ABSTRACT

Title of dissertation: COMPUTER VISION FOR SCENE TEXT ANALYSIS

Ali Zandifar, Doctor of Philosophy, 2004

Dissertation directed by: Professor Rama Chellappa
Electrical Engineering Department

Co-Advisors: Dr. Ramani Duraiswami
Professor Larry S. Davis
Department of Computer Science

The motivation of this dissertation is to develop a 'Seeing-Eye' video-based interface for the visually impaired to access environmental text information. We are concerned with those daily activities of the low-vision people involved with interpreting *'environmental text'* or *'scene text'* e.g., reading a newspaper, can labels and street signs.

First, we discuss the devopement of such a video-based interface. In this interface, the processed image of a scene text is read by off-the-shelf OCR and converted back to speech by Text-to-Speech(TTS) software. Our challenge is to feed a high quality image of a scene text for off-the-shelf OCR software under general pose of the the surface on which text is printed. To achieve this, various problems related to feature detection, mosaicing, auto-focus, zoom, and systems integration

were solved in the development of the system, and these are described.

We employ the video-based interface for the analysis of video of lectures/posters. In this application, the text is assumed to be on a plane. It is necessary for automatic analysis of video content to add modules such as enhancement, text segmentation, preprocessing video content, metric rectification, etc. We provide qualitative results to justify the algorithm and system integration.

For more general classes of surfaces that the text is printed on, such as bent or worked paper, we develop a novel method for 3D structure recovery and unwarping method. Deformed paper is isometric with a plane and the Gaussian curvature vanishes on every point on the surface. We show that these constraints lead to a closed set of equations that allow the recovery of the full geometric structure from a single image. We prove that these partial differential equations can be reduced to the Hopf equation that arises in non-linear wave propagation, and deformations of the paper can be interpreted in terms of the characteristics of this equation. A new exact integration of these equations relates the 3D structure of the surface to an image of a paper. In addition, we can generate such surfaces using the underlying equations. This method only uses information derived from the image of the boundary.

Furthermore, we employ the shape-from-texture method as an alternative to the method above to infer its 3D structure. We showed that for the consistency of normal vector field, we need to add extra conditions based on the surface model. Such conditions are are isometry and zero Gaussian curvature of the surface.

The theory underlying the method is novel and it raises new open research issues in the area of 3D reconstruction from single views. The novel contributions are: first, it is shown that certain linear and non-linear clues (contour knowledge information) are sufficient to recover the 3D structure of scene text; second, that with a priori of a page layout information, we can reconstruct a fronto-parallel view of a deformed page from differential geometric properties of a surface; third, that with a known cameral model we can recover 3D structure of a bent surface; forth, we present an integrated framework for analysis and rectification of scene texts from single views in general format; fifth, we provide the comparison with shape from texture approach and finally this work can be integrated as a visual prostheses for the visually impaired.

Our work has many applications in computer vision and computer graphics. The applications are diverse e.g. a generalized scanning device, digital flattening of creased documents, 3D reconstruction problem when correspondence fails, 3D reconstruction of single old photos, bending and creasing virtual paper, object classification, semantic extraction, scene description and so on.

COMPUTER VISION FOR SCENE TEXT ANALYSIS

by

Ali Zandifar

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2004

Advisory Commmittee:

Professor Rama Chellappa
Dr. Ramani Duraiswami
Professor Larry S. Davis
Professor Eyad Abed
Professor Ralph Etienne-Cummings

ACKNOWLEDGMENTS

I would like to appreciate the committee for their time and support. It is an honor to have each of them serve in my committee.

First and foremost I am grateful to thank my advisor, Dr. Ramani Duraiswami for his endless support and guidance throughout my Ph.D. work. He has been a superb advisor and true friend for the last five years. I have learned from him how to tackle perplexed problems in the most efficient and organized way.

I am enchanted to have the opportunity to work with Professor Larry S. Davis and Professor Rama Chellappa throughout my Ph.D. thesis. It has been a pleasure to work with and learn from such extraordinary individuals.

I would also like to thank Dr. Nail Gumerov. Without his extraordinary theoretical ideas and computational expertise, this thesis would have been a distant dream. Special thanks to Dr. Daniel DeMenthon, Dr. David Doermann and my colleagues whom I enjoyed working with including: Ser-nam Lim, Vinay Shet, Harsh Nanda, Ahmed Elgammal.

Above all, I would like to dedicate my love to my parents for their endless support from the first day of my education. Without their encouragement and love, finishing this thesis would have been impossible.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1.1 Preamble

‘The blind live in a world of hearing and touch (where) knowledge of the object is gained through the sense of touch since hearing gives no information unless sound is an integral part. Thus, objects that are too finite or too large or too far for the sense of touch cannot be experienced’ (Sholl-Schur [75]). Text falls under this category.

1.2 Motivation

Imagine the frustration of walking in a street where you can not read street signs, of not being able to read newspaper. If you are visually impaired, this is the scenario for your lifetime. An ever-increasing segment of the population (6 million in the US) suffers from low vision brought about by complication of diseases and old age as human longevity increases [2]. The total is far greater if we consider those who cannot see fine print without spectacles. While this group may be classified as blind but they do have some vision which can be assisted by prostheses and visual aids. By the definition of the American Foundation of the Blind (AFB) [2], the biggest problem for such people is lack of information about the environment. Therefore, the challenge is to provide access to environmental information.

Here, we are concerned with those daily activities involved with interpreting 'Environmental text' or 'Scene text'. Scene text of various kinds, including bound printed material(books, newspapers, magazines, brochures, etc.), distant scene text(street signs,warning, shop labels and directions) and other miscellaneous text (nutrition labels, ATM instructions, etc.) abound in the world. In table 1.2, we gathered different scene text categories [51].

Such textual information above is too far or too finite for a sense of touch. Therefore, the visually impaired people cannot be able to recognize and interpret them for their daily activities. A recent extensive survey by Massof [54, 55] shows that the visually impaired are most desirous of portable technologies that enable them to take personal care of their needs. Since much of this population lives alone, there is a need for devices and services that can assist them in getting about, and providing them with access to information [19].

There are many efforts to provide access to textual information for the visually impaired. There is a huge effort on recording books [5], newspapers and magazines [63] and commercially available books on audio-cassette. However, for news articles and documents, audio conversion is an unavoidable step, and due to the excessive volume of audio-cassette library storage, not all information is going to be available

Item	**Features**	**Complications**
Mail	Letterheads, addresses,	Background, font size and style handwritten, textured surface
Street signs	Names, directional, colored	Colors and shapes, background (on block and scene), surfaces, location, size, lighting, occlusion
Notices	Notice boards contain many notices of varying relevance and urgency	Wide variety of fonts, sizes, often overlapping, some handwritten and graphical
Medicines	Dosage, drug name, warning labels	Shape of container, label layout, often small font
Food Packaging	Instructions, quantities, contents brand names, used-by date	Colors, graphics, shape of packaging variable
Flat Printed text	Newspapers, magazines, posters	Text, graphics, color, size
Curled Printed text	Reading article while in the hand, animation	Non-flat surface
Advertising	Wide variety of information	Different formats from pamphlets to books, colors, layouts, graphics
Maps	Location of text, spatial representation, symbols	Screen characters, button association, time restrictions, animated
Money	Denomination	Graphical, poor quality
Electronic devices	Digital screen and associated buttons	Colored, textured surfaces, shape

Table 1.1: Scene Text Categories

when required. Personal scanning devices in conjunction with OCR and speech synthesis provide access to text in books and magazines while scanned by the scanner [4]. However, there are other scene texts which are not necessarily located on pages of books or magazines or not close to a scanning device, e.g. nutrition labels on arbitrary shape cans, street signs ,etc. In these cases, the scanner-based interface can not be used.

Over the last decade, there has been major advances in hardware e.g., cheaper and more efficient cameras, display, faster and smaller computers and in software and algorithms e.g., advances in OCR, text-to-speech software, computer vision algorithms. Therefore, video-based acquisition of text is an alternative that provides portable access to text for the visually impaired. The interface includes a computer, a digital video-camera, audio interface and off-the shelf OCR software. Our 'Seeing-Eye' computer is composed of three modules:

- *Data Acquisition:* Image acquisition device e.g. a digital camera that is used to interact with the environment.

- *Processing:* This module that stabilizes and enhances images, identifies text images, corrects for physical and projective distortion, reading text and converting to speech format.

- *Output interface:* A speech synthesis display for the visually impaired to convey the recognized scene text to the user.

In this system (shown in Fig. 1.1), the camera captures scene text, with full intelligent control on focus and zoom. The computing device pre-processes the video before feeding it to OCR. There are constraints on the quality of the input image to OCR software. In general, for better quality of OCR output, we require that:

- Text images are binarized and enhanced;

- Text images are from flat scene texts;

- All text has the same degree of skew and slant;

- Mixture of text and graphics have components layout seperated;

- The text image have at least some minimum number of pixels per character.

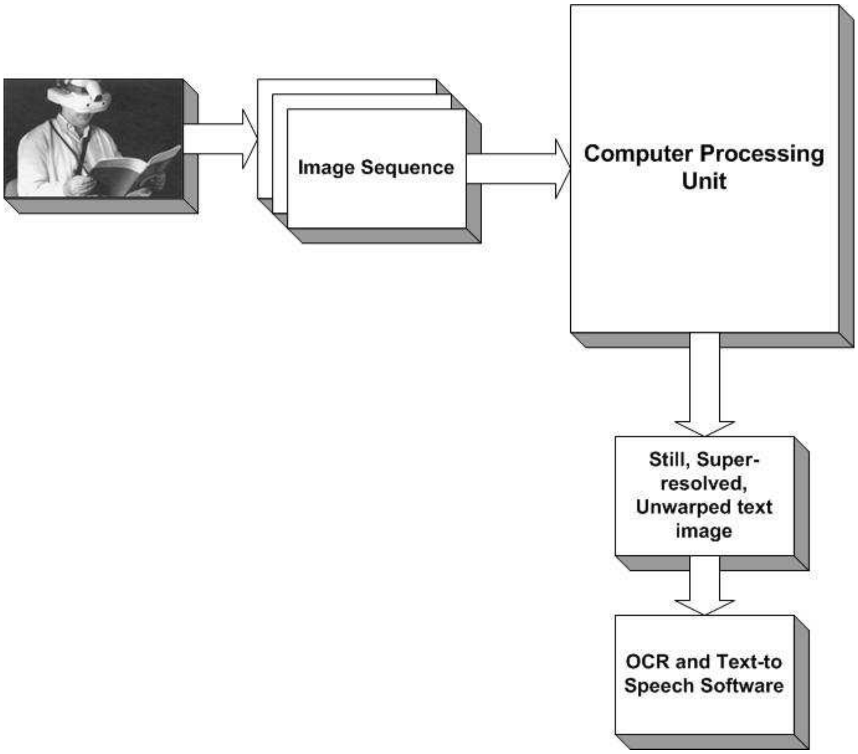


Image Sequence
Computer Processing Unit
Still, Super-resolved, Unwarped text image
OCR and Text-to Speech Software

Figure 1.1: Schematic of Seeing-eye computer

Therefore, the challenges for the vision system can be summarized into these research issues:

1. Preprocessing captured video-content;
2. Text segmentation and identification;
3. Image enhancement;
4. 3D structure recovery and unwarping text images;
5. OCR from degraded text;
6. System Integration.

In the next section, we will present the main contribution of the thesis which includes the solution to the problems stated above.

1.3 Contribution Overview

1.3.1 Preliminary Video-based Interface to Scene Text

We presented a preliminary prototype device for scene text acquisition and processing in [93]. In this system, the camera captures *'scene text'* with control on focus and zoom that depends on orientation and quality of the document video. The computing device pre-processes video before feeding into the OCR, by performing operations such as image mosaicing, auto-focus and binarization.

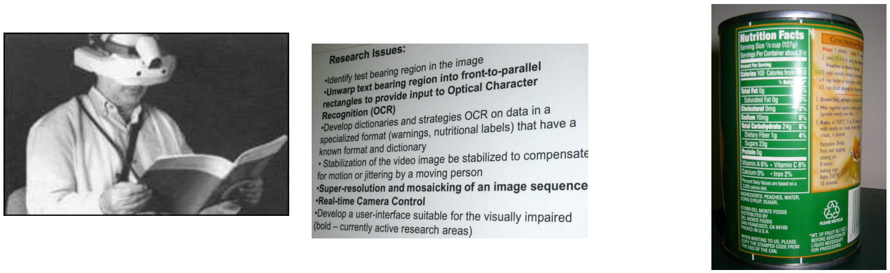
1.3.2 Computer Vision for Planar Scene Text

Detection and recognition of textual information in an image or video is important for many applications. The increased resolution and capabilities of digital cameras and faster mobile processing allow for the development of interesting systems. We present an application based on the capture of information presented at a slide-show presentation, or at a poster session. We describe the development of a system to process the textual and graphical information in such presentations. The application integrates video and image processing, document layout understanding, optical character recognition (OCR) and pattern recognition. The digital imaging device captures slides/poster images, and the computing module pre-processes and annotates the content. Various problems related to metric rectification, key frame extraction, text detection, enhancement and system integration are addressed. The results are promising for applications such as a mobile text reader for the visually impaired. By using powerful text-processing algorithms, we can extend this framework to other applications e.g.document and conference archiving, camera-based semantics extraction and ontology creation.

1.3.3 Unwarping and 3D structure recovery for surfaces applicable to planes

We consider the problem of 3D structure recovery and unwarping text images from single views as the core of the dissertation. We show that differential geometric shape properties are sufficient to solve 3D structure equations. The input is the image of a scene text, possibly curled and the goal is to construct a fronto-parallel view of the text image before inputting OCR.

When a picture of text is captured by a camera, our problem is to unwarp the captured image to its flat, fronto-parallel representation. In the simple case that the text surface is flat, the problem reduces to one of undoing a projection of a rectangle, and the rectification (or unwarping) can be achieved by computing a simple homography. A harder problem is posed when we consider the case when the piece of paper is itself deformed or bent. In this case the unwarping must undo both the effects of the three-dimensional bending of the surface, and the effect of the projection. Examples includes reading newspaper without any constraint on flatness of a paper sheet(Fig 1.2-a), reading planar signs (Fig 1.2-b)and can labels (Fig 1.2-c).



Research Issues:

- Identify test bearing region in the image
- **Unwarp text bearing region into front-to-parallel rectangles to provide input to Optical Character Recognition (OCR)**
- Develop dictionaries and strategies OCR on data in a specialized format (warnings, nutritional labels) that have a known format and dictionary
- Stabilization of the video image be stabilized to compensate for motion or jittering by a moving person
- **Super-resolution and mosaicking of an image sequence**
- **Real-time Camera Control**
- Develop a user-interface suitable for the visually impaired

(bold – currently active research areas)

(a) (b) (c)

Figure 1.2: a) Curled newspaper b) Planar Scene Text c) Cylindrical Can

Knowledge of the differential geometry of surfaces can provide a very powerful set of relations for analysis. However, most quantitative use of differential geometry has been restricted to range data, while the analysis of image data has been primarily qualitative. A celebrated exception is the work of Koenderink on occluding contours.

The deformations of paper surfaces satisfy the conditions of isometry (the lengths of curves and areas enclosed by them, are conserved by the bending) and vanishing Gaussian curvature (also called intrinsic curvature).

Previous authors have attempted to enforce these conditions in reconstruction. However, in their approaches, they essentially enforced these as *constraints* to a process of polynomial/spline fitting using data obtained on the surface. In contrast, we *solve* these system of equations, and show that *information on the bounding contour is sufficient to determine structure completely.* Further, exact correspondence information on the bounding contour is only needed at corner points.

1.3.4 Unwarping and 3D structure recovery using texture information

we developed the shape-from-texture method as an alternative to the method above to infer the 3D structure. We showed that for the consistency of normal vector field, we need to add extra conditions based on the surface model. Such conditions are isometry and zero Gaussian curvature of the surface.

1.4 Other Applications

Other than its theoretical importance, our research can potentially benefit diverse computer vision applications, e.g. rectifying scanner output, digital flattening of creased documents, 3D reconstruction without correspondence or with occlusion of paper, and perhaps most importantly, optical character recognition of scene text. Other than its theoretical importance, our research can potentially benefit diverse computer vision applications, e.g. a generalized scanning device, digital flattening of creased documents, 3D reconstruction problem when correspondence fails, 3D reconstruction of single old photos, bending and creasing virtual paper, object classification, semantic extraction, scene description and so on.

1.5 Dissertation Organization

The outline of the dissertation is as follows: Chapter 2 describes a preliminary video-based interface to textual information. In chapter 3 we develop a computer vision framework for analysis of scene text printed on planes. In this chapter, we give the solution based on the previous solid computer vision algorithms e.g. metric rectification of planes. To motivate the problem, we consider poster/presentation analysis problem for such interface. in Chapter 4, we investigate 3D structure recovery and unwarping for more general class of surfaces (applicable to planes) from single views. In chapter 5, we present an alternative shape from texture method to 3D structure recovery.

