{D}(0.9) = \begin{bmatrix} 0 & 0 \\ 0 & 0.4171 \end{bmatrix}$$

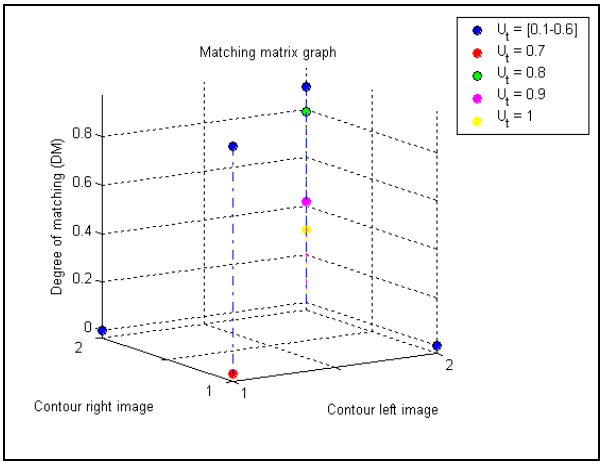


Figure A.11. Representation of DM_{ij} as a function of U_t (4 possible combinations for the case of two contours in each image). Notice that some dots mask others.

Thus, the selection of this parameter plays an important role for the final outcome. If a fuzzy logic procedure is sensible, and we cant really comment on its efficacy at this stage, then the setting of this parameter would have to be set automatically and justified. Further, the degree of matching *DM* (equations (A.31) and (A.32)) is a function of a parameter α, which crudely weights position as opposed to shape of the contour.

Finally, figures A.12 and A.13 depict, back in the spatial domain, the two sets of epipolar lines in image plane *2* extracted from the spectra of the slopes and intercepts computed by the algorithm. In figure A.14 we show the result of applying the algorithm to an indoor scene where three close contours have been detected. The algorithm satisfactorily rejects non-corresponding contours but there appears one mismatch.

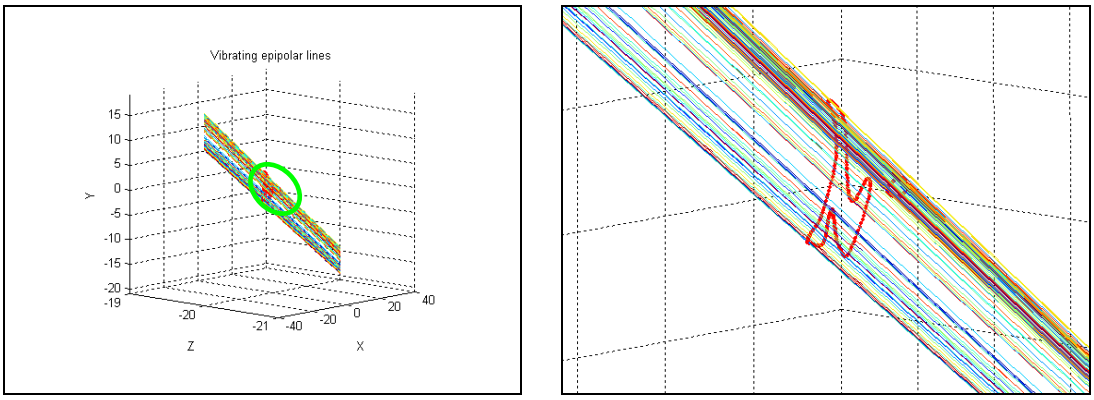


Figure A.12. (Left) Epipolar lines on image plane 2 constructed from contour points from image plane 2. Notice the epipolar lines are contained on the image plane for a focal length f=20. (Right) Zoom.