Chapter 2

A video based interface to Scene Text

2.1 Introduction

Video-based acquisition of text is an alternative that provides portable access to text for the visually impaired. However, before such a system can be successfully implemented, several problems arising from text identification in images, low resolution sensors, image stabilization, text being warped, and others on the one hand, and practical system integration issues, on the other, have to be solved. We describe here the development of a preliminary prototype device for scene text acquisition and processing. The system consists of a computer, a digital Video Camera, an audio interface and off-the-shelf OCR software. The camera captures text from the scene, with full control of focus and zoom that depends on orientation and quality of the document video. Video is 'conditioned' before OCR, by performing operations

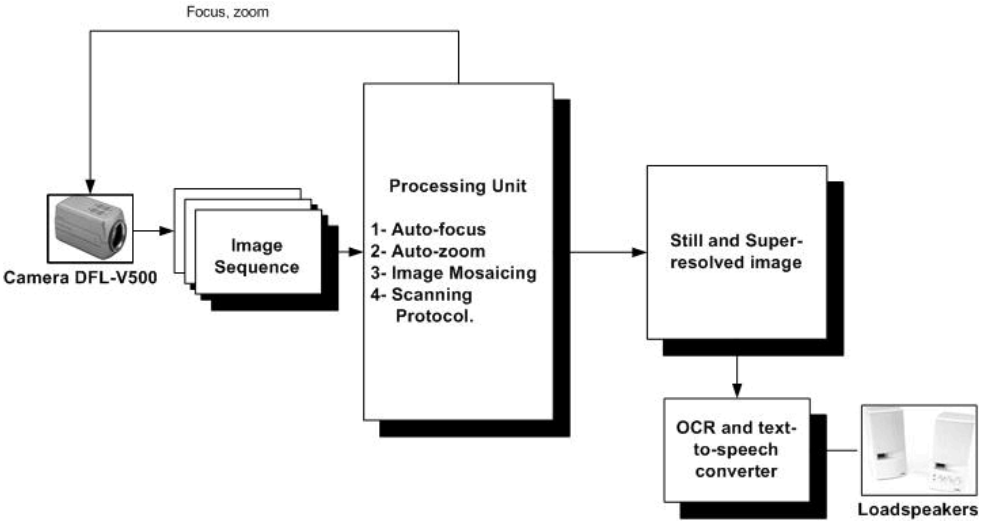


Figure 2.1: Schematic of a Seeing-Eye computer system

such as image mosaicing, binarization, etc. The OCR software recognizes text from still and super-resolved images of whole text blocks, and the recognized text is read back by speech-to-text. In general, off-the-shelf OCR systems are successful if:

- Document images are binarized and enhanced.
- All Text has the same degree of skew and slant.
- The text image has sufficient number of pixels per character experimentally calculated as ≥ 12.

These criteria are often not met by the images of text captured by commercial digital video cameras which have low-resolution and narrow field of view. To successfully use off-the-shelf OCR software, we must construct high resolution input to OCR, we use image registration and mosaicing techniques cited at [6][9][38][47][70][77][79] to read patches of text and stitch them together to make a super-resolved text image [39][73][58][59].

To calculate number of frames(patches), it is necessary to determine font-size of text, we then zoom into each patch to obtain the image that satisfy font-size constraint and capture the whole page while it is in-focus. Then, the super-resolved image from the mosaicing algorithm is interpreted by OCR and TTS software.

In section 2, we discuss a fast mosaicing method that at the individual frame level employs image processing to clean up the text, followed by a two stage registration procedure and mosaicing. Both mosaicing and successful OCR require clear images with maximal resolution, and hence the system needs auto-focus and auto-zoom capabilities. These are described in section 2.3. Section 2.4 describes the system development and presents some sample results.

2.2 Image Registration Algorithm

Mosaicing is a way to create large still images from a set of images taken by a moving camera. Creating a mosaic from video sequences is useful for many application such as image browsing, video surveillance and virtual reality. Although a page of a magazine or a book can be captured in a single image, it may not be readable by OCR. We present an image mosaicing algorithm to enhance OCR results.

Our image mosaicing method uses a frequency-based analysis to obtain an initial transformation matrix. Then, the initial estimate is refined using an intensity-based method to get both speed and best match [9][47][70][77].

In figure 2, a rectangle(page) is projected to an image plane of a camera at different pose(time); images 1 and 2, respectively . Consider a point on the rectangle,

(X, Y, Z); its projection onto image 1 and image 2 are (x, y), $(x^{'}, y^{'})$, respectively: The point on image 1 is related to corresponding point on image 2 by:

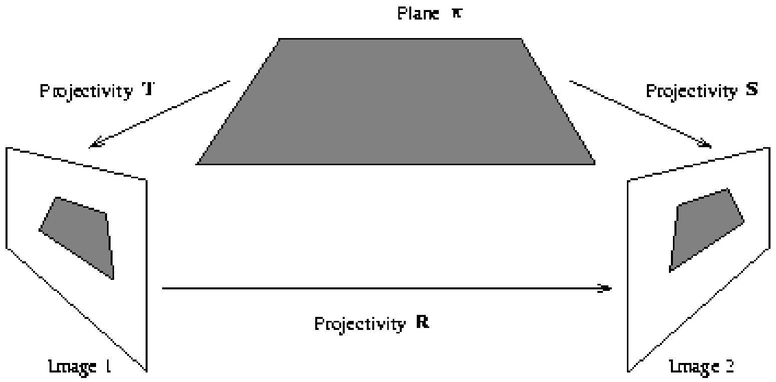


Figure 2.2: Projection of a plane onto image plane of a camera at different pose

$$\begin{pmatrix} x^{'} \\ y^{'} \\ 1 \end{pmatrix} = \omega \begin{pmatrix} m_0 & m_1 & m_2 \\ m_3 & m_4 & m_5 \\ m_6 & m_7 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{2.1}$$

The equation above defines a homographic deformation between the two frames. There are two common situations in which images are related by homographic transformations: images obtained by rotating the camera about its center (a motion constraint); images of a plane from varying viewpoints (a structure constraint). Here, we concentrate only on sequences obtained from any viewpoint of a planar surface.

In the former case, the eight unknowns are the three Euler angles (θ, ψ, ϕ) of a camera and the five intrinsic camera parameters. In the latter case, we have the three parameters of a planar surface and the five camera intrinsic parameters [77].

Frequency domain based image registration has a long history, but has largely been restricted only to translation between image pairs through calculating phase correlation. This method is fast in computation and independent of illumination change between the two frames. There is a more general phase correlation method to solve for affine image transformations, but is computationally expensive [70]. We use phase-correlation method solution to the transformation matrix as an initial guess for an intensity-based method [47]. Consider two successive frames of a video sequence, f_1 and f_2. If the motion vector is assumed to be purely translation $(\triangle x, \triangle y)$, then :

$$f_2(x, y) \approx f_1(x - \triangle x, y - \triangle y) \tag{2.2}$$

Let $F_1(u, v)$ and $F_2(u, v)$ be the Fourier transforms of f_1 and f_2. Applying the Fourier shift theorem gives:

$$F_2(u, v) \approx e^{-2\pi j(u\triangle x + v\triangle y)} F_1(u, v) \tag{2.3}$$

The Cross-power spectrum of F_1 and F_2 (where F_2^* is the complex conjugate of F_2) as :

$$CPS = \frac{F_1(u, v)F_2^*(u, v)}{|F_1(u, v)F_2^*(u, v)|} \approx e^{-2\pi j(u\triangle x + v\triangle y)} \tag{2.4}$$

Ideally, the inverse of the cross-power spectrum is an impulse at location $(\triangle x, \triangle y)$. In real applications, there would be many impulses due to different motions in the scene, parallax effects, etc. In real applications two consecutive frames are not completely shifted replicas of each other. The above equation is satisfied only in the overlap region between the two frames. In order to remove the repeating nature of the frequency spectrum and to give less value to the boundaries, we used a raised cosine function to make a window that smoothly reaches zero at the boundaries. This spatial filter gives more weights to the pixel close to center of an image rather than the boundaries. This spatial filter is formulated as bellow:

$$w(i) = 0.54 - 0.46\cos(\frac{2\pi i}{N}), i = 0 \cdots N(Hamming\ \ Cosine\ \ Window) \tag{2.5}$$

Summarizing the phase correlation algorithm:

- Apply the Hamming cosine spatial window to both frames.
- Compute the maximum of the Fourier inverse of the Cross-power spectrum between the Fourier of the filtered frames.
- The spatial location of the maximum is the translation vector(magnitude of maximum shows qualitatively the percentage of overlapped region).

and the initial guess for M is:

$$M^{(0)} = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix}$$

Now, starting from this initial transformation matrix, we search for a matrix that yields a minimized intensity error in the overlapped area shared between image 1 and 2. This method has the advantage of not requiring any easily identifiable feature points, and of being statistically optimal once we are in the vicinity of the true solution. We can remove the term ω in Eqn. (2.1) by dividing the first two set of equations by the third equation. Therefore:

$$x_i^{'} = \frac{m_0 x_i + m_0 y_i + m_2}{m_6 x_i + m_7 y_i + 1}, y_i^{'} = \frac{m_3 x_i + m_4 y_i + m_5}{m_6 x_i + m_7 y_i + 1} \tag{2.6}$$

Our technique minimizes the sum of the squared intensities errors:

$$E = \sum_i e_i^2 = \sum_i [I_2(x_i^{'}, y_i^{'}) - I_1(x, y)]^2 \tag{2.7}$$

This sum is calculated over all pixels i which are inside both images $I_1(x, y)$ and $I_2(x^{'}, y^{'})$ (pixels which are outside the overlap region do not contribute) [In the above equation, $x^{'}$ and $y^{'}$ are pixels of the second image mapped to the first image].

To perform this minimization, we use the Levenberg-Marquardt (LM) nonlinear optimization algorithm. To implement this algorithm, the partial derivatives of e_i with respect to $m_0 \cdot \ldots \cdot m_7$ are needed and computed as:

$$\frac{\partial e_i}{\partial m_0} = \frac{x_i}{D_i} \frac{\partial I_2}{\partial x_i^{'}} \cdots \frac{\partial e_i}{\partial m_7} = \frac{-y_i}{D_i} (x_i^{'} \frac{\partial I_2}{\partial x_i^{'}} + y_i^{'} \frac{\partial I_2}{\partial y_i^{'}}) \tag{2.8}$$

We employ a Taylor series expansion to E at $m + \triangle m$. So:

$$E(m + \triangle m) \approx E(m) - \triangle m^T b + \frac{1}{2} \triangle m^T A \triangle m + O(\triangle m^3) \tag{2.9}$$

Where A (Hessian of E) and b (gradient of E) are:

$$[A]_{ij} \equiv \frac{\partial^2 E}{\partial m_i m_j}, b \equiv -\nabla E \tag{2.10}$$

Once the partial derivatives of e_i are calculated, a Hessian matrix A and gradient vector b can be generated as:

$$a_{kl} = \sum_i \frac{\partial e_i}{\partial m_k} \frac{\partial e_i}{\partial m_l}, b_{kl} = -\sum_i e_i \frac{\partial e_i}{\partial m_k} \tag{2.11}$$

Therefore, $\triangle m$ is solved iteratively from:

$$(A + \lambda I) \triangle m = b \tag{2.12}$$

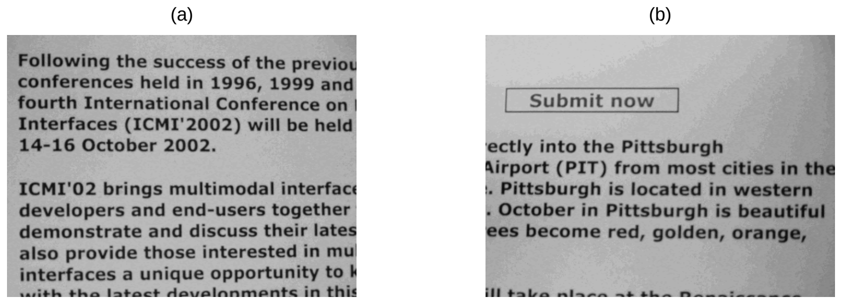
where λ is a tuning parameter that is adjusted according to the change in the sum of squared differences at each iteration.

Once the images are aligned, the next major task is blending the overlapped regions seamlessly. In the mosaicing literature, there are different type of blending routines from simple intensity averaging to more complex Vornoi weights of the images or Gaussian spline interpolations. In our method, for any pixel in the overlapped area, we give more weight to the frame whose center is close to that pixel.

2.2.1 Image Mosaicing Results

Using techniques that are only intensity-based image mosaicing is locally optimal and computationally expensive. So, we use a phase correlation-based technique for coarse estimation of the transformation matrix. The initial guess can be computed by choosing 4 corresponding feature points in each of two consecutive frames. Although this method gives us a statistically very close solution to transformation matrix, it has two disadvantages; it is not automatic, and it fails in case of choosing non-optimal feature points (non optimal choice happens when the corresponding points lie on a line or too close to each other causing the transformation matrix to be ill-conditioned).

While we assume that the relationship between two consecutive frames is a homographic transformation, real camera optical distortions that may cause undesirable results.



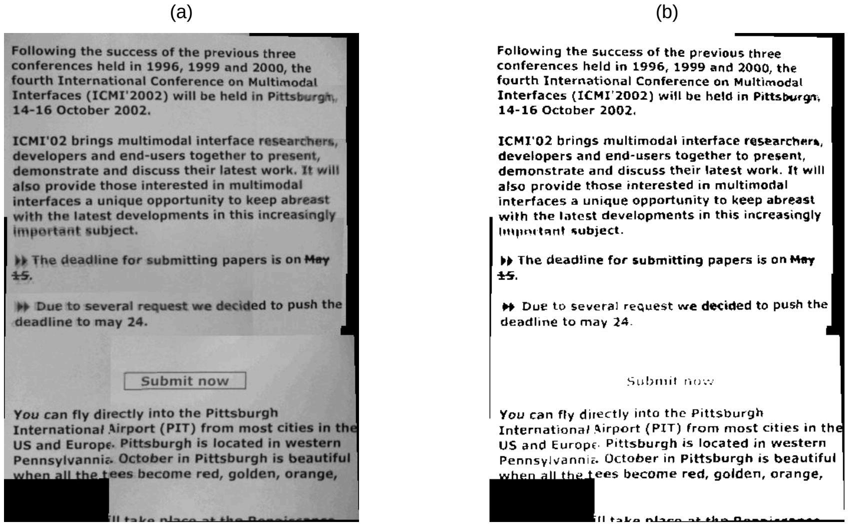
Following the success of the previou
conferences held in 1996, 1999 and
fourth International Conference on I
Interfaces (ICMI'2002) will be held
14-16 October 2002.

ICMI'02 brings multimodal interfac
developers and end-users together
demonstrate and discuss their lates
also provide those interested in mu
interfaces a unique opportunity to k
with the latest developments in this

Submit now

ectly into the Pittsburgh
Airport (PIT) from most cities in the
. Pittsburgh is located in western
. October in Pittsburgh is beautiful
ees become red, golden, orange,

Figure 2.3: a) Frame 1 b) Frame 11



Following the success of the previous three conferences held in 1996, 1999 and 2000, the fourth International Conference on Multimodal Interfaces (ICMI'2002) will be held in Pittsburgh, 14-16 October 2002.

ICMI'02 brings multimodal interface researchers, developers and end-users together to present, demonstrate and discuss their latest work. It will also provide those interested in multimodal interfaces a unique opportunity to keep abreast with the latest developments in this increasingly important subject.

▸▸ The deadline for submitting papers is on ~~May 15~~.

▸▸ Due to several request we decided to push the deadline to may 24.

Submit now

You can fly directly into the Pittsburgh International Airport (PIT) from most cities in the US and Europe. Pittsburgh is located in western Pennsylvannia. October in Pittsburgh is beautiful when all the tees become red, golden, orange,

Following the success of the previous three conferences held in 1996, 1999 and 2000, the fourth International Conference on Multimodal Interfaces (ICMI'2002) will be held in Pittsburgh, 14-16 October 2002.

ICMI'02 brings multimodal interface researchers, developers and end-users together to present, demonstrate and discuss their latest work. It will also provide those interested in multimodal interfaces a unique opportunity to keep abreast with the latest developments in this increasingly important subject.

▸▸ The deadline for submitting papers is on ~~May 15~~.

▸▸ Due to several request we decided to push the deadline to may 24.

Submit now

You can fly directly into the Pittsburgh International Airport (PIT) from most cities in the US and Europe. Pittsburgh is located in western Pennsylvannia. October in Pittsburgh is beautiful when all the tees become red, golden, orange,

Figure 2.4: a) Mosaic image b) binarized and de-skewed mosaic

Figure 2.3 shows the first and eleventh frames from a sequence. The result of image mosaicing algorithm is shown in figure 4(the original number of frames was 132 but we employ several heuristics to reduce this number and increase the computation speed). The super-resolved still image is processed by OCR software and the classification rate is almost $\geq 98\%$. Frames are of 320x240 size and the mosaic is shown in a scaled view.

2.3 Improving Image Quality

2.3.1 Auto-Focusing

In general, focused images have higher frequency components than de-focused images of a scene. In auto-focus techniques the objective is to set the focus of a camera by maximizing high frequency components in an image sequence. For our device we need to search for the best focus in near real time.

Possible criteria for extracting the best focusing position of the lens includes the Tenengrad , sum-modules difference and sum-modified laplacian [39]. The choice of criteria function is crucial to the auto-focusing Algorithm. The criteria function should satisfy the following properties:

- The maximum position of the criteria function corresponds to the best focusing position of lens.
- It is robust to different textures, and works on both outdoor and indoor scenes.
- It has no local maxima to make optimizing the focus difficult.
- It focuses on near and dominant objects closer to the center.

Sum-Modules-Difference (SMD) Criterion is calculated by summing the intensity difference between adjacent pixels:

$$SMD = SMD_x + SMD_y = \sum_i \sum_j |f_{ij} - f_{i,j-1}| + |f_{ij} - f_{i-1,j}|, \tag{2.13}$$

Tenengrad Criterion is simply the magnitude of the gradient at each pixel. If we define horizontal and vertical gradient as f_x and f_y, then:

$$\begin{aligned} Tenengrad &= \textstyle\sum_{\forall x,y} |\nabla f(x,y)| = \sum_{\forall x,y} \sqrt{f_x^2 + f_y^2} \\ f_x = i_x * f(x,y), f_y = i_y * f(x,y) \end{aligned} \qquad (2.14)$$

where i_x and i_y can be any gradient mask. We employed Sobel Operator.

Sum-Modified-Laplacian (SML) estimates high frequency components in an image:

$$SML = |2f(x,y) - f(x-1,y) - f(x+1,y)| + |2f(x,y) - f(x,y-1) - f(x,y+1)| \qquad (2.15)$$

Search Algorithm

We find the in-focus position of the camera lens by determining the maximum of a neighborhood contrast function iteratively searching for the maximum criterion position. We use a parabolic bracketing and hill climbing method, simultaneously [69]. For each iteration we need to get focus values and corresponding focus parameters of a camera. Focus value and focus parameter are H and f, respectively. The steps of the searching algorithm are as follows:

- Initial step: $f_0 = f_{min}$, $f_1 = f_{cur}$, $f_2 = f_{max}$
- Do until Convergence: $|f_{new} - f_{cur}| < tol$

```
    if $H(f_0) < H(f_1) < H(f_2)$ , then:
        $f_{new} \leftarrow$ parabolic Bracketing method
```

```
else if $H(f_0) > H(f_1)$ then:
    $f_{new} \leftarrow f_0 - fstep$
else:
    $f_{new} \leftarrow f_2 + fstep$
pick $f_{new}$ and the other closest two focus values to $f_{new}$
```

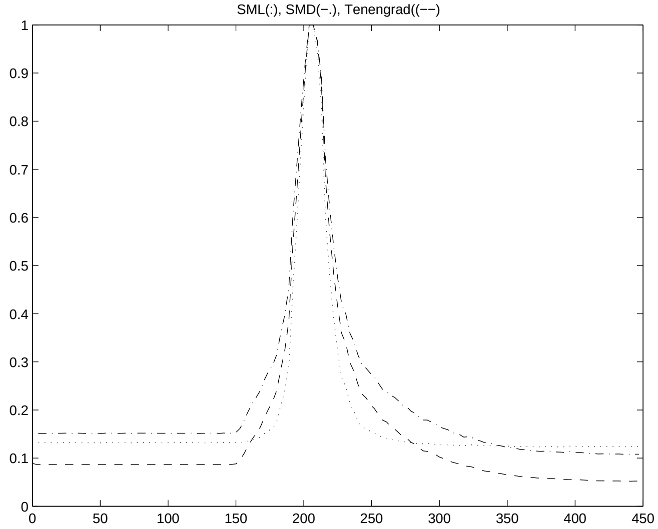
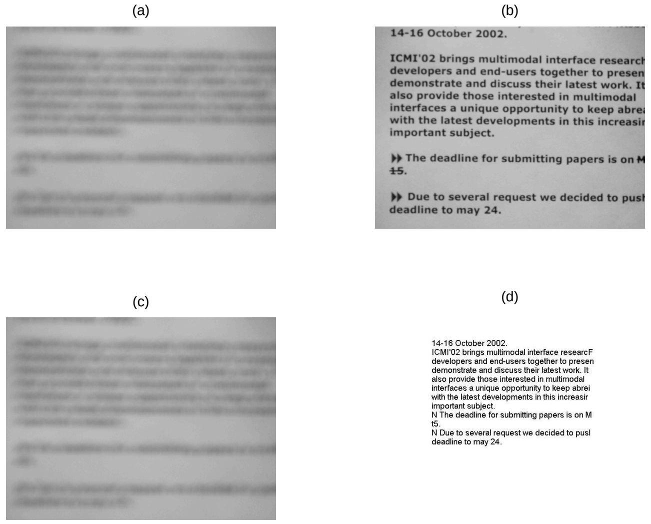


Figure 2.5: Camera characteristic curve for different methods, y-axis: focus value scale to one and x-axis: focus parameter in mm

The best in-focus state of the camera is calculated on dominant edges in a frame. The frames are pre-filtered by a 5x5 Gaussian kernel. We tried all the three for documents and non-document objects and there was no notable difference with respect to speed and accuracy (Fig. 2.5). The in-focus state of a camera is 210mm. All methods find the same focus parameter. We used SMD for Figures 2.6 and 2.7 to show the quality and accuracy of the auto-focusing algorithm.



14-16 October 2002.

ICMI'02 brings multimodal interface research
developers and end-users together to presen
demonstrate and discuss their latest work. It
also provide those interested in multimodal
interfaces a unique opportunity to keep abrea
with the latest developments in this increasir
important subject.

▸▸ The deadline for submitting papers is on ~~M~~
~~15~~.

▸▸ Due to several request we decided to push
deadline to may 24.

```
14-16 October 2002.
ICMI'02 brings multimodal interface researcF
developers and end-users together to presen
demonstrate and discuss their latest work. It
also provide those interested in multimodal
interfaces a unique opportunity to keep abrei
with the latest developments in this increasir
important subject.
N The deadline for submitting papers is on M
t5.
N Due to several request we decided to pusl
deadline to may 24.
```

Figure 2.6: Auto-focus on page: a) de-focus(78mm), b) in-focus(210mm), c) de-focus(366mm), d) OCR results for case(b)

2.3.2 Auto-zooming

We present a methodology to set the zoom parameter of camera based on font-size of text. In general, for low-resolution and narrow field camera, we need to zoom further into sections of the text. If we zoom too little, the resolution of text may be too low and the text will be too small for OCR interpretation. If we zoom too much, characters may become too large for OCR interpretation and moreover, there will be more frames to process.

To find the font-size, we calculate the horizontal projection profile of a de-skewed and binarized image (skew estimation, image thresholding and clean up is done by OCR software). Then,we compute the font-size by finding the average width of pulses on the horizontal projection profile. Based on the font-size calculation, our application automatically zooms to reach the OCR font-size constraint(figure 2.8).

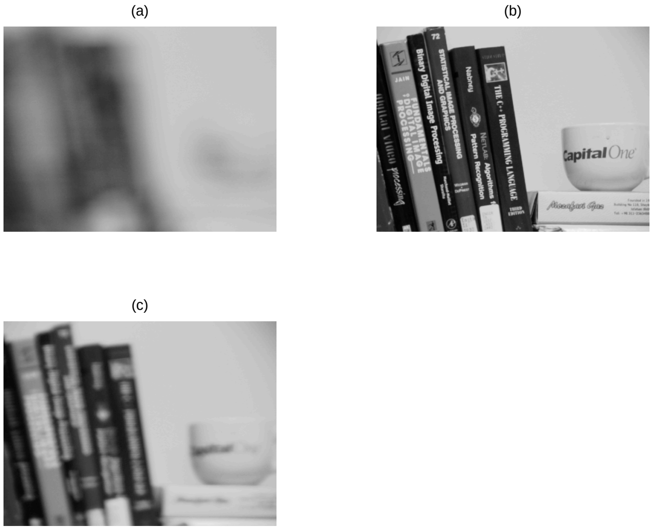
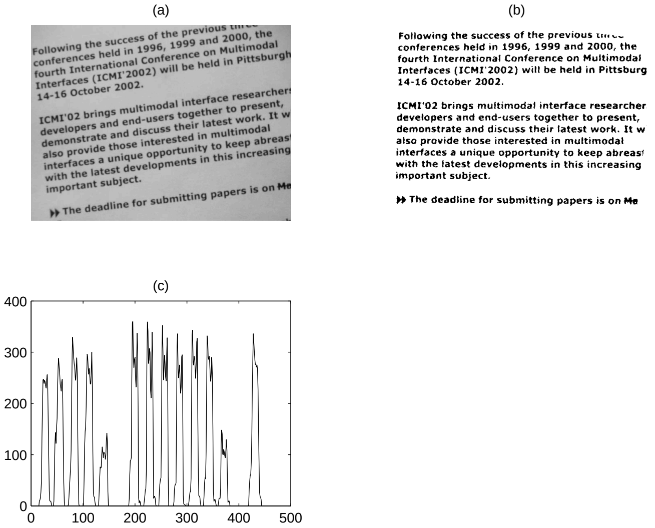


Figure 2.7: Auto-focus on document and non-document objects:a) de-focus (78mm), b) in-focus(318mm), c) de-focus(444mm)

In figure 2.9 $(a - b)$, we show the original image of text processed by the auto-zoom algorithm.

2.4 System Integration and Development

We developed an interface to textual information in the environment for the visually impaired. Our integrated system consists of a digital Camera Sony DFW-VL500, pentium III 866 Mhz computer, loudspeakers. This interface scans document images in the scene and converts them to speech. Our software is written in C++ using MFC library for developing software environment, IPL and OpenCV library for image processing [65], Microsoft Vision SDK for full camera intelligent control(focus, zoom, iris, exposure), Scan soft 2000 for OCR [74], Microsoft Speech SDK for text-to-speech conversion. Fig. 2.10 shows a snapshot of the video-based interface on



Following the success of the previous three
conferences held in 1996, 1999 and 2000, the
fourth International Conference on Multimodal
Interfaces (ICMI'2002) will be held in Pittsburgh
14-16 October 2002.

ICMI'02 brings multimodal interface researchers
developers and end-users together to present,
demonstrate and discuss their latest work. It w
also provide those interested in multimodal
interfaces a unique opportunity to keep abreast
with the latest developments in this increasing
important subject.

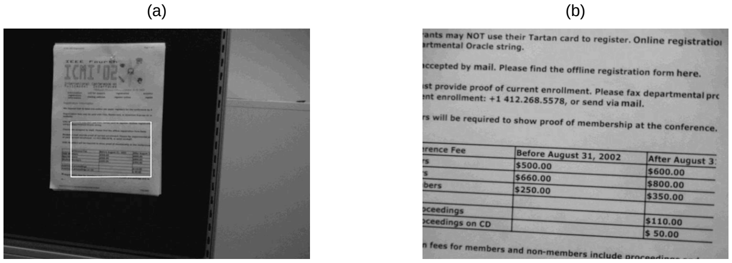
▸▸ The deadline for submitting papers is on Ma

Following the success of the previous three
conferences held in 1996, 1999 and 2000, the
fourth International Conference on Multimodal
Interfaces (ICMI'2002) will be held in Pittsburg
14-16 October 2002.

ICMI'02 brings multimodal interface researcher
developers and end-users together to present,
demonstrate and discuss their latest work. It w
also provide those interested in multimodal
interfaces a unique opportunity to keep abreast
with the latest developments in this increasing
important subject.

▸▸ The deadline for submitting papers is on Ma

Figure 2.8: font-size calculation :a) original image , b) binarized and de-skewed image, c) horizontal projection profile



ants may NOT use their Tartan card to register. Online registratio
artmental Oracle string.

accepted by mail. Please find the offline registration form here.

ist provide proof of current enrollment. Please fax departmental prc
ent enrollment: +1 412.268.5578, or send via mail.

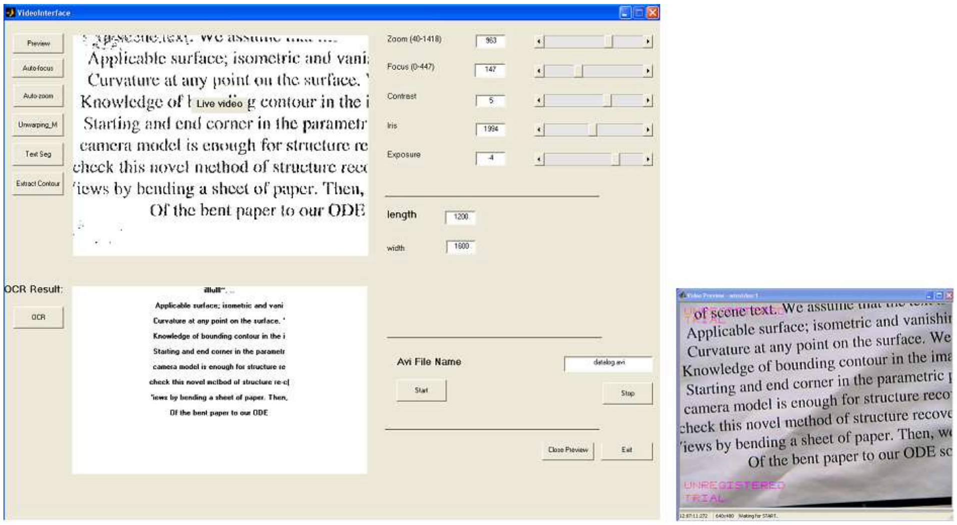
rs will be required to show proof of membership at the conference.

erence Fee	Before August 31, 2002	After August 3:
rs	$500.00	$600.00
s	$660.00	$800.00
bers	$250.00	$350.00
oceedings		$110.00
oceedings on CD		$ 50.00

n fees for members and non-members include proceedings

Figure 2.9: a) original frame b) auto-zoomed frame

the left side and the real-video content. We have similar version of our interface in both the VC++ environment and the MATLAB environment.



Applicable surface; isometric and vani
Curvature at any point on the surface. '
Knowledge of b g contour in the i
Starting and end corner in the parametr
camera model is enough for structure re
check this novel method of structure reco
'iews by bending a sheet of paper. Then,
Of the bent paper to our ODE

Zoom (40-1418) 963
Focus (0-447) 147
Contrast 5
Iris 1994
Exposure -4

length 1200
width 1600

OCR Result:

illull~. ..
Applicable surface; isometric and vani
Curvature at any point on the surface. '
Knowledge of bounding contour in the i
Starting and end corner in the parametr
camera model is enough for structure re
check this novel method of structure re-c|
'iews by bending a sheet of paper. Then,
Of the bent paper to our ODE

Avi File Name datalog.avi

of scene text. We assume that the text
Applicable surface; isometric and vanishi
Curvature at any point on the surface. We
Knowledge of bounding contour in the ima
Starting and end corner in the parametric p
camera model is enough for structure reco
check this novel method of structure recove
'iews by bending a sheet of paper. Then, w
Of the bent paper to our ODE sc

12:07:11.272 640x480 Waiting for START..

Figure 2.10: Left: Video-based interface snapshot, Right: Video-content preview snapshot

Chapter 3

Computer Vision for Planar Scene Text

3.1 Introduction

On a more practical level, one of the chief methods for scientific and business communication is the use of slide shows and posters. Often organizations or individuals record these presentations, but have no means to index or retrieve these digital images by subject. Both these problems need the ability to detect and recognize the layout of text in images, and make sense of the image.

In this chapter we present results from the development of a vision system for the processing of scene text in a relatively restricted context: the processing of images captured in a presentation or a poster session. Our system aims at mapping the layout of a slide or a poster into text and image blocks, performing appropriate rectification, image processing of the text blocks, followed by optical character recognition.

Such a system could be useful to a visually impaired person or for meeting archiving. Text processing algorithms that extract latent semantics [81] have become very powerful. The availability of the text in the presentations (without having access to the digital source slides) can allow these slides to be indexed and retrieved.

3.2 Scenario and Problems

Our goal is to change information from one medium (lecture presentation/slide/poster) to another (text and graph bounding boxes followed by OCR and text-to-speech). Here, we consider that images of slides/posters are taken by a digital camera. These images are composed of text and graphic blocks and background. After image blocks are stored, the rectified text blocks are binarized and passed to OCR software. Finally, we store the detected text and images in a searchable format. Moreover, for recognized texts blocks, we include the content and font-size information. Prior knowledge consists of expected image layout since slides/posters consist of text/graph blocks.