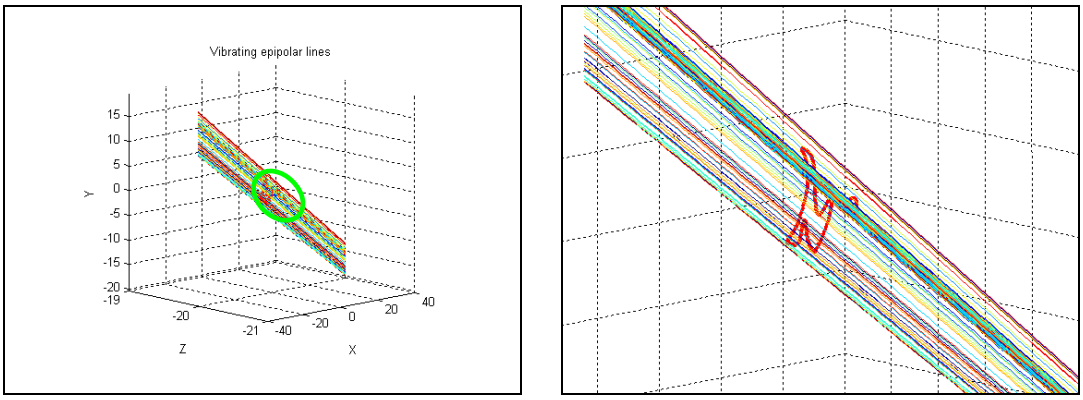


Figure A.13. (Left) Epipolar lines on image plane 2 constructed from contour points from image plane 1. Notice the epipolar lines are contained within the image plane for a focal length f=20. (Right) Zoom.

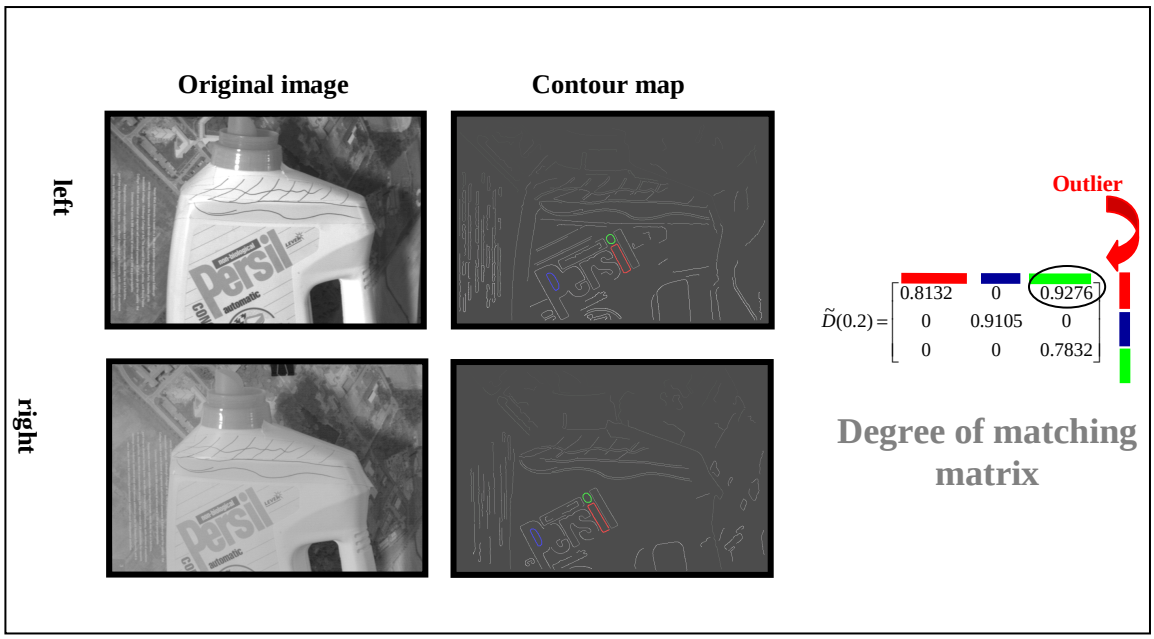


Figure A.14. Confusion matrix of a real scene. The algorithm is run over the three closed contours selected from the scene.

A.7 Summary

We have implemented an algorithm to match closed contours based on the fundamentals of epipolar geometry, following the earlier work of Wu and Sheu. We have applied it only to synthetic data, and shown that it is successful in identifying correspondences between simple contours, provided the epipolar geometry is known. Working in the frequency domain, the method has the benefits that it can be less sensitive to noise by taking only a determined number of frequency components, and of lower complexity than a full Fourier implementation. Further, normalization can provide scale invariance, and the algorithm appears to be invariant to starting point on the contour since the measure of dissimilarity is based solely on magnitude spectra. Invariance against rotation and translation is implicit since the geometry of the camera scene is contained in the fundamental matrix.

Examples in the literature typically apply Fourier descriptors to closed contours due to the need for periodicity for the Fourier analysis. However, there can be strategies to devise periodicity from open contour information. For example, when the two endpoints are reliable the travelling-back sequence from the last point to the initial point can be added to the original curve string [88]. If the endpoints are not reliable, a threshold

measurement based on the curvature extrema of the contour can be used to define limiting points.

As described, the method relies on knowledge of the camera intrinsic and extrinsic parameters. That may seem contradictory since the objective of the method is the matching of contours and, for the usual case of uncalibrated images, this is unknown. However, a fundamental matrix may be extracted once an initial set of potential matches has been computed, e.g. from some of the methods described in Chapters 2 and 4. The procedure could be considered as a robust method to support a pre-computed set of putative matches from an image pair that might give a rough estimation of the fundamental matrix. Consequently, it may be possible to develop a stark hypothesis (a fundamental matrix) and test, or an optimisation procedure.

References

[1] S. Adam, “Interprétation de Documents Techniques: des Outils à leur Intégration dans un Système à Base de Connaissances”, PhD Thesis, 2001

[2] N. A. Alvarez and J. M. Sanchiz, “Image Registration from Mutual Information of Edge Correspondences”, Progress in Pattern Recognition and Image Analysis and Applications, Proceedings, Vol. 3773, pp. 528-539, 2005

[3] R. Alferez and Y. Wang, “Geometric and Illumination Invariants for Object Recognition”, IEEE Transactions on Pattern Analysis and Machine Intelligence 21(6):505-536, June 1999

[4] A. Baumberg, “Reliable Feature Matching across Widely Separated Views” CVPR00, pp I: 774-781, 2000

[5] H. Bay, A. Ess, T. Tuytelaars and L. Van Gool, “Speeded Up Robust Features (SURF)”, Computer Vision and Image Understanding, vol. 110, pp. 346-359, 2008.

[6] R. Bamieh and R. Figueiredo, “A General Moments-Invariants/Attributed Graph Method for Three-Dimensional Object Recognition from a Single Image”, IEEE J Robotics Automation, Vol. 2, pp 31-41, March 1986

[7] T.O. Binford and T.S. Levitt, “Quasi-invariants: Theory and Exploitation”, Proc DARPA Image Understanding Workshop, pp.819-829, 1993

[8] E. Bayro-Corrochano and J. Lasenby, “Analysis and Computation of Projective Invariants from Multiple Views in the Geometric Algebra Framework”, International Journal of Pattern Recognition and Artificial Intelligence, Vol. 13, Part 8, pp. 1105-1122, 1999

[9] R. E. Blahut, “Principles and Practice of Information Theory”, Addison-Wesley, Reading, Mass, 1987

[10] P. J. Bessl and N. D. McKay, “A Method for Registration of 3D Shapes”, IEEE Transactions on Pattern Analysis and Machine Intelligence (14), pp 239-254, 1992

[11] C. de Boor, “A Practical Guide to Splines”, Applied Mathematical Sciences 27, Springer, 2002

[12] J.E. Bresenham, “Algorithm for Computer Control of a Digital Plotter”, IBM Systems Journal 4, No. 1, 25-30, 1965

[13] L. G. Brown, “A Survey of Image Registration Techniques”, ACM Computing Surveys 24, pp 326-376, 1992

[14] A. B. Bhatia and E. Wolf, “On the circle polynomials of Zernike and related orthogonal sets”, Proc. Cambridge Philosophical Society, 50, pp. 40-48, 1954.

[15] P. R. Bevington, “Data Reduction and Error Analysis for the Physical Sciences”, Mcgraw-Hill, 1969

[16] http://www.vision.caltech.edu/html-files/archive.html.