For off-the-shelf OCR software the output of character recognition is reliable only if the text blocks are provided in the fronto-parallel view. In practice, the images are deformed when the optical axis of a camera is not perpendicular to presentation/poster surface. Therefore, the challenge is to extract fronto-parallel views of the deformed image. This is called 'metric rectification'. In a fronto-parallel view, right angles are projected to right angles and parallel lines are projected to parallel lines. Features must be found to perform metric rectification, . Such features are parallel lines and right angles in the image. Hence, in Section 3.4.3 we will introduce an automatic and precise line segment detection algorithm to detect these features. Then, text and image regions are segmented from rectified images. Before providing text boxes to OCR, we pre-process them to improve OCR output quality. Note that all these problems stated are for one image, not an image sequence. In practice, a digital camera takes a video of text printed on a surface. A video contains of a lot of redundant frames with the same information. The problem is how to extract changes in a video due to changes in slides/posters content and not due to illumination change or camera jitter. The schematic of the video-based slides/posters recording framework is shown in Figure 3.1.

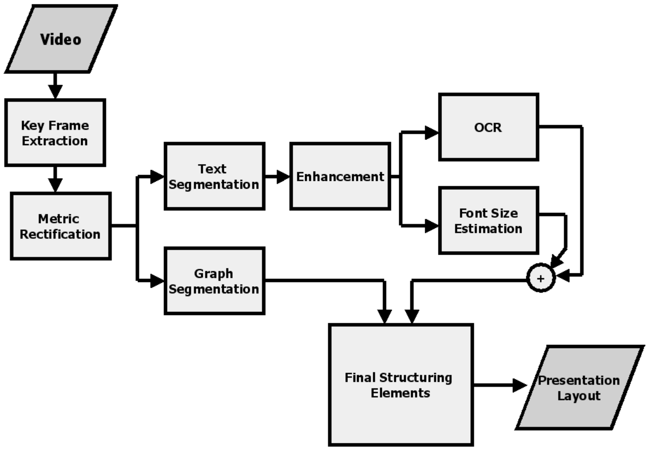


Figure 3.1: Schematic of the system for Annotation and Analysis of Lectures/posters

3.3 Previous work

Camera-based document image analysis is addressed in a recent review article [24]. The following papers touch on problems of video analysis of scene text.

Camera-based acquisition: Ref. [59] addressed a simple scheme for auto-zoom of a camera. This method is useful if the background around an object has low variance compared to the object. Then, in the observation window variance is used as an indicator of best zoom. In [93], a video-based interface to access textual information for the visually impaired was discussed. Auto-focusing and auto-zooming algorithms were presented. The best focus is achieved when the edges are strongest in the image. The best zoom is set when the readable font-size of a text region is more than the OCR readable font-size constraint. We consider this method for preprocessing the real-time recorded video content by controlling the zoom and focus of a camera.

Key-frame extraction: Since in the video of lectures, textual information is not varying rapidly, we need to detect the changes in video and remove redundant frames. In [85], a simple difference operation was introduced. This algorithm is very accurate for still camera pose and constant illumination condition. We used the phase correlation method from the image registration literature to detect the changes in slide or poster video content. This algorithm is stable under global illumination changes and small camera jitter [28, 47].

Metric rectification: The common method in the literature is to extract vanishing lines and right angles in the image [13, 14, 48]. Extraction of vanishing lines is achieved by different methods, such as the projection profile method [14] and using illusory and non-illusory lines in textual layouts [61]. We employ an automatic line segment algorithm for line detection due to Dementhon. We cluster the line segments in feature space (edge angle and edge distance as features)using a mean shift algorithm [17]. We implemented the algorithm of [48, 49] which is suitable for our problem scenario since the image of a poster/slide includes rectangular boxes and lines.

Text segmentation: There are various text segmentation algorithms in the computer vision and document understanding literature that all address the following three basic problems: feature extraction, clustering and validation. For feature extraction, there are different filtering methods: steerable pyramids, Laplacian pyramids [29] and Gabor filters, etc.. Our system uses Gabor filters for the feature extraction part. For clustering, we employ a K-means algorithm, and more generally, a mean shift filter does not require prior knowledge of cluster numbers [17]. In [15], different features from local moments of pixel intensity were used. We use the text segmentation module in [27, 91, 37]. In this paper, we consider the clustering method although in more elaborate and robust methods [50], learning is the choice.

Enhancement: A Global thresholding scheme is not ideal for camera-captured images due to lighting variation and complex background [29]. The survey in [82], compared eleven different adaptive thresholding methods and concluded that Niblack [78] is the best. In this paper, we apply the Niblack method for binarization of text boxes before providing it to OCR.

Contribution: While many of the individual components have been described previously, our contribution is the development of a video-based interface, a unified framework to analyze text and graphs printed in video-lectures and storing them in a searchable format. In [64, 84], a system that supports selection of text in video, and several techniques for segmentation and resolution enhancement of camera images, were described.

3.4 Preprocessing

3.4.1 Key Frame Extraction

Since in video of lectures, textual information is not varying rapidly, many frames will have the same information and we do not want to waste processing resources on the redundant frames. In [85], a simple difference operation was used on three consecutive frames. The difference between two consecutive frames at time t is:

$$FD(t) = \frac{1}{mn}\sum_{\forall x,y} |I(x,y;t+1) - I(x,y;t)| \ , \tag{3.1}$$

where m and n are the pixel dimensions of a frame. Here, we set a frame as key frame, if:

$$|FD(t) - FD(t-1)| > \ e. \tag{3.2}$$

The input to the key frame extraction module is a video and the output is a set of sorted frames in time. This algorithm works extremely well if the camera is still and the same illumination condition holds. It often happens that the illumination varies during the lecture presentation and moreover there is a small camera movement while capturing the content. In this case the simple difference algorithm fails. Our solution is to use phase correlation [47] for key frame extraction. This method , which is well known in the image registration literature, uses the Discrete Fourier Transform (DFT) of two consecutive frames to compute the overlap percentage. Consider two consecutive frames denoted as $f_1 = I(x, y; t)$ and $f_2 = I(x, y; t + 1)$. Denote the DFT of these frames as $F_1(u, v)$ and $F_2(u, v)$. Then the cross power spectrum is:

$$CPS = \frac{F_1(u,v) \cdot F_2^*(u,v)}{|F_1(u,v)||F_2(u,v)|} \tag{3.3}$$

If f_2 is a translated version of f_1, then:

$$f_2(x,y) \approx \alpha f_1(x - \triangle x, y - \triangle y) \tag{3.4}$$

where α is a constant illumination factor. So the CPM is:

$$CPM \approx e^{-j2\pi(u\triangle x + v\triangle y)} \tag{3.5}$$

Therefore, the inverse of the CPM gives an impulse at $(\triangle x, \triangle y)$ and the impulse height is the amount of normalized similarity overlap between $f_1$and f_2(0 corresponds to no overlap and 1 to the maximum area overlap). This method is fast enough for real-time applications and is invariant to constant illumination changes. To suppress the repeating nature of the frequency spectrum and to give less weight to the boundary pixels, we used a raised cosine function, as a window that smoothly reaches 0 at the boundaries. This spatial filter gives more weight to pixels close to center of an image rather than the boundaries. This spatial filter (Hamming cosine window) is formulated in 1D as:

$$w(i) = 0.54 - 0.46\cos(\frac{2\pi}{N}); \quad i = 0 : N - 1 \tag{3.6}$$

Summarizing the key frame extraction method,

1. The first frame is a key frame.
2. While receiving the video sequence for some threshold *tol* (we experimentally choose 0.2), do:

   (a) Apply Hamming cosine window to the previous key frame and the new frame.

   (b) Compute the overlap percentage on the filtered images; if it is less than *tol* then record the new key frame.

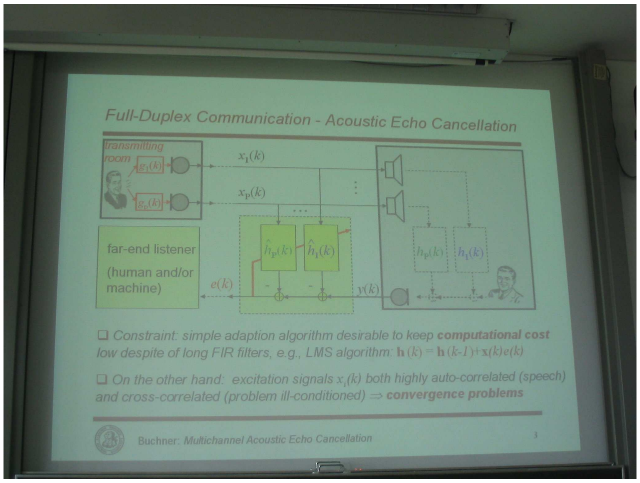
This overlap indicator is extremely efficient and robust for all types of translations and constant illumination changes. At each time step, we keep only two frames in memory and the process is very fast using the FFT (Fast Fourier Transform).

3.4.2 Metric Rectification

An image of a presentation that is not fronto-parallel to the image plane of a camera is deformed due to perspective projection. This distortion is called the *keystone effect*. That means parallel lines and right angles are not projected as parallel lines and right angles in the image plane (Figure 3.2). For planar surfaces the deformation can be modelled by a 3 × 3 matrix, a *'homographic transformation'*, that maps the pixels of the unwarped image to the warped image [26].

$$(u, v) \xmapsto{H} (x, y), \tag{3.7}$$

where H is the homographic mapping, (u, v) is the spatial location of a pixel in the image of fronto-parallel view and (x, y) is the corresponding pixel in the image captured by the camera. Knowledge of at least four corners in the image is enough to estimate the eight unknown parameters of the mapping by least squares estimation algorithms (up to scale). Often we do not have the exact correspondences and also the corners may not be visible. However, we can use the linear features (lines and right angles) in the image for the rectification process. In presentations, lines and boxes in the image provide such linear clues. We address the solution to the keystone correction by estimating vanishing lines and right angles. Before describing the algorithm, we review a few definitions from projective geometry.



Full-Duplex Communication - Acoustic Echo Cancellation

❑ Constraint: simple adaption algorithm desirable to keep **computational cost** low despite of long FIR filters, e.g., LMS algorithm: $\mathbf{h}(k) = \mathbf{h}(k\text{-}1)+\mathbf{x}(k)e(k)$

❑ On the other hand: excitation signals $x_i(k)$ both highly auto-correlated (speech) and cross-correlated (problem ill-conditioned) ⇒ **convergence problems**

Buchner: *Multichannel Acoustic Echo Cancellation* 3

Figure 3.2: Warped image: parallel lines and right angles are not perceived respectively parallel and right angles in the image plane.

Points and lines in homogenous representation: Let $\mathbf{l}$ be a line in 2D plane denoted by $ax + by + c = 0$. A line is represented by $(a, b, c)^T$ and if a point (x, y) on a plane is represented in homogenous coordinates as $\mathbf{x} = (x, y, 1)^T$, then the line equation in homogenous coordinate is $\mathbf{l}^T\mathbf{x} = 0$. The intersection of two lines $\mathbf{l}$ and $\mathbf{l}^{'}$ is the point $\mathbf{x}$; $\mathbf{x} = \mathbf{l} \times \mathbf{l}^{'}$. The line joining two points $\mathbf{x}$ and $\mathbf{x}^{'}$ is $\mathbf{l} = \mathbf{x} \times \mathbf{x}^{'}$. Therefore, lines and points are dual in projective geometry.

Intersection of two parallel lines: Consider two parallel lines $\mathbf{l}$ and $\mathbf{l}^{'}$ with coordinates of $(a, b, c)^T$ and $(a, b, c^{'})^T$. The intersection of two parallel lines is $\mathbf{x} = \mathbf{l} \times \mathbf{l}^{'} = (c^{'} - c)(b, -a, 0)$. Ignoring the scale factor $(c^{'} - c)$, the intersection would be $(b, -a, 0)^T$ which does not belong to R^2. In general, the intersection of two parallel lines, an *ideal point*, is of the form of $(x_1, x_2, 0)^T$. A line at infinity that passes through an ideal point (from Eqn. $\mathbf{l}^T\mathbf{x} = 0$) is represented as $\mathbf{l}_\infty = (0, 0, 1)^T$.

Transformation of lines: If a point $\mathbf{x}'$ is mapped by a matrix H to a point $\mathbf{x}$. Then, we can show that $\mathbf{x}' = H\mathbf{x}$, where H is a 3×3 homographic matrix as:

$$H = \begin{pmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & 1 \end{pmatrix}, \tag{3.8}$$

where all entries are scaled by h_9. Therefore, if a point $\mathbf{x}$ belongs to a line $\mathbf{l}$, then:

$$\mathbf{l}^T\mathbf{x} = 0 \Rightarrow \mathbf{l}^T H^{-1} H\mathbf{x}' = \mathbf{l}^{'T}\mathbf{x}' = 0 \tag{3.9}$$

and consequently line $\mathbf{l}$ is mapped to $\mathbf{l}^{'}$ by a matrix H^{-T}:

$$\mathbf{l}^{'} = H^{-T}\mathbf{l} \tag{3.10}$$

Vanishing points and vanishing line: In a perspective image of a plane, an ideal point is mapped by a homographic transformation H to a vanishing point. The vanishing line is an image of line at infinity in the image plane. Figure 3.3

demonstrates two vanishing points and the vanishing line of a perspectively skewed image. Here, we denote the two spaces: affine skewed space and perspectively skewed space E and F respectively. Therefore, as Eqn. (3.10), we can find a transformation that maps a line at infinity in E to a vanishing line in the image (F).

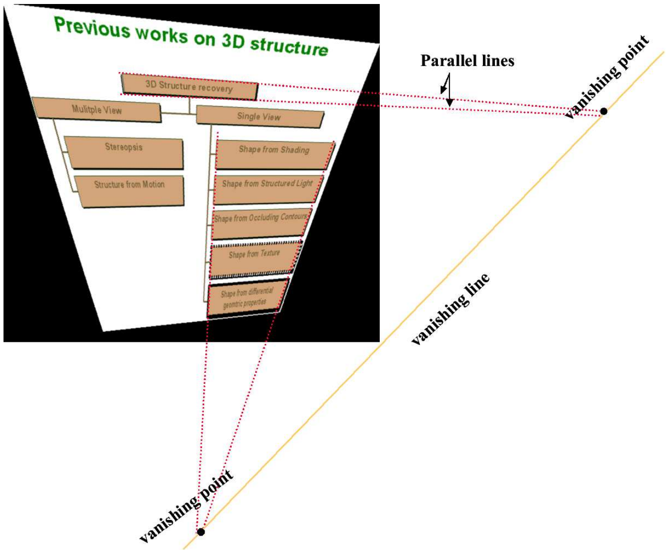


Figure 3.3: Vanishing line and vanishing points.

Decomposition of a projective transformation: It is known that H can be decomposed to S, A and P matrices; Similarity, Affine and Projection matrix [48]. Therefore:

$$\begin{aligned} H &= SAP, \qquad (3.11) \\ &= \begin{pmatrix} sR & t \\ o & 1 \end{pmatrix} \begin{pmatrix} \frac{1}{\beta} & \frac{-\alpha}{\beta} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ l_1 & l_2 & l_3 \end{pmatrix}, \end{aligned}$$

where (l_1, l_2, l_3) is a vanishing line vector in the image plane (F), $R_{2\times 2}$ is the rotation matrix around the image axis of a camera. Therefore, for the metric rectification, we compute $(\alpha, \beta, l_1, l_2, l_3)^T$.

Circular points: are points on the line at infinity which are fixed under any similarity transformation. These points are often called *absolute points* **I** and **J** ; $(1, \pm i, 0)^T$ denoted in the homogenous coordinates. These points are the intersection of any circle with a line at infinity. These points are mapped to $(\alpha \mp \beta i, 1, 0)^T$ on the affine plane (E) by a matrix A and to $((\alpha \mp i\beta)l_3, l_3, -\alpha l_1 \mp i\beta l_1 - l_2)^T$ in the projective plane (F) by a matrix $(A * P)$. Unfortunately, we cannot compute circular points directly because they are complex numbers. Instead we calculate them indirectly through their dual conic representation.

Absolute Conic: It is known that the absolute conic is dual to the circular points as $C^*_\infty = IJ^T + JI^T$ where C^*_∞ is a absolute dual conic.

Rectification Algorithm: Here, we can solve for the metric rectification in two ways. In the first method, we extract vanishing lines and then at least two right angles for the metric rectification. We then compute matrix P from the vanishing line and then A from two right angles. In the second method, we extract five right angles (five pairs of orthogonal lines) and solve for the image of absolute dual conics D in the projective plane. D is denoted as:

$$\begin{gathered} D = MN^T + NM^T, \\ M, N = ((\alpha \mp i\beta)l_3, l_3, -\alpha l_1 \mp i\beta l_1 - l_2)^T \end{gathered} \tag{3.12}$$

where M and N are images of circular points in the projective plane. Each pair of orthogonal lines places a linear constraint on D. From D entries, the 5 known unknown parameters $(\alpha, \beta, l_1, l_2, l_3)^T$are extracted. Based on the angle between lines in projective geometry, we can show that orhtogonal lines are conjugate with respect to D. Each pair of orthogonal lines adds a linear constraint on D:

$$l_a^T D l_b = 0, \tag{3.13}$$

for orthogonal lines l_a and l_b. In 3.4.3, we describe the precise line detection algorithm we use.