[17] J. Canny, “A Computational Approach to Edge Detection”, IEEE Transaction on Pattern Analysis and Machine Intelligence”, 8, pp 679-698, 1986

[18] S. Carlsson, “Recognizing Walking People”, ECCV 2000, 2000

[19] T. F. Cootes, G.J. Edwards and C.J. Taylor, “Active Appearance Models”, Proc. European Conference on Computer Vision, Vol. 2, pp. 484-498, Springer, 1998

[20] J. P. Collomosse and P. M. Hall, "Cubist Style Rendering from Photographs", IEEE Transactions on Visualization and Computer Graphics (TVCG), 9(4): pp. 443-453, 2003

[21] D. Chetverikov and J. Matas, “Periodic Textures as Distinguished Regions for Wide-Baseline Stereo Correspondence”, Proceedings Texture 2002, pp 25-29, Copenhagen, 2002

[22] O. Chum and J. Matas, “Geometric Hashing with Local Affine Frames”, Computer Vision and Pattern Recognition, pp. 879-884, 2006

[23] S. Derrode and F. Ghorbel, “Robust and Efficient Fourier-Mellin Transform Approximations for Gray-Level Image Reconstruction and Complete Invariant Description”, Computer Vision and Image Understanding, Vol. 83, No 1, pp.57-78, 2001

[24] R. Diestel, “Graph Theory”, Graduate Texts in Mathematics, Vol. 173, Springer-Verlag, 2005

[25] A. Dempster, N. Laird and D. Rubin, “Maximum Likelihood from Incomplete Data via the EM Algorithm”, Journal of the Royal Statistical Society, Series B, 39(1), pp.1-38, 1977

[26] G. Dorkó and C. Schmid, “Maximally Stable Local Description for Scale Selection”, in Proceedings of the 9th European Conference on Computer Vision (ECCV ’06), vol. 3954 of Lecture Notes in Computer Science, pp. 504–516, Graz, Austria, May 2006

[27] Y. Dufournaud, C. Schmid and R. Horaud, “Image Matching with Scale Adjustment”, Computer Vision and Image Understanding 93, pp 175-194, 2004

[28] O.G. Duarte, “Sistemas de lógica difusa – fundamentos”, Ingeniería e Investigación, (42), pp. 22-30, 1999

[29] J. H. Elder, “Ecological Statistics of Contour Grouping”, Lecture Notes in Computer Science, Vol. 2525, Proceedings of the Second International Workshop on Biologically Motivated Computer Vision, pp. 230-238, 2002

[30] S. Escalera, O. Pujol and P. Radeva, “Detection of Complex Salient Regions”, EURASIP Journal on Advances in Signal Processing, Vol. 2008, Article ID 451389, 11 pages, 2008

[31] M. A. Fischler and R. C. Bolles, “Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography”, Commun. Assoc. Cop. Mach., Vol 24:381-95, 1981

[32] F. Fraundorfer and H. Bischof, “Utilizing Saliency Operators for Image Matching”, Proc. International Workshop on Attention and Performance in Computer Vision. Graz, 2003

[33] G. Farin, J. Hoschek and M. –S. Kim, “Handbook of Computer Aided Design”, Elsevier, 2002

[34] P-E. Forssén and D. G. Lowe, “Shape Descriptors for Maximally Stable Regions”, IEEE 11th International Conference on Computer Vision, Vol. 1-6, pp. 1530-1537, 2007

[35] P-E. Forssén, “Maximally Stable Colour Regions for Recognition and Matching”, IEEE Conference on Computer Vision, Minneapolis, USA, June 2007

[36] D. A. Forsyth and J. Ponce, “Computer Vision a Modern Approach”, International Edition, Prentice Hall, 2003

[37] J. Flusser and T. Suk, “Degraded Image Analysis: An Invariant Approach”, IEEE Transactions of Pattern Analysis and Machine Intelligence, Vol. 20, No. 6, June 1998

[38] V. Ferrari, T. Tuytelaars and L. Van Gool, “Wide-baseline Multiple-view Correspondences”, Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’03), 2003

[39] J. Flusser and B. Zitová, “Combined Invariants to Linear Filtering and Rotation”, International Journal of Pattern Recognition and Artificial Intelligence, Vol. 13, No. 9, pp. 1123-1135, 1999

[40] D. J. Gawley, “Towards an Estimation Framework for some Problems in Computer Vision”, PhD Thesis, 2004

[41] R. Gal and D. Cohen-Or, “Salient Similarity Features for Partial Shape Matching and Similarity”, ACM Transactions on Graphics, Vol. 25, Issue 1, pp. 130-150, 2006