In some cases, presentations appear on curved surfaces. These surfaces, applicable surfaces, have special differential geometric properties of vanishing Gaussian curvature at any point and isometry with flat surfaces. We addressed and developed the 3D structure recovery and unwarping of applicable surfaces using differential geometry in [34].

3.4.3 Line Detection

We follow an algorithm suggested by DeMenthon. We compute the edge map of the input image using the Robert operator [29] which is thinned by non-maxima suppression [20]. Then, we make a feature vector with components of edge angles and edge distances. The distance used is that of an edge line segment to the center of the image and the angle is the angle of an edge line segment with respect to the horizontal axis. In feature space, we find the center of clusters using the mean shift algorithm with large mean shift radius of kernel [17]. Then, for each set of pixels with a specific label, relabel each connected component. Now, the angle map and distance map of edges are recomputed and pixels are reclustered with the small kernel radius. At the final stage, we determine the end points of pixels with the same labels. After lines are segmented precisely, the dominant direction of the segmented lines are chosen using the histogram of the segmented line angles. Since lines of different dominant direction are assumed to be orthogonal, so we relabel a pair of orthogonal lines for the metric rectification method either method I or II.

3.4.4 Text Segmentation and Enhancement

In camera-based OCR systems, unlike in scanner-based systems, the image is low in quality and blurred, so that the output of the OCR is poor. The quality of the image is a function of the presentation quality, the camera parameters, camera motion and so on. Here, we assume that the camera is fixed while capturing a video of lectures. Therefore, the challenge is to enhance the image before providing it to OCR. The steps are text segmentation and adaptive binarization.

Treating text as a distinctive texture, we use Gabor filter banks associated with an edge map for text segmentation. The Gabor filter method gives both the benefits of Fourier methods and local spatial distribution methods. The feature responses of the filters at each pixel, are designed to identify text bearing regions. Although none of the filters can individually identify text and non-text regions, a concatenation of the filters provides text detection. This method is robust and precise for text segmentation in natural scenes, text in different size and orientation and complex background. To improve the segmentation results, we will later introduce post-processing algorithms on the output of the text segmentation module. A two dimensional Gabor function $g(x, y)$ in polar coordinates can be written as:

$$g(x, y\ ;\ \sigma_x, \sigma_y, w, \theta) =$$
$$\frac{1}{2\pi\sigma_x\sigma_y}\ \exp\{-\pi(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2})\} \exp\{jw(x\cos(\theta) + y\sin(\theta)\}, \tag{3.14}$$

where σ_x and σ_y are the standard deviations of the Gaussian mask along the x and y directions, w_x and w_y are the center frequencies of the filter, $\theta = \tan^{-1}(\frac{w_y}{w_x})$ is the orientation and $w = \sqrt{w_x^2 + w_y^2}$ is the radial frequency. Gabor functions with different scales and orientations form a complete but nonorthogonal basis set. Expanding an image using this basis provides a localized frequency description. In Figure 3.4, the filters in two scale and four direction are shown. One characteristic of the Gabor filters is its orientation selectivity. Assume the orientation $\theta = \theta^*$, the Gaussian mask filters the image in the θ^* orientation only and blocks other orientations. For the feature extraction part, we choose two scales with four orientations $(0^o, 45^o, 90^o, 135^o)$ at each scale. We implemented the Gabor bank filter from [83]. To increase the precision of the feature extraction part, we choose the magnitude of the responses for each pixel filtered by a nonlinear soft thresholding function of:

$$\Phi(x) = \tanh(\alpha x) = \frac{1 - \exp(-2\alpha x)}{1 + \exp(-2\alpha x)} \tag{3.15}$$

where $\alpha = 0.2$ (experimentally). We associate further a partially redundant feature, a local edge density measure. This feature improves the accuracy and robustness of the method while reducing false detections. Before the clustering step, features are normalized to have zero mean and unit variance [27].

For clustering features in 9D space, we use a K-means clustering algorithm to cluster feature vectors. Empirically, the number of clusters (value of K) was set to three. This value works well with all test images. The cluster whose center is closest to the origin of features vector space is labeled as background (there is no significant edge in any orientation and scale if the background is almost uniform pattern) while the furthest one is labeled as text. If the background is not stationary or highly textured (as often happens in lecture presentation), we could learn the background and subtract it from the key-frame slide. We do not discuss this here.

The output of the clustering is filtered by a median filter to remove small noise due to non-uniformity of the background. Using a morphological operator (closing with disk), we increase the area of text region candidates. Then, we use connected component analysis to label all the text box candidates for the future processing. The final stage of text detection module is a validation module that confirms text boxes. To increase the text segmentation module precision and efficiency, there are few heuristics which are helpful to remove the outlier detected text boxes. We can remove the box if:

1. The OCR output is null.
2. The area of text box is less than some threshold value (This value is empirically set to 100 because OCR can not read text with the width less than 7 pixels and the height less 13 pixels).

Adaptive thresholding processing plays a key role in text image binarization. It is shown in the literature that the global thresholding scheme is not ideal for camera-captured images due to lighting variation and complex background [29]. In the histogram space, the foreground and background density functions are intermixed so a reliable decision boundary (global threshold) cannot be achieved. With a wrong threshold, we will either lose important textual information or add more unwanted edges to the OCR. We implemented the Niblack adaptive thresholding scheme to binarize each text box extracted by text segmentation module [78]. In this algorithm, we compute the local threshold value in a local window as:

$$T(x, y) = M(x, y) + k\sqrt{V(x, y)} \tag{3.16}$$

where $M(x, y)$ and $V(x, y)$ are mean and variance at each local window size w centered at pixel (x, y). The Niblack parameter k is the input parameter to the binarization module. For our system, we set k to -0.2.

3.4.5 Structured Output

In a camera-based presentation analysis framework, we seek an annotating scheme to extract important and compressed information about slides/posters. For the text data embedded in a slide/poster we can recover the font-size of each text box (like the algorithm in [93]) and its spatial location. Therefore, we can sort them in a structured format like a power-point representation e.g. title, text box and graph captions, for each box we record the coordinate, textual content read by OCR and font-size. To find the font-size, we calculate the horizontal projection profile of a binarized text box. Such horizontal profile includes pulses (Figure 3.5). The average font-size is defined as a median over all widths of pulses.

3.5 Implementation Issues and Results

We developed a video-based framework for analysis of presentations 2. We tested the key frame extraction in two ways, simulated and real video sequences. In the simulated version, we initially consider all the slides of a presentation. Then, we randomly add in between the frames by a random generator e.g. after the initial frame 1 we add 14 frames with different random uniform illumination of frame 1 after frame 1. These are the random frame numbers; 14, 20, 16, 18, 14, 12, 18, 19, 14 and 19. The output of this forward simulation is 164 frames. So, we apply the key frame extraction method and we calculated the exact number of key frames which initially was 10. We applied the same method to the video of lectures and posters with the value of 0.2 for the overlapping percentage factor and the results were precise and robust.

The first image in Figure 3.6 portrays a still image image of a poster (one frame extracted from the key frame extraction) module. We applied the automatic metric rectification algorithm described in Section 3.4.2. The OCR output by Scansoft2000 of the rectified image is shown in Figure 3.6. The smaller figures are in-between steps of the automatic metric rectification. We applied the algorithm to different images gone under projection and the algorithm works extremely well if the slide/poster layout consist of text/graph boxes.

In Figure 3.7, we tested the power of our segmentation module for different examples; original color images are on the left and the output of text segmentation module after the morphological operation at right. We gathered different text sizes and orientations on different backgrounds. The first example is for complex backgrounds and highly-textured graphs. The second image is for simple graphs and texts. The third image is for the different text orientations on simple background (Scansoft2000 can process up to 30^o rotation). We convert color images to grayscale images. In all the cases we had the ground-truth and the missing rate was negligible. The text boxes font-size in such cases were more than more than OCR readable font-size. In the first example of Figure 3.7 the rotated text was not readable by the algorithm. The rotated text in gray color space was not clear from the background pattern. These are the main results of our video-based interface:

1. Automatic metric rectification is possible because in lecture presentations/posters we have structured formats like rectangular and lines. This algorithm fails if the necessary information of parallel and orthogonal lines is missed e.g. a slide with one line of text.

2. Key frame extraction is robust and precise under uniform illumination change. It detects major changes in the presentations depending on the *tol* value , with the prescribed value in this paper it can not detect text animation in the slides.

3. The text segmentation module is promising under different text sizes and orientation and complex background.

4. It is assumed that text segmentation is done on the slides which is in fronto-parallel view because segmentation module is placed after the rectification algorithm.

5. The text segmentation is done on grayscale images. The best way to do analysis is to include color information.

6. OCR is sensitive on the text color. We tested on different text colors and if the text was black on white background the results were much more accurate.

7. Our system is capable of reading textual information of lectures/poster videos, detecting the text box coordinates, estimating the font-size in pixels.

In table (3.1), we show the overall performance of each module. The test data is a collection of 50 posters taken by a digital camera Powershot S200 (image size is 2 mega pixel) and 25 presentation videos taken by a digital camera Sony DFW-VL500(frame size 480x640). The hit rate is the correct detection percentage, the false rate is the false detection percentage, and miss rate is the missing percentage.

Module	**hit**	**false**	**miss**
Line detection	96.1%	6.3%	3.9%
Orthogonal line detection	86.2%	16.7%	13.8%
Text segmentation	98.2%	3.3%	1.8%
Key frame extraction	98.6%	2.6%	1.4%

Table 3.1: Quantitative results

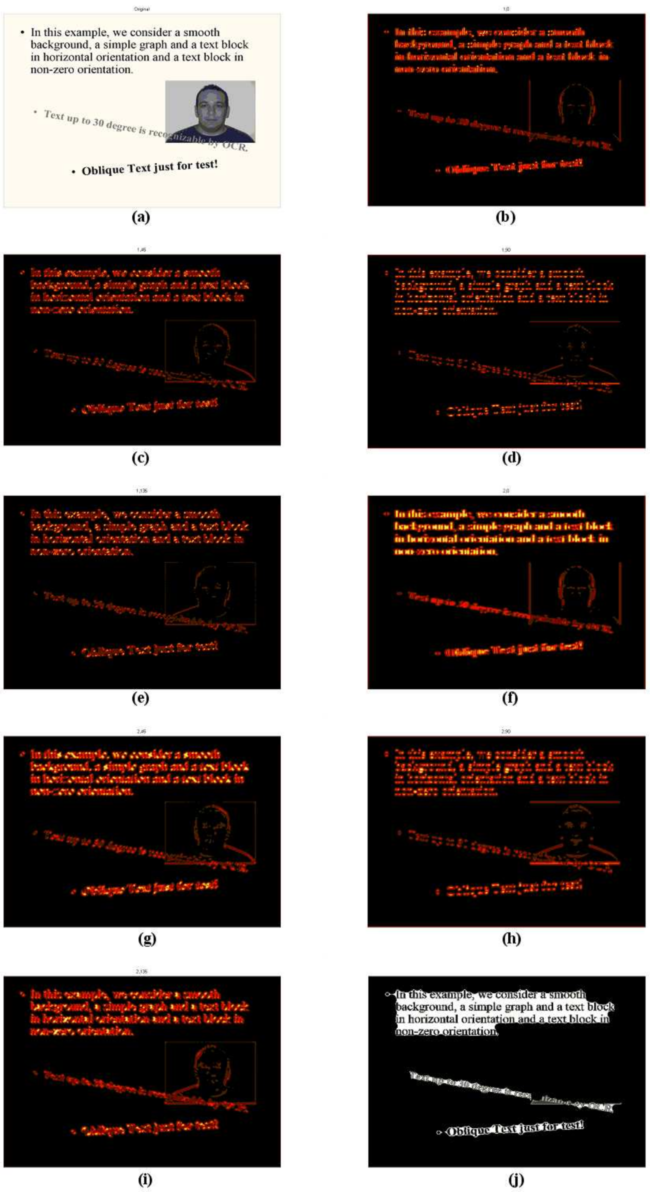
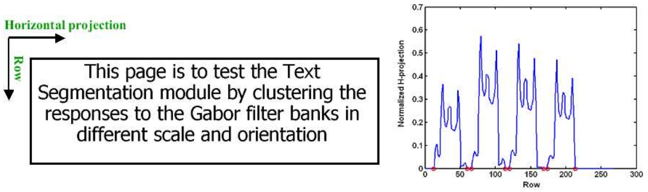
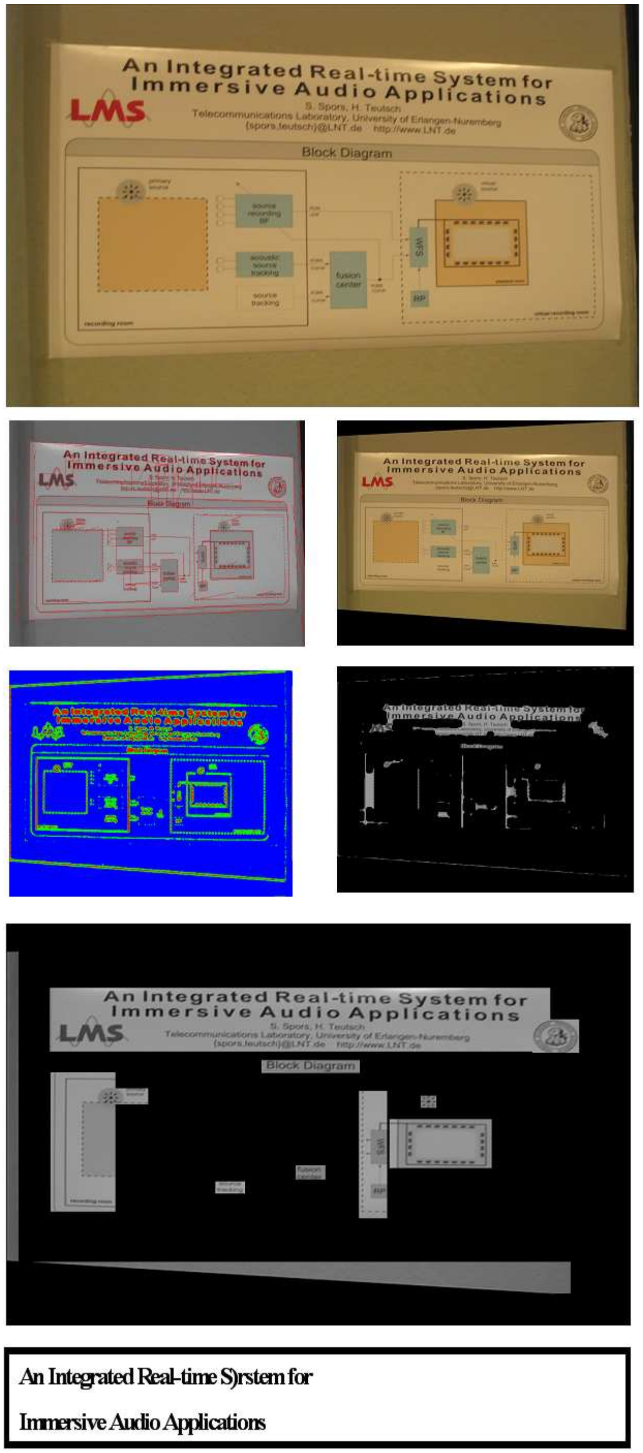


Figure 3.4: Original image and Gabor filter's output for each scale and direction. The color bar for this figure is red for the minimum and yellow for the maximum. a) original image b) $s = 1, o = 0^o$ c) $s = 1, o = 45^o$ d) $s = 1, o = 90^o$ e)$s = 1, o = 135^o$ f) $s = 2, o = 0^o$ g) $s = 2, o = 45^o$ h) $s = 2, o = 90^o$ i)$s = 2, o = 135^o$ j) segmented text regions.