[42] P. R. Gill, W. Murray and M. H. Wright, “The Levenberg-Marquardt Method”, Practical Optimization, London: Academic Press, pp. 136-137, 1981

[43] T. Gevers and A. Smeulders, “A Comparative Study of several Color Models for Color Image Invariant Retrieval", Proceedings 1st International Workshop on Image Databases and Multimedia Search, p.17, Amsterdam, Netherlands, 1996

[44] H. W. Guggenheimer, “Differential Geometry”, McGraw-Hill Series in Higher Mathematics, 1963

[45] R. C. Gonzalez and P. Wintz, “Digital Image Processing”, Second Edition, Addison-Wesley Publishing Company, 1987

[46] R. C. Gonzalez and R. E. Woods, “Digital Image Processing”, Second Edition, Prentice Hall, 2002

[47] R. I. Hartley, “In Defense of the Eight-Point Algorithm”, IEEE Transaction on Pattern Recognition and Machine Intelligence, 19 (6), pp. 580-593, 1997

[48] L. Hajder, D. Cherverikov and I. Vajk, “Robust Structure from Motion under Weak Perspective”, Proceedings of the 2nd International Symposium on 3D Data Processing, Visualisation and Transmission (3DPVT’04), 2004

[49] F. S. Hill, Jr. “Computer Graphics using Open GL”, Prentice Hall, 2001

[50] D. L. Hall and J. Llinas, “Handbook of Multisensor Data Fusion”, The Electrical Engineering and Applied Signal Processing Series, 2001

[51] E. Haber and J. Modersitzki, “Intensity Gradient Based Registration and Fusion of Multi-Modal Images”, Optical Engineering, Vol. 46 Issue 5, 2007

[52] C. Harris and M. Stephens. “A Combined Corner and Edge Detector”, In Proceedings of Alvey Vision Conference, pp. 147-151, 1988.

[53] M-K. Hu, “Visual Pattern Recognition by Moment Invariants”, IRE Trans. Information Theory , IT-8, pp. 179-187, 1962

[54] R. Hartley and A. Zisserman, “Multiple View Geometry in Computer Vision”, Cambridge University Press, 2000

[55] T. Kadir “Scale, Saliency and Scene Description”, PhD Thesis, University of Oxford, 2002

[56] J. T. Kajiya, “The Rendering Equation” Computer Graphics (Proceedings of Siggraph ’86), 20(4):143-150, August 1986

[57] T. Kadir and M. Brady, “Saliency, Scale and Image Description” International Journal of Computer Vision, 45(2):83-105, November 2001

[58] T. Kadir, D. Boukerroui and M. Brady, “An Analysis of the Scale Saliency Algorithm” Technical Report OUEL No: 2264/03, University of Oxford, 2003.

[59] C. D. Kuglin and D. C. Hines, “The Phase Correlation Image Alignment Method”, Proceedings IEEE 1975, International Conference Cybernetics and Society, pp 163-165, September 1975

[60] A. Khotanzad and Y.H. Hong, “Invariant Image Recognition by Zernike Moments”, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 12, No. 5, pp 489-497, 1990

[61] P. Kovesi, “A Dimensionless Measure of Edge Significance”, In Proc. Digital Image Computing: Techniques and Applications, pp. 281–288, 1991

[62] P. Kovesi, “Image Features from Phase Congruency”, Technical Report 95/4, Dept. Comp. Sci., Univ. Western Australia, 1995

[63] S. J. Krotosky and M. M. Trivedi, “Mutual Information based Registration of Multimodal Stereo Videos for Person Tracking”, Computer Vision and Image Understanding, Vol. 106, Issue: 2-3, pp. 270-287, May-Jun 2007

[64] T. Kadir, A. Zisserman and M. Brady, “An Affine Invariant Salient Region Detector” Proceedings of the 8th European Conference on Computer Vision, Prague, Czech Republic, 2004

[65] J. Li and N. M. Allison, “A Comprehensive Review of Current Local Features for Computer Vision”, Neurocomputing, Vol. 71, Issue 20-71, pp. 1771-1787, June 2008

[66] T. Lindeberg and J. Gårding, “Shape-adapted Smoothing in Estimation of 3-d Shape Cues from Affine Deformations of Local 2-d Brightness Structure”. IVC, 15(6):415-434, June 1997

[67] Y. Li, “Reforming the Theory of Invariant Moments for Pattern Recognition”, Pattern Recognition Letters, Vol. 25, pp 723-730, July 1992

[68] T. Lindeberg, “Feature Detection with Automatic Scale Selection”, Int’l Journal of Computer Vision, Vol. 30(2), pp. 79-116, 1998

[69] B. Leibe, A. Leonardis and B. Schiele, “Robust Object Detection with Interleaved Categorization and Segmentation”, International Journal of Computer Vision, Vol. 77, Issue 1-3, pp. 259-289, May 2008

[70] H.C. Longuet-Higgins, “A computer algorithm for reconstructing a scene from two projections”, Nature, vol. 293, pp. 133-135, September 1981

[71] D. Lowe, “Three-Dimensional Object Recognition from Single Two-Dimensional Images”, Artificial Intelligence, Vol. 31, No. 3, pp. 355-395, 1987

[72] D. Lowe, “Object Recognition from Local Scale-invariant Features”, Proceedings of the 7th International Conference on Computer Vision, Kerkyra, Greece, pp.1150-1157, 1999

[73] D. Lowe, “Distinctive Image Features from Scale-invariant Keypoints”, International Journal of Computer Vision, Vol. 60 (2), pp. 91-110, 2004

[74] Y. Li and R. L. Stevenson, “Multimodal Image Registration based on edges and junctions”, Visual Communications and Image Processing, Vol. 6508 pp. 5080, SPIE, 2007

[75] S. Obdrzalek and J. Matas, “Object Recognition using Local Affine Frames on Maximally Stable Extremal Regions”, Lecture Notes in Computer Science, Vol. 4170, pp. 83-104, 2006

[76] P. Olver, G. Sapiro and A. Tannembaum, “Affine Invariant Detection: Edge Maps, Anisotropic Diffusion, and Active Contours”, Acta Applicandae Mathematicae: An International Survey Journal on Applying Mathematics and Mathematical Applications, Vol. 59, Number 1, October 1999

[77] S. Maitra, “Moment Invariants”, Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, Vol. 67, pp 667-699, 1979

[78] D. Marr, “Early Processing of Visual Information”, Philosophical Transactions of the Royal Society of London, Series B, 275, pp. 483-524, 1976

[79] J. Matas, O. Chum, M. Urban and T. Pajdla, “Robust Wide Baseline Stereo from Maximally Stable Extremal Regions”, BMVC, 384-393, 2002

[80] F. Maes, A. Collignon, D. Vandermuelen, G. Marchal and P. Suetens, “Multi-Modality Image Registration by Maximisation of Mutual Information”, IEEE Transactions on Medical Imaging 16, pp 187-198, 1997

[81] K. Mikolajczyk and C. Schmid, “Indexing based on Scale Invariant Interest Points” Proc. ICCV, Vancouver, Canada, pp 525-531, 2001

[82] F. Mindru, T. Tuytelaars, L. Van Gool and T. Moons, “Moment Invariants for Recognition under Changing Viewpoint and Illumination”, Computer Vision and Image Understanding, Vol. 94, Issue 1-3, pp. 3-27, 2004

[83] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir and L. Van Gool, “A Comparison of Affine Region Detectors”, IJCV 65 (1/2), pp. 43-72, 2005

[84] A. Noble, "Descriptions of Image Surfaces", PhD thesis, Department of Engineering Science, Oxford University, 1989.

[85] K. Nomizu and T. Sasaki, “Affine Differential Geometry”, Cambridge University Press, 1994

[86] http://www.robots.ox.ac.uk/_vgg/research/a_ne/index.html. Accessed August 2008.