This page is to test the Text Segmentation module by clustering the responses to the Gabor filter banks in different scale and orientation

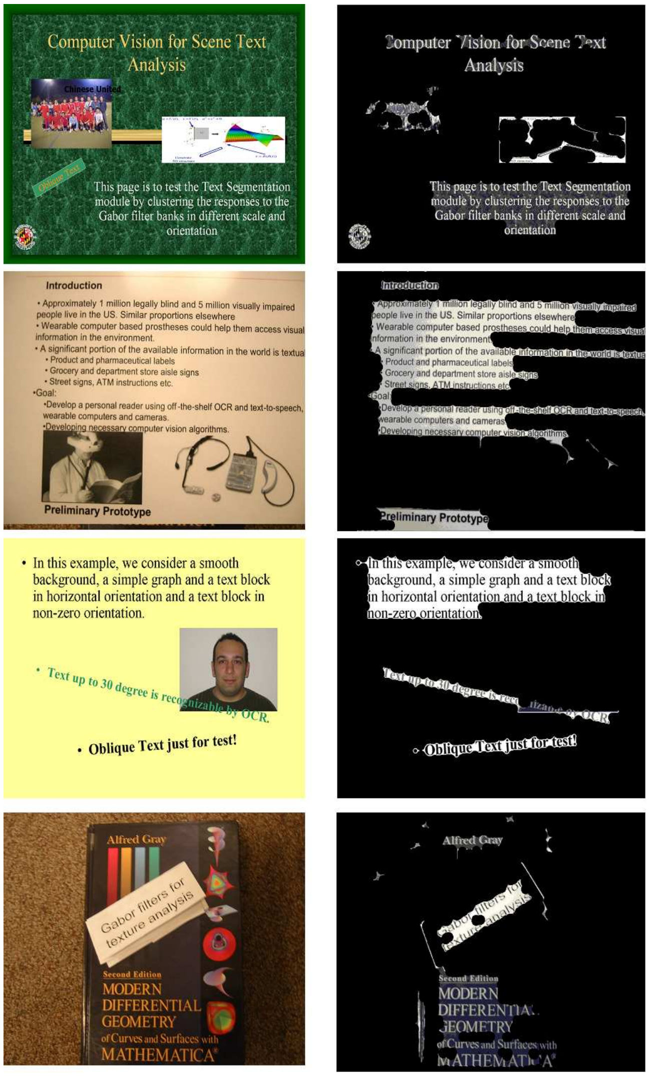
Figure 3.5: Font-size calculation from horizontal projection profile: Figure on left is the binarized text box and Figure on right is the horizontal projection profile of the complement of the image on the left texts are in white and background is in black. Font-size is 24 in pixels.


An Integrated Real-time System for
Immersive Audio Applications
LMS
S. Spors, H. Teutsch
Telecommunications Laboratory, University of Erlangen-Nuremberg
{spors,teutsch}@LNT.de http://www.LNT.de
Block Diagram
source recording
acoustic source tracking
source tracking
fusion center
WFS
RP
recording room
virtual recording room

An Integrated Real-time S)rstem for

Immersive Audio Applications

Figure 3.6: This example is to test the overall algorithm (pictures are (a)-(g) scanned from top-down and left-right). (a) Original extracted frame (b) Detected segmented lines by Mean shift algorithm (c) The rectified image (d) Labeled Image; text-graph-background are represented in RGB (e)Segmented text after morphological operation (f) Text Box Regions (g) OCR output. In this example, the small text



Computer Vision for Scene Text Analysis

Chinese United

Oblique Text

This page is to test the Text Segmentation module by clustering the responses to the Gabor filter banks in different scale and orientation

Introduction

- Approximately 1 million legally blind and 5 million visually impaired people live in the US. Similar proportions elsewhere
- Wearable computer based prostheses could help them access visual information in the environment.
- A significant portion of the available information in the world is textual
  - Product and pharmaceutical labels
  - Grocery and department store aisle signs
  - Street signs, ATM instructions etc.
- Goal:
  - Develop a personal reader using off-the-shelf OCR and text-to-speech, wearable computers and cameras.
  - Developing necessary computer vision algorithms.

Preliminary Prototype

- In this example, we consider a smooth background, a simple graph and a text block in horizontal orientation and a text block in non-zero orientation.
- Text up to 30 degree is recognizable by OCR.
- Oblique Text just for test!

Alfred Gray
Gabor filters for texture analysis
Second Edition
MODERN DIFFERENTIAL GEOMETRY of Curves and Surfaces with MATHEMATICA®

Figure 3.7: In this example, we test the power of text segmentation algorithm for different presentation/poster and outdoor scene text layout. From top-down images are (a)-(d). (a)and (c) are presentation slides with different text size and orientation, different graphs and complex or simple background. (c) is an image of a poster with simple background. (d) is an image of a book on the textured background. The results of text segmentation are shown in the second column.

Chapter 4

3D Structure Recovery and Unwarping Surfaces applicable to Planes

4.1 Introduction

When a picture or text printed on paper is imaged, we are presented with a problem of unwarping the captured digital image to its flat, fronto-parallel representation, as a preprocessing step before performing tasks such as identification, or Optical Character Recognition (OCR). In the case that the paper is flat, the problem reduces to one of undoing a projection of an initial shape such as a rectangle, and the rectification (or unwarping) can be achieved by computing a simple homography. A harder problem is when the piece of paper is itself deformed or bent. In this case the unwarping must undo both the effects of the three-dimensional bending of the surface, and the imaging process. The differential geometry of surfaces provides a very powerful set of relations for analysis of the unwarping. However, most quantitative use of differential geometry has been restricted to range data, while its use for image data has been primarily qualitative. The deformation of paper surfaces satisfies the conditions of isometry and vanishing Gaussian curvature. Here, we show that these conditions can be analytically integrated to infer the complete 3D structure of the surface from an image of its bounding contour.

Previous authors have attempted to enforce these conditions in 3D reconstruction. However, they essentially enforced these as *constraints* to a process of polynomial/spline fitting using data obtained on the surface [62]. In contrast, we *solve* these equations, and show that *information on the bounding contour is sufficient to determine structure completely.* Further, exact correspondence information along the bounding contour is not needed. We only need the correspondences of a few points, e.g., corners. Other than its theoretical importance, our research can potentially benefit diverse computer vision applications, e.g. portable scanning devices, digital flattening of creased documents, 3D reconstruction without correspondence, and perhaps most importantly, OCR of scene text.

4.2 Previous Work

A seminal paper by Koenderink [42] addressed the understanding of 3D structure qualitatively from occluding contours in images. It was shown that the concavities and convexities of visual contours are sufficient to infer the local shape of a surface. Here, we perform quantitative recovery of 3D surface structure for the case of applicable surfaces. While we were not able to find similar papers dealing with analytical integration of the equations of differential geometry to obtain structure, the following papers deal with related problems of unwarping scene text, or using differential geometric constraints for reconstruction.

Metric rectification of planar surfaces: In [14, 48, 61] algorithms for performing metric rectification of planar surfaces were considered. These papers extract from the images, features such as vanishing lines and right angles and perform rectification. Extraction of vanishing lines is achieved by different methods; such as the projection profile method [14] and the illusory and non-illusory lines in textual layouts [61].

Undoing paper curl for non-planar surfaces knowing range data: A number of papers deal with correcting the curl of documents using known shape (e.g. cylinders) [40, 92]. These approaches all need 3D points on the surface to solve for the inverse mapping. In [62] sparse 3D data on the curled paper surface was obtained from a laser device. An approximate algorithm to fit an applicable surface through these points was developed that allowed obtaining dense depth data. The isometry constraint was approximately enforced by requiring that distances between adjacent nodes be constant. In [10] a mass-spring particle system framework was used for digital flattening of destroyed documents using depth measurements, though the differential geometry constraints are not enforced.

Isometric mapping: In [41] an algorithm is developed to bend virtual paper without shearing or tearing. Ref. [60] considers the shape-from-motion problem for shapes deformed under isometric mapping.

4.3 Theory

4.3.1 Basic Surface Representation

A surface is the exterior boundary of an object/body. In a 3D world coordinate system, a surface $\mathbf{r} = \mathbf{r}(X, Y, Z)$, (where (X, Y, Z) is any point on the surface) is mathematically represented in explicit, implicit and parametric forms respectively as:

$$z = f(x, y), \quad F(x, y, z) = 0, \quad \mathbf{r}(u, v) = (X(u, v), Y(u, v), Z(u, v)). \tag{4.1}$$

Consider a smooth surface S expressed parametrically as:

$$\mathbf{r}(u,v) = (X(u, v), Y(u, v), Z(u, v)), \tag{4.2}$$

which is a mapping from any point (u, v) in the parametric (or undeformed) plane (uv-plane) to a point (X, Y, Z) on the surface in 3D (Figure 3). The sets $\{\mathbf{r}(u, v),\ v = const\}$ and $\{\mathbf{r}(u, v),\ u = const\}$ represent two families of curves on the surface, whose partial derivatives are tangent vectors to the curves $v = const$ and $u = const$

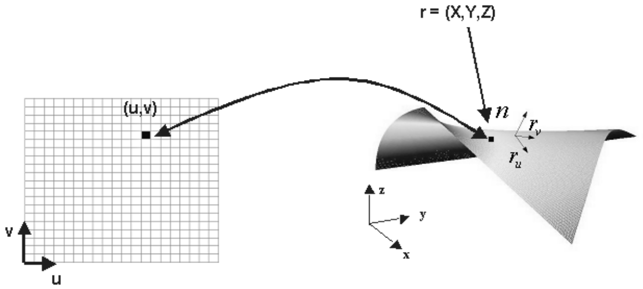


Figure 4.1: Parametric representation of a surface

respectively. These derivatives are often called *tangent vectors* [44]. Let the second derivatives of $\mathbf{r}$ with respect to u and v be $\mathbf{r}_{uu}$, $\mathbf{r}_{uv}$ and $\mathbf{r}_{vv}$. The element of distance $ds = |d\mathbf{r}|$ on the surface is given at each surface point (u, v) by the *first fundamental form* of a surface

$$ds^2 = |d\mathbf{r}|^2 = ||\mathbf{r}_u||^2 du^2 + 2\mathbf{r}_u \cdot \mathbf{r}_v \, dudv + ||\mathbf{r}_v||^2 dv^2 = E \, du^2 + 2F \, dudv + G \, dv^2, \tag{4.3}$$

$$E(u,v) = ||\mathbf{r}_u||^2, \quad F(u,v) = \mathbf{r}_u \cdot \mathbf{r}_v, \quad G(u,v) = ||\mathbf{r}_v||^2.$$

The surface coordinates are orthogonal iff $F \equiv 0$. The surface normal $\mathbf{n}$ and area element $d\mathbf{n}$ can be defined in terms of the tangent vectors as:

$$\mathbf{n} = \frac{\mathbf{r}_u \times \mathbf{r}_v}{|\mathbf{r}_u \times \mathbf{r}_v|} = \sqrt{EG - F^2}, \quad d\mathbf{n} = |\mathbf{r}_u \times \mathbf{r}_v| \, dudv = \sqrt{EG - F^2} \, dudv. \tag{4.4}$$

The *second fundamental* form of a surface at a point (u, v) measures how far the surface is from being planar. It is given by

$$-d\mathbf{r} \cdot d\mathbf{n} = L(u,v)du^2 + 2M(u,v)dudv + N(u,v)dv^2, \tag{4.5}$$

where L, M and N are defined as[44]:

$$\begin{aligned} L(u,v) &= -\mathbf{r}_u \cdot \mathbf{n}_u = \mathbf{r}_{uu} \cdot \mathbf{n}, \\ M(u,v) &= -\mathbf{r}_u \cdot \mathbf{n}_v = \mathbf{r}_{uv} \cdot \mathbf{n}, \\ N(u,v) &= -\mathbf{r}_v \cdot \mathbf{n}_v = \mathbf{r}_{vv} \cdot \mathbf{n}. \end{aligned} \tag{4.6}$$

For every normal section through (u, v) there exist two principal curvatures (k_1, k_2). The mean and Gaussian curvature; $H(u, v)$ and $K(u, v)$ are

$$H \equiv \frac{k_1 + k_2}{2} = \frac{1}{2} \frac{EN - 2FM + GL}{EG - F^2}, \quad K \equiv k_1 k_2 = \frac{LN - M^2}{EG - F^2}. \tag{4.7}$$

4.3.2 Special Surfaces

Let us assume that we have a mapping of a point in the parametric plane (u, v) to a point in 3D (X, Y, Z). The mapping is *isometric* if the length of a curve or element of area is invariant with the mapping, i.e.

$$E(u, v) = ||\mathbf{r}_u||^2 = 1, \quad F(u, v) = \mathbf{r}_u \cdot \mathbf{r}_v = 0, \quad G(u, v) = ||\mathbf{r}_v||^2 = 1. \tag{4.8}$$

Lengths and areas are conserved in an isometric mapping

$$\begin{gathered} ds^2 = |d\mathbf{r}|^2 = E(u, v)du^2 + 2F(u, v)dudv + G(u, v)dv^2 = du^2 + dv^2, \\ dA = \sqrt{EG - F^2}\, dudv = dudv. \end{gathered} \tag{4.9}$$

The mapping is *conformal* if the angle between curves on a surface is invariant of the mapping $(F = 0)$. It is *developable* if the Gaussian curvature is zero everywhere.

$$K = 0 \implies LN - M^2 = 0. \tag{4.10}$$

It is *applicable to a flat surface* if the surface is isometric with a flat surface (Eq. 4.8) and the Gaussian curvature vanishes (Eq. 4.10) for every point on the surface. It is stated in ([43]) that two surfaces are applicable if the first fundamental form that embodies the metric, the Gaussian curvature are conserved. Here, we denote a surface applicable to a flat surface as an applicable surface.

4.3.3 Differential Equations for a surface applicable to a plane

If we differentiate the first and third equations at (4.8) with respect to u and v, we have:

$$\mathbf{r}_{uu} \cdot \mathbf{r}_u = \mathbf{r}_{uu} \cdot \mathbf{r}_v = \mathbf{r}_{uv} \cdot \mathbf{r}_u = \mathbf{r}_{uv} \cdot \mathbf{r}_v = \mathbf{r}_{vv} \cdot \mathbf{r}_u = \mathbf{r}_{vv} \cdot \mathbf{r}_v = 0. \tag{4.11}$$

This shows that $\mathbf{r}_{uu} = (X_{uu}, Y_{uu}, Z_{uu})$, $\mathbf{r}_{uv} = (X_{uv}, Y_{uv}, Z_{uv})$ and $\mathbf{r}_{vv} = (X_{vv}, Y_{vv}, Z_{vv})$ are perpendicular to $\mathbf{r}_u$ and $\mathbf{r}_v$ and consequently, are collinear with the normal vector to the surface.

$$\mathbf{n} \parallel (\mathbf{r}_u \times \mathbf{r}_v) \; || \; \mathbf{r}_{uu} \parallel \mathbf{r}_{uv} \parallel \mathbf{r}_{vv}, \tag{4.12}$$

where $||$ denotes "is parallel to". We can thus express $\mathbf{n}$ as

$$\mathbf{n} = a\mathbf{r}_{uu} = b\mathbf{r}_{uv} = c\mathbf{r}_{vv}. \tag{4.13}$$

Theorem 1: For a surface $\mathbf{r} = \mathbf{r}(u, v)$ isometric with a plane, a normal vector $\mathbf{n}$ is parallel to the derivatives of tangent vectors $\mathbf{r}_u$ and $\mathbf{r}_v$ in the u and v direction $(\mathbf{r}_{uu}, \mathbf{r}_{uv}, \mathbf{r}_{vv})$.

We can rewrite (4.10) using (4.13) as:

$$LN - M^2 = 0 \quad \Longrightarrow (\mathbf{r}_{uu} \cdot \mathbf{n})(\mathbf{r}_{vv} \cdot \mathbf{n}) - (\mathbf{r}_{uv} \cdot \mathbf{n})^2 \quad = 0, \tag{4.14}$$

then,

$$a||\mathbf{n}||^2 c||\mathbf{n}||^2 - b^2||\mathbf{n}||^2||\mathbf{n}||^2 = 0 \quad \Longrightarrow \quad ac - b^2 = 0, \tag{4.15}$$

where a, b, and c are scalars, and

$$\frac{\mathbf{r}_{uv}}{\mathbf{r}_{uu}} = \frac{a}{b} = \frac{b}{c} = \frac{\mathbf{r}_{vv}}{\mathbf{r}_{uv}}. \tag{4.16}$$

Therefore from (4.16) we have:

Theorem 2: For a surface $\mathbf{r} = \mathbf{r}(u, v) = (X(u, v), Y(u, v), Z(u, v))$ *applicable to a plane, there is a nonlinear higher order paritial differentail equations governing the surface as follows:*

$$\frac{\partial^2 W}{\partial v^2}\frac{\partial^2 W}{\partial u^2} = \left(\frac{\partial^2 W}{\partial u \partial v}\right)^2, \quad \text{for} \quad W = X, Y, Z. \tag{4.17}$$

Solving the set of nonlinear higher order partial differential equations (PDEs) (Eq. 4.17), we can compute the surface structure $\mathbf{r}$ in 3D, given boundary conditions (curves) for an applicable surface. These equations may be solved by conventional methods of solving PDEs e.g. Finite Differences or FEM. However, we provide a much more efficient method, based on reducing the solution to integration of several simultaneous ODEs.