[87] C. Park, K-H. Bae, S. Choi and J-H. Jung, “Image Fusion in Infrared Image and Visual Image using Normalized Mutual Information”, Signal Processing, Sensor Fusion and Target Recognition XVII, Proceedings of SPIE, 2008

[88] W. Philips, “Adaptive contour coding using warped polynomials”, Conference Guide ICASSP 96, Vol. 4, pp. 1867-1870, 1996

[89] P. Pritchett and A. Zisserman, “Wide Baseline Stereo Matching”, IEEE 6th ICCV, 754-760, 1998

[90] T.H. Reiss, “The Revised Fundamental Theorem of Moment Invariants”, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 13, pp 830-834, August 1991

[91] A. Roche, G. Malandain and N. Ayache, “The Correlation Ratio as a New Similarity Measure for Multimodal Image Registration”, Proceedings of the First International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI´98), Lecture Notes in Computer Science, Cambridge, USA, vol. 1496, pp 1115-1124, 1998

[92] D. B. Russakoff, C. Tomasi, T. Rohlfing and C. R. Maurer, “Image Similarity using Mutual Information of Regions”, Lecture Notes in Computer Science, Vol. 3023, pp. 596-607, 2004

[93] P. L. Rosin and G. A. W. West, “Nonparametric Segmentation of Curves into Various Representations”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 17, No. 12, December 1995

[94] P. D. Sampson, “Fitting Conic Sections to ‘Very Scattered’ Data: An iterative refinement of the Bookstein Algorithm’, Computer Vision Graphics and Image Processing, Vol. 18, pp. 97-108, 1982

[95] M. Swain and D. Ballard, “Color indexing”, International Journal of Computer Vision, Vol. 7, pp. 11-32, 1991

[96] S.M. Smith and J.M. Brady, “SUSAN - a New Approach to Low Level Image Processing”, *International Journal of Computer Vision*, 23(1):45-78, May 1997.

[97] I. Shimshoni, R. Basri and E. Rivlin, “A Geometric Interpretation of Weak-Perspective Motion”, IEEE Trans. on Pattern Analysis and Machine Intelligence, 21(3): pp. 252-257, 1999.

[98] I. J. Schoenberg, “Contributions to the Problem of Approximation of Equidistant Data by Analytic Functions”, Quart. Appl. Math., Vol. 4, pp. 45-99 and 112-141, 1946

[99] C.K. Shene, http://www.cs.mtu.edu/~shene/COURSES/cs3621/NOTES/spline, Accessed July 2008.

[100] L. Shao, T. Kadir and M. Brady, “Geometric and Photometric Invariant Distinctive Regions Detection”, Information Sciences, Vol. 177, Issue 4, pp. 1088-1122, 2007

[101] C. Schmid and R. Mohr, “Local Greyvalue Invariants for Image Retrieval”, IEEE transactions on pattern Analysis and Machine Intelligence, 19(5):872-877, May 1997

[102] C. Schmid, R. Mohr and C. Bauckhage, “Evaluation of Interest Point Detectors”, Int’l Journal of Computer Vision, 37(2), 151-172, 2000

[103] Y. Sun, J. Paik, A. Koschan, D. L. Page and M. A. Abidi, “Point Fingerprint: A New 3-D Object Representation Scheme”, IEEE Trans. On Systems, Man and Cybernetics - Part B: Cybernetics, Vol. 33, No. 4, pp. 712-717, August 2003

[104] C. Strecha, T. Tuytelaars and L. Van Gool, “Dense Matching of Multiple Wide-baseline Views”, International Conference on Computer Vision, pp 1194-1201, Nice, France 2003

[105] D. Tell and S. Carlsson, “Wide Baseline Point Matching using Affine Invariants Computed from Intensity Profiles” Proc 6th ECCV , Dublin, Ireland, Springer LNCS 1842-1843, June 2000

[106] M. R. Teague, “Image analysis via the general theory of moments”, Journal of the Optical Society of America, 70(8), pp. 920-930, 1979

[107] D. Tomazevic, B. Likar and F. Pernus, “Multi-feature Mutual Information”, Proceedings of the Society of Photo-Optical Instrumentation Engineers (SPIE), Vol. 5370, pp. 143-154, 2004

[108] T. Tuytelaars, "Local Invariant Features for Registration and Recognition", PhD Thesis, 2000

[109] E. Trucco and A. Verri, “Introductory Techniques for 3-D Computer Vision”, Prentice Hall, 1998

[110] T. Tuytelaars and L. Van Gool, “Wide Baseline Stereo Matching based on Local, Affinely Invariant Regions” British Machine Vision Conference, pp 412-422, 2000