4.3.4 A First Integration: Reduction to ODEs

Let $W_u = \partial W / \partial u$, $W_v = \partial W / \partial v$. The functions $W_u(u, v)$ and $W_v(u, v)$ satisfy the consistency conditions

$$\frac{\partial W_u}{\partial v} = \frac{\partial W_v}{\partial u}, \qquad W = X, Y, Z. \tag{4.18}$$

i.e. cross-derivatives are the same. From Eqs. (4.17) and (4.18) we have

$$\frac{\partial W_u}{\partial u}\frac{\partial W_v}{\partial v} - \frac{\partial W_u}{\partial v}\frac{\partial W_v}{\partial u} = \frac{\partial (W_u, W_v)}{\partial (u, v)} = 0. \tag{4.19}$$

Therefore Eq. (4.19) can be treated as a degeneracy condition for the Jacobian of the mapping from $(u, v) \longmapsto (W_u, W_v)$. This degeneracy means that the functions

W_u and W_v are functions of a single variable, t, which in turn is a function of (u, v). In other words:

$$\exists\, t \; = t(u,v) \text{ such that } W_u\left(u,v\right) = W_u\left(t\right), \quad W_v\left(u,v\right) = W_v\left(t\right), \tag{4.20}$$

where $W = X, Y, Z$. In this case $t = const$ is a line in the parametric plane. Since W denotes any of X, Y and Z, Eq. (4.20) could hold separately for each component, with some different mapping functions $t_x(u,v)$, $t_y(u,v)$, and $t_z(u,v)$ specific to each coordinate. However, these functions must all be equal because all are functions of the single variable $t(u,v)$, which can be called the *mapping* or *characteristic function* for the surface S.

Theorem 3: For a surface $\mathbf{r} = \mathbf{r}(u,v)$ *applicable to a plane there exists a characteristic line t=t(u,v) that tangent vectors of the surface are functions of it.*

$$\mathbf{r}_u = \mathbf{r}_u\left(t\right), \quad \mathbf{r}_v = \mathbf{r}_v\left(t\right), \tag{4.21}$$

where $t = t(u,v)$. Denoting by the superscript dot the derivative of a function with respect to t, we can write $\mathbf{r}_{uu}$ and $\mathbf{r}_{vv}$ as

$$\mathbf{r}_{uu} = \dot{\mathbf{r}}_u \frac{\partial t}{\partial u}, \quad \mathbf{r}_{vv} = \dot{\mathbf{r}}_v \frac{\partial t}{\partial v}. \tag{4.22}$$

From Eqns. (4.12) and (4.22), we see that $\dot{\mathbf{r}}_u$ and $\dot{\mathbf{r}}_v$ are collinear with the surface normal i.e. $\dot{\mathbf{r}}_u || \mathbf{n}$, $\dot{\mathbf{r}}_v || \mathbf{n}$. Let us define a new vector $\mathbf{w}$ as :

$$\mathbf{w} = u\dot{\mathbf{r}}_u\left(t\right) + v\dot{\mathbf{r}}_v\left(t\right). \tag{4.23}$$

Also note that $\mathbf{w}$ is a function of the characteristic variable t, since the Jacobian of a mapping from $(u,v) \longmapsto (t, \mathbf{m}\cdot\mathbf{w})$ for a constant vector $\mathbf{m}$ vanishes:

$$\frac{\partial\left(t,\mathbf{w}\cdot\mathbf{m}\right)}{\partial\left(u,v\right)} = \frac{\partial t}{\partial u}\frac{\partial \mathbf{w}\cdot\mathbf{m}}{\partial v} - \frac{\partial t}{\partial v}\frac{\partial \mathbf{w}\cdot\mathbf{m}}{\partial u} = \frac{\partial t}{\partial u}\dot{\mathbf{r}}_v\left(t\right)\cdot\mathbf{m} - \frac{\partial t}{\partial u}\dot{\mathbf{r}}_{\mathbf{v}}\left(\mathbf{t}\right)\cdot\mathbf{m}$$

$$= \mathbf{r}_{uv}\cdot\mathbf{m} - \mathbf{r}_{uv}\cdot\mathbf{m} \quad \Longrightarrow \quad \frac{\partial\left(t,\mathbf{w}\cdot\mathbf{m}\right)}{\partial\left(u,v\right)} = 0. \tag{4.24}$$

This means that $\mathbf{w}$ is a function of t alone; $\mathbf{w} = \mathbf{w}\left(t\right)$. From collinearity of $\mathbf{w}$ with $\dot{\mathbf{r}}_u$ and $\dot{\mathbf{r}}_v$ it follows that two scalar functions $h_u\left(t\right)$ and $h_v(t)$ can be introduced as

$$\dot{\mathbf{r}}_u\left(t\right) = h_u\left(t\right)\mathbf{w}\left(t\right), \quad \dot{\mathbf{r}}_v\left(t\right) = h_v(t)\mathbf{w}\left(t\right) \Rightarrow h_v(t)\dot{\mathbf{r}}_u\left(t\right) - h_u\left(t\right)\dot{\mathbf{r}}_v\left(t\right) = 0. \tag{4.25}$$

By substituting Eqn. (4.25) in Eqn. (4.23),

$$\mathbf{w}(t) = uh_u\left(t\right)\mathbf{w}\left(t\right) + vh_v(t)\mathbf{w}\left(t\right) \Rightarrow uh_u\left(t\right) + vh_v(t) = 1. \tag{4.26}$$

therefore,

$$uh_u\left(t\right) + vh_v(t) = 1, \quad h_v(t)\dot{\mathbf{r}}_u\left(t\right) - h_u\left(t\right)\dot{\mathbf{r}}_v\left(t\right) = 0. \tag{4.27}$$

Therefore, Eq.(4.27) defines a characteristic line in the uv-plane for $t = const$. While the latter equation provides a relation between functions of t, the former implicitly determines $t\left(u,v\right)$. Once $h_u\left(t\right)$ and $h_v(t)$ are known, Eq. (4.27) gives $t\left(u,v\right)$. Note that t satisfies the equation

$$h_v\left(t\right)\frac{\partial t}{\partial u} - h_u\left(t\right)\frac{\partial t}{\partial v} = 0, \tag{4.28}$$

which is a *Hopf* equation, a common nonlinear hyperbolic equation in shock-wave theory [87]. The characteristics of this equation are $t = t\left(u,v\right)$ which satisfies

$$t\left(u,v\right) = t\left(u + c(t)v\right), \quad c(t) = \frac{h_u\left(t\right)}{h_v\left(t\right)}. \tag{4.29}$$

Therefore, for any $t = const$ the characteristic is a line in the uv-plane. The properties of the Hopf equation are well studied in the theory of propagation of shock waves in nonlinear media [87]. Along the characteristics, $t = t(u, v) = const$, all functions of t are constant, including $h_u(t)$ and $h_v(t)$. As follows from Eq. (4.27), in the (u, v)-plane these characteristics are straight lines. The lines corresponding to characteristics are also straight lines on the surface. In fact to generate an applicable surface, we can sweep a line in space and the generated envelope will be an applicable surface to a flat surface. Through every point on the surface there is a straight line as shown (Figure 4.2) by:

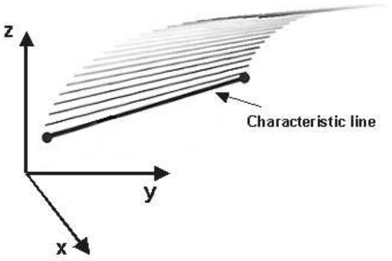


Figure 4.2: Characteristics lines as generator lines

$$\mathbf{r}(t) = u\mathbf{r}_u(t) + v\mathbf{r}_v(t) + \rho(t) \quad , \quad \dot{\rho}(t) = -\mathbf{w}(t), \tag{4.30}$$

Theorem 4: for an applicable surface, characteristic lines in the parametric plane are lines in 3D.

Proof: Consider a characteristic line $t = t^*$. Therefore, the tangent vectors $(\mathbf{r}_u(t^*), \mathbf{r}_v(t^*))$ and $\rho(t^*)$ are constant vectors. Also, any point (u, v) on the characteristic line $t = t(u, v) = t^*$ lies in:

$$uh_u(t^*) + vh_v(t^*) = 1, \tag{4.31}$$

where $h_u\left(t^*\right)$ and $h_v\left(t^*\right)$ are constant scalars. Therefore,

$$\mathbf{r}\left(t^*\right)=u\mathbf{r}_u\left(t^*\right)+\frac{\left(1-uh_u\left(t^*\right)\right)}{h_v\left(t^*\right)}\mathbf{r}_v\left(t^*\right)+\rho\left(t^*\right)\quad, \tag{4.32}$$

so we can rewrite it as:

$$\mathbf{r}=u\mathbf{q}+\mathbf{q}_0\ , \tag{4.33}$$

where $\mathbf{q}_0=\frac{1}{h_v(t^*)}\mathbf{r}_v\left(t^*\right)+\rho\left(t^*\right)$ and $\mathbf{q}=\mathbf{r}_u\left(t^*\right)-\frac{h_u(t^*))}{h_v(t^*)}\mathbf{r}_v\left(t^*\right).$ Recall that this is the line equation in 3D.

The above equations are sufficient to solve the basic warping and unwarping problems for images based on information about the shapes of the image boundaries. The goal is to find for any characteristic line, the variables $\mathbf{r}_u\left(t\right), \mathbf{r}_v\left(t\right), \rho\left(t\right), h_u\left(t\right)$ and $h_v\left(t\right)$ and, finally, $\mathbf{r}\left(t\right)$ from available information. To summarize the differential and algebraic relations for applicable surfaces, we have

$$\begin{gathered}
\mathbf{r}\left(u,v\right)=u\mathbf{r}_u\left(t\right)+v\mathbf{r}_v\left(t\right)+\rho\left(t\right),\\
\dot{\mathbf{r}}_u\left(t\right)=h_u\left(t\right)\mathbf{w}\left(t\right),\\
\dot{\mathbf{r}}_v\left(t\right)=h_v(t)\mathbf{w}\left(t\right),\\
\dot{\rho}\left(t\right)=-\mathbf{w}\left(t\right),\\
uh_u\left(t\right)+vh_v(t)=1,\\
||\mathbf{r}_u||^2=1,\ \ \mathbf{r}_u\cdot\mathbf{r}_v=0,\ \ ||\mathbf{r}_v||^2=1.
\end{gathered} \tag{4.34}$$

4.3.5 Forward Problem: Surface with a Specified Boundary Curve

Here, we specify the bending of a flat page in 3D so that one edge conforms to a given 3D curve. We call this the forward problem. We generate the warped surface to demonstrate the solution to Eq. (4.34). Let Γ'be an open curve on a patch $\Omega' \subset P$ in the uv-plane, corresponding to an open curve Γ in 3D. To generate an applicable surface in 3D, knowledge of the corresponding curves Γ' and Γ and the patch boundaries in the uv-plane (Figure 4.3) are sufficient. We know that the curve Γ' starts from a point $A' = (u_0, v_0)$ and the corresponding curve Γ passes from $A = (X_0, Y_0, Z_0)$ and the point B corresponds to the point B'. Due to isometry, the lengths of the two curves are the same, and there is a one-to-one mapping from a domain $\Omega' \subset P$ to $\Omega \subset S$, which are respectively bounded by Γ' and Γ. For any point $(u^*, v^*) \in \Omega'$ there exists a characteristic, $t = t^*$, which also passes through some point on Γ'. Assume now that Γ' is specified by the parametric equations

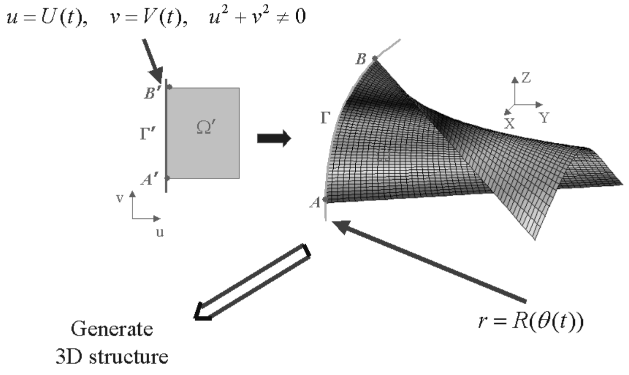


Figure 4.3: Generation of an applicable surface with a 3D curve. In this example a straight line Γ' in the uv-plane is mapped on a given 3D curve Γ.

$$u = U\left(t\right), \quad v = V(t), \quad u^2 + v^2 \neq 0.$$

Without loss of generality, we can select t to be a natural parameterization of Γ' , measured from point A'; i.e. the arc length s along the curve Γ, measured from the curve starting point $t = t_0$,

$$s \equiv \int_{t_0}^{t} ds \equiv \int_{t_0}^{t} \sqrt{d\mathbf{r}.d\mathbf{r}}. \tag{4.35}$$

parameterizes the curve. Let $\Gamma' : (U(t), V(t))$ be in $[t_{\min,}t_{\max}]$. If we represent Γ in parametric form as $\mathbf{r} = \mathbf{R}(t)$, then due to isometry, t will also be a natural parameter for Γ', and

$$\dot{U}^2 + \dot{V}^2 = 1, \quad \dot{\mathbf{R}} \cdot \dot{\mathbf{R}} = 1. \tag{4.36}$$

The surface equations for any $(u, v) \in \Omega'$ are

$$\begin{gathered}\mathbf{r}_u \cdot \mathbf{r}_u = 1, \quad \mathbf{r}_u \cdot \mathbf{r}_v = 0, \quad \mathbf{r}_v \cdot \mathbf{r}_v = 1, \\ U h_u + V h_v = 1, \quad h_v \dot{\mathbf{r}}_u - h_u \dot{\mathbf{r}}_v = \mathbf{0}, \quad U\mathbf{r}_u + V\mathbf{r}_v + \rho = \mathbf{R}. \end{gathered} \tag{4.37}$$

While the number of unknowns here is 11 $(\mathbf{r}_u, \mathbf{r}_v, \rho, h_u, h_v)$ and the number of equations are 12 (Eqs. 4.36,4.37) but two of them are dependent(Eqs. including h_u and h_v). For unique solution of Eqs. (4.36,4.37), we differentiate Eq. (4.36) to obtain sufficient equations to solve the forward problem

$$\begin{gathered}\dot{\mathbf{r}}_u = \frac{h_u \mathbf{F}}{\dot{U} h_u + \dot{V} h_v}, \\ \dot{\mathbf{r}}_v = \frac{h_v \mathbf{F}}{\dot{U} h_u + \dot{V} h_v}, \\ h_u = \frac{g_u}{V g_v + U g_u}, \quad h_v = \frac{g_v}{V g_v + U g_u}, \\ \mathbf{F} = \ddot{\mathbf{R}} - \ddot{U}\mathbf{r}_u - \ddot{V}\mathbf{r}_v, g_u = \dddot{U} - \dddot{\mathbf{R}} \cdot \mathbf{r}_u, \; g_v = \dddot{V} - \dddot{\mathbf{R}} \cdot \mathbf{r}_v. \end{gathered} \tag{4.38}$$

These equations must be integrated numerically using, e.g., the Runge-Kutta method [68]. To generate the structure of the applicable surface we need for any characteristic line, the functions $\mathbf{r}_u\left(t\right)$,$\mathbf{r}_v\left(t\right)$ and $\rho\left(t\right)$; $\left(\mathbf{r}_u\left(t\right),\ \mathbf{r}_v(t)\right)$ are obtained from the solution to ODEs, while $\rho\left(t\right)$ is computed from the fifth equation in (4.37). The solution to our problem is a two-point boundary value problem (bvp). Most software for ODEs are written for initial value problems. To solve a bvp using an initial value solver, we need to estimate $\mathbf{r}_{u0} = \mathbf{r}_u\left(0\right)$ and $\mathbf{r}_{v0} = \mathbf{r}_v\left(0\right)$.which achieves the correct boundary value. The vectors $\mathbf{r}_{u0}$ and $\mathbf{r}_{v0}$ are dependent, since they satisfy the first three equations (4.37), which describe two orthonormal vectors. Assuming that $(\mathbf{r}_u, \mathbf{r}_v, \mathbf{r}_u \times \mathbf{r}_v)$ is a right-handed basis, we can always rotate the reference frame of the world coordinates so that in the rotated coordinates we have $\mathbf{r}_{u0} = \left(1,0,0\right)$, $\mathbf{r}_{v0} = \left(0,1,0\right)$. Consistent initial conditions $\mathbf{r}_{u0}$ and $\mathbf{r}_{v0}$ for Eq. (4.37) can be obtained by application of a rotation matrix $Q\left(\alpha,\beta,\gamma\right)$ with Euler angles α,β and γ, to the vectors $\left(1,0,0\right)$ and $\left(0,1,0\right)$, respectively. We note that for some particular cases it may happen that both the functions g_v and g_u in Eq. (4.38) may be zero. In this case the equations for h_u and h_v can be replaced by the limiting expressions for $g_v \rightarrow 0$, $g_u \rightarrow 0$. In the special case (rectangular patch in the parametric plane), we can show that there is an analytical solution given by:

$$\mathbf{r}_u = \frac{\ddot{\mathbf{R}} \times \dot{\mathbf{R}}}{\left|\ddot{\mathbf{R}}\right|}, \quad \mathbf{r}_v = \dot{\mathbf{R}}. \tag{4.39}$$