[111] P. H. S. Torr and A. Zisserman, “MLESAC: A New Robust Estimator with Application to Estimating Image Geometry”, Computer Vision and Image Understanding, Vol. 78, pp. 138-156, 2000

[112] P. Viola, “Complex Feature Recognition: A Bayesian Approach for Learning to Recognize Objects”, Technical Report MIT AI Lab 1591, 1996

[113] E. Vincent and R. Laganière, “An Empirical Study of some Feature Matching Strategies”, Proceedings 15th International Conference on Vision Interface, pp. 139-145, Calgary, Canada, May 2002

[114] P. A. van den Elsen, E. J. D. Pol and M. A. Viergever, “Medical Image Matching – A Review with Classification”, IEEE Engineering in Medicine and Biology 12, pp 26-39, 1993

[115] L. Van Gool, T. Moons, E. Pauwels and A. Oosterlinck, “Vision and Lie’s Approach to Invariance”, Image and Vision Computing, Vol. 13, No. 4, pp 259-277, 1995

[116] L. Van Gool, T. Tuytelaars and A. Turina, “Local Features for Image Retrieval”, State-of-the-Art in Content-Based Image and Video Retrieval 1999: 21-41, 1999

[117] P. Viola and W. M. Wells, “Alignment by Maximization of Mutual Information”, International Journal of Computer Vision 24 (2), pp 137-154, 1997

[118] L. Wald, “Some terms of Reference in Data Fusion”, IEEE Transactions on Geosciences and Remote Sensing, 37,3, pp 1190-1193, 1999

[119] A. M. Wallace, “Tracking and Semantic Labeling of Boundary Data”, International Journal of Pattern Recognition and Artificial Intelligence, Vol. 13. No.6, pp. 859-879

[120] K. N. Walker, T. F. Cootes and C. J. Taylor, “Locating Salient Object Features”, Proceedings BMVC, vol.2, pp. 557-567, 1998

[121] I. Weiss, “Physics-like Invariants for Vision”, IUW, pp 1413-1421, 1995

[122] W. Wertheimer, “Untersuchungen zur Zehre von der Gestalt”, Psychologische Forschung, 4, pp 301-350

[123] S. Wang, T. Kubota, J. M. Siskind and J. Wang, “Salient Close Boundary Extraction with Ratio Contour”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27, Issue 4, pp. 546-561, 2005

[124] G. Wolberg, “Digital Image Warping”, IEE Computer Society Press, 1992

[125] J. Wood, “Invariant Pattern Recognition: A Review”, Pattern Recognition, 29(1):1-17, 1996

[126] L. Wang and T. Pavlidis, “Direct Gray-Scale Extraction of Features for Character Recognition”, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 15, pp 1053-1067, October 1993

[127] M-F. Wu and H-T. Sheu, “Contour-based correspondence using Fourier descriptors”, IEE Proc. Vision, Image Signal Processing Vol. 144, No. 3, June 1997.

[128] A. P. Witkin and J. M. Tenenbaum, “On the Role of Structure in Vision”, Human and Machine Vision, Beck, Hope & Rosenfeld, New York: Academic Press, 1983

[129] Z. Zhang, P. Anandan, and H. Shum, “What can be determined from a full and a weak perspective image?”, In Proceedings of the 7th International Conference on Computer Vision, pp 680-687, Kerkyra, Greece, IEEE Computer Society, IEEE Computer Society Press, 1999

[130] Z. Zhang, R. Deriche, O. Faugueras and Q-T. Luong, “A Robust Technique for Matching Two Uncalibrated Images through the Recovery of the Unknown Epipolar Geometry”, Rapport de Recherche No. 2273, Robotique, Image et Vision, INRIA, May 1994

[131] B. Zitová and J. Flusher, “Image Registration Methods: a Survey”, Image and Vision Computing, pp 977-1000, 2003

[132] P. Zimmons, “An Introduction to the Rendering Equation and Solution Methods”, Lecture Notes, Department of Computer Science, The University of North Carolina at Chapel Hill, 1997

[133] A. Zisserman and F. Schaffalitzky, “Viewpoint Invariant Descriptors for Image Matching”, 2001