Proof: From Eqn. (4.22), we can imply that:

$$\dot{\mathbf{r}}_u \parallel \dot{\mathbf{r}}_v \parallel \mathbf{r}_{uu} \parallel \mathbf{r}_{uv} \parallel \mathbf{r}_{vv} || \left(\mathbf{r}_u \times \mathbf{r}_v\right), \tag{4.40}$$

Therefore $\dot{\mathbf{r}}_u$ and $\dot{\mathbf{r}}_v$ are perpendicular to $\mathbf{r}_u$ and $\mathbf{r}_v$. From Eqn. (4.38):

$$\begin{aligned} \dot{\mathbf{r}}_u \cdot \mathbf{r}_u = \frac{h_u(\mathbf{F} \cdot \mathbf{r}_u)}{\dot{U} h_u + \dot{V} h_v} = 0 \Rightarrow \mathbf{F} \cdot \mathbf{r}_u = 0, \\ \dot{\mathbf{r}}_v \cdot \mathbf{r}_v = \frac{h_v(\mathbf{F} \cdot \mathbf{r}_v)}{\dot{U} h_u + \dot{V} h_v} = 0 \Rightarrow \mathbf{F} \cdot \mathbf{r}_v = 0. \end{aligned} \tag{4.41}$$

Then, if we multiply $\mathbf{F} = \ddot{\mathbf{R}} – \ddot{U}\mathbf{r}_u - \ddot{V}\mathbf{r}_v$ in $\mathbf{r}_u$ and $\mathbf{r}_v$, we have the equation below:

$$\begin{aligned} \mathbf{F} \cdot \mathbf{r}_u &= \ddot{\mathbf{R}} \cdot \mathbf{r}_u - \ddot{U} = 0, \\ \mathbf{F} \cdot \mathbf{r}_v &= \ddot{\mathbf{R}} \cdot \mathbf{r}_v - \ddot{V} = 0. \end{aligned} \tag{4.42}$$

and in more simple form as:

$$\ddot{\mathbf{R}} \cdot \mathbf{r}_u = \ddot{U}, \;\; \ddot{\mathbf{R}} \cdot \mathbf{r}_v = \ddot{V}. \tag{4.43}$$

Moreover, by differentiating Eqn. $U\mathbf{r}_u + V\mathbf{r}_v + \rho = \mathbf{R}$ with respect to t, we have:

$$\dot{U}\mathbf{r}_u + \dot{V}\mathbf{r}_v = \dot{\mathbf{R}}. \tag{4.44}$$

Consider special rectangular patch in the parametric plane $U = \ddot{V} = 0$. Then denote

$$\mathbf{e}_z = \frac{\ddot{\mathbf{R}}}{\left|\ddot{\mathbf{R}}\right|}, \quad \mathbf{e}_x = \dot{\mathbf{R}}, \quad \mathbf{e}_y = \mathbf{e}_z \times \mathbf{e}_x, \tag{4.45}$$

where $(\mathbf{e}_x, \mathbf{e}_y, \mathbf{e}_z)$ makes a right-handed coordinate system. Let us expand $\mathbf{r}_u$ and $\mathbf{r}_v$ in this coordinate system as:

$$\mathbf{r}_u = \mathbf{e}_x \cos\psi + \mathbf{e}_y \sin\psi, \quad \mathbf{r}_v = \mathbf{e}_x \sin\psi - \mathbf{e}_y \cos\psi. \tag{4.46}$$

So by substituting in Eqn. (4.44),

$$\dot{U}\left(\mathbf{e}_x \cos\psi + \mathbf{e}_y \sin\psi\right) + \dot{V}\left(\mathbf{e}_x \sin\psi - \mathbf{e}_y \cos\psi\right) = \mathbf{e}_x,$$

$$\dot{U}\cos\psi + \dot{V}\sin\psi = 1,$$

$$-\dot{V}\cos\psi + \dot{U}\sin\psi = 0. \tag{4.47}$$

then the solution is

$$\cos\psi = \dot{U},\ \sin\psi = \dot{V}. \tag{4.48}$$

particularly, for $(\ \dot{U} = 0, \quad \dot{V} = 1)$, we have:

$$\cos\psi = 0,\ \sin\psi = 1. \tag{4.49}$$

then

$$\mathbf{r}_u = \mathbf{e}_y, \quad \mathbf{r}_v = \mathbf{e}_x. \tag{4.50}$$

therefore:

$$\mathbf{r}_u = \frac{\ddot{\mathbf{R}} \times \dot{\mathbf{R}}}{\left|\ddot{\mathbf{R}}\right|}, \quad \mathbf{r}_v = \dot{\mathbf{R}}.$$

4.3.6 Inverse Problem: 3D Structure Recovery of Applicable Surfaces

Here, we seek to estimate the 3D structure of an applicable surface from a single view (with known camera model) and knowledge of the undeformed *uv* plane boundary. For any point (x, y) in the image plane, we can estimate the corresponding point in the *uv*-plane and vice versa by solving the ODEs for the problem. The input parameters are the known camera model, the patch contours in the *uv*-plane and

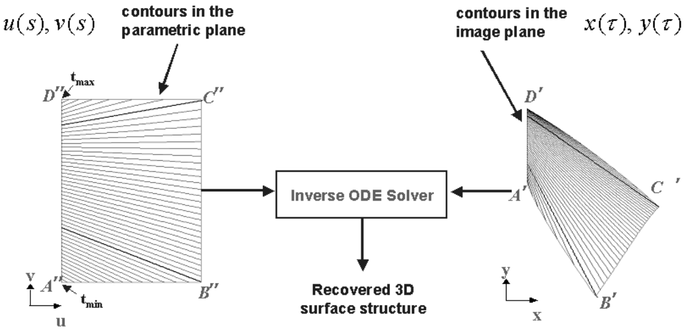

u(s), v(s)
contours in the parametric plane
contours in the image plane
x(τ), y(τ)
t_max
D''
C''
D'
Inverse ODE Solver
A'
C '
Recovered 3D surface structure
v
A''
t_min
B''
u
y
x
B'

Figure 4.4: Inverse Problem Schematic

the image plane. Assume that the image of the patch (Ω') is bounded by two curves Γ_1' and Γ_2', the corresponding patch (Ω) in the uv-plane is bounded by Γ_1 and Γ_2 and that the patch Ω bounded by the two characteristics, $t = t_{\min}$, and $t = t_{\max}$ (Fig. 4.4). We assume that Γ_1 and Γ_2 are piecewise continuous curves in the uv-plane, and not tangential to the characteristic lines $t_{\min} < t < t_{\max}$. For any point $(u_*, v_*) \in \Omega$ there exists a characteristic, $t = t_*$, which passes through some points on Γ_1 and some points on Γ_2. In the uv-plane these curves can be specified by a natural parameterization $u = U_1(s_1)$, $v = V_1(s_1)$ for Γ_1, and $u = U_2\left(s_2\right), v = V_2(s_2)$ for Γ_2, with $u^2 + v^2 \neq 0$. Here $s_1\left(t\right)$ and $s_2\left(t\right)$ are unknown and must be found in the process of solution.

Γ_1 and Γ_2 correspond to the 3D curves $\mathbf{r} = \mathbf{r}_1\left(t\right)$ and $\mathbf{r} = \mathbf{r}_2\left(t\right)$, which are unknown and found in the process of solution. Note that at the starting point or end point, Γ_1 and Γ_2 may intersect. At such a point the characteristic $t = t_{\min}$ or $t = t_{\max}$ is tangential to the boundary or the boundary is not smooth (e.g. we are at a corner). In case Γ_1 and Γ_2 intersect at $t = t_{\min}$ and $t = t_{\max}$ they completely define the boundary of the patch Ω. These cases are *not* special and can be handled by the general method described below. Assume that the camera is calibrated, and the relation between the world coordinates $\mathbf{r} = (X, Y, Z)$ and coordinates of the image plane (x, y) are known as $x = F_x(\mathbf{r})$ and $y = F_y(\mathbf{r})$. What is also known are the equations for Γ_1' and Γ_2' that are images of the patch boundaries Γ_1 and Γ_2. These equations, assumed to be in the form $x = x_1(\tau_1)$, $y = y_1(\tau_1)$ for Γ_1'; and $x = x_2(\tau_2)$, $y = y_2(\tau_2)$ for Γ_2'. Here τ_1 and τ_2 are the natural parameters of these curves; $\tau_1(t)$ and $\tau_2(t)$ are obtained from the solution. The specification of the curve parameters as "natural" means:

$$U_i'^2 + V_i'^2 = 1, \quad x_i'^2 + y_i'^2 = 1, \;\; i = 1, 2. \tag{4.51}$$

A complete set of equations describing the surface can be reduced then to

$$\begin{gathered}
\mathbf{r}_u \cdot \mathbf{r}_u = 1, \quad \mathbf{r}_u \cdot \mathbf{r}_v = 0, \quad \mathbf{r}_v \cdot \mathbf{r}_v = 1, \\
\mathbf{r}_2 = (U_2 - U_1)\,\mathbf{r}_u + (V_2 - V_1)\,\mathbf{r}_v + \mathbf{r}_1, \quad \dot{\mathbf{r}}_i = \dot{s}_i\left(U_i'\mathbf{r}_u + V_i'\mathbf{r}_v\right), \\
F_x(\mathbf{r}_i) = x_i(\tau_i), \quad F_y(\mathbf{r}_i) = y_i(\tau_i), \quad i = 1, 2.
\end{gathered} \tag{4.52}$$

We have 16 equations relating the 15 unknowns $(\mathbf{r}_u, \mathbf{r}_v, \mathbf{r}_1, \mathbf{r}_2, s_1, s_2, \tau_1, \tau_2)$. As in the previous case, one equation depends the other 15 and so the system is consistent. After $s(t), \mathbf{r}_1(t), \mathbf{r}_u(t)$, and $\mathbf{r}_v(t)$ are found, h_u, h_v, and ρ can be determined as

$$h_u = \frac{V_2 - V_1}{U_1V_2 - U_2V_1}, \;\; h_v = \frac{U_1 - U_2}{U_1V_2 - U_2V_1}, \;\; \rho = \mathbf{r}_1 - U_1\mathbf{r}_u - V_1\mathbf{r}_v. \tag{4.53}$$

This enables determination of $t(u, v)$ and $\mathbf{r}(u, v)$, similar to the forward problem. Here too the vector $\mathbf{w}$ is collinear to the normal to the surface (Eq. 4.23) and satisfies $\mathbf{w} = k\mathbf{n}$. Let the rate of change of s_1 be a constant, $\dot{s_{10}}$. The ODEs containing the unknowns $(s_1, s_2, \tau_1, \tau_2, \mathbf{r}_u, \mathbf{r}_v, \rho)$ can be written as follows:

$$
\begin{gathered}
s_1 = \dot{s}_{10} t,\ \dot{\tau}_1 = \dot{s}_{10} \mathbf{c}_1 \cdot \mathbf{a}_1, \\
\dot{s}_2 = -\frac{k \mathbf{f}_2 \cdot \mathbf{b}_2}{\mathbf{e}_2 \cdot \mathbf{b}_2 + \mathbf{c}_2 \cdot \left[\left(\mathbf{c}_2 \cdot \mathbf{a}_2 \right) \mathbf{d}_2 + \mathbf{G}_2 \cdot \mathbf{c}_2 \right]},\quad \dot{\tau}_2 = \dot{s}_2 \mathbf{c}_2 \cdot \mathbf{a}_2, \\
k = -\frac{\mathbf{e}_1 \cdot \mathbf{b}_1 + \mathbf{c}_1 \cdot \left[\left(\mathbf{c}_1 \cdot \mathbf{a}_1 \right) \mathbf{d}_1 + \mathbf{G}_1 \cdot \mathbf{c}_1 \right]}{\mathbf{f}_1 \cdot \mathbf{b}_1} \dot{s}_{10}, \\
\dot{\mathbf{r}}_u = k h_u \mathbf{n}, \\
\dot{\mathbf{r}}_v = k h_v \mathbf{n}, \\
\dot{\rho} = -k \mathbf{n}, \\
h_u = \frac{v_2 - v_1}{u_1 v_2 - u_2 v_1},\ h_v = \frac{u_1 - u_2}{u_1 v_2 - u_2 v_1}, \\
\mathbf{a}_i \left(\tau_i, \mathbf{r}_i \right) = \frac{x_i' \nabla F_x \left(\mathbf{r}_1 \right) + y_i' \nabla F_y \left(\mathbf{r}_i \right)}{x_i'^2 + y_i'^2}, \\
\mathbf{b}_i \left(\tau_i, \mathbf{r}_i \right) = y_i' \nabla F_x \left(\mathbf{r}_i \right) - x_i' \nabla F_y \left(\mathbf{r}_i \right), \\
\mathbf{c}_i \left(s_i, \mathbf{r}_u, \mathbf{r}_v \right) = u_i' \mathbf{r}_u + v_i' \mathbf{r}_v,\ \mathbf{d}_i = y_i'' \nabla F_x \left(\mathbf{r}_i \right) - x_i'' \nabla F_y \left(\mathbf{r}_i \right), \\
\mathbf{e}_i = u_i'' \mathbf{r}_u + v_i'' \mathbf{r}_v,\ \mathbf{f}_i = \left(u_i' h_u + v_i' h_v \right) \mathbf{n}, \\
\mathbf{G}_i = y_i' \nabla \nabla F_x \left(\mathbf{r}_i \right) - x_i' \nabla \nabla F_y \left(\mathbf{r}_i \right). \qquad (4.54)
\end{gathered}
$$

To start the integration of the inverse problem, we need initial conditions for $(s_1, s_2, \tau_1, \tau_2, \mathbf{r}_u, \mathbf{r}_v, \rho)$.

Solution to the Boundary Value Problem

While the equation above can be solved for a general camera model, we will consider the simple orthographic case here. We can show these initial values here are:

$$
\begin{aligned}
t_0 &= s_{10} = s_{20} = \tau_{10} = \tau_{20} = 0,\ \mathbf{r}_{10} = \mathbf{r}_{20} = \mathbf{r}_0, \\
u_{10} &= u_{20} = u_0,\ \ v_{10} = v_{20} = v_0,\ x_{10} = x_{20} = F_x \left(\mathbf{r}_0 \right),\ \ y_{10} = y_{20} = F_y \left(\mathbf{r}_0 \right),
\end{aligned}
$$

and for the starting point in 3D, $\mathbf{r}_0 = \mathbf{r}_0\left(x_0, y_0, z_0\right)$ where z_0 is some free parameter in the orthographic case. Note also that at the initial point the formulae for $h_u$and h_v

$$h_u = \frac{v_2 - v_1}{u_1 v_2 - u_2 v_1}, \quad h_v = \frac{u_1 - u_2}{u_1 v_2 - u_2 v_1}. \tag{4.55}$$

are not acceptable, since the numerators and denominators are zero. However, we can find h_{u0} and h_{v0} from

$$u_0 h_{u0} + v_0 h_{v0} = 1, \quad \dot{s}_{10}\left(u'_{10} h_{u0} + v'_{10} h_{v0}\right) = \dot{s}_{20}\left(u'_{20} h_{u0} + v'_{20} h_{v0}\right). \tag{4.56}$$

The solution of this linear system specifies h_{u0} and h_{v0} as a function of $\dot{s}_{20}$, which can be estimated from the free parameter, and is in fact one of the Euler angles γ_0 . Recalling that $\left(\mathbf{r}_u, \mathbf{r}_v, \mathbf{r}_u \times \mathbf{r}_v\right)$ is a right-handed basis, we can rotate the reference frame of the world coordinates by Euler angles $\left(\alpha_0, \beta_0, \gamma_0\right)$ so that we have $\mathbf{r}_{u0} = (1, 0, 0)\,,\ \mathbf{r}_{v0} = (0, 1, 0)$. Further:

$$\begin{gathered}
\dot{s}_{10}\mathbf{e}_{10} \cdot \mathbf{b}_{10} + k_0 \mathbf{f}_{10} \cdot \mathbf{b}_{10} + \dot{s}_{10}\mathbf{c}_{10} \cdot \left[\left(\mathbf{c}_{10} \cdot \mathbf{a}_{10}\right)\mathbf{d}_{10} + \mathbf{G}_{10} \cdot \mathbf{c}_{10}\right] = 0, \\
\dot{s}_{20}\mathbf{e}_{20} \cdot \mathbf{b}_{20} + k_0 \mathbf{f}_{20} \cdot \mathbf{b}_{20} + \dot{s}_{20}\mathbf{c}_{20} \cdot \left[\left(\mathbf{c}_{20} \cdot \mathbf{a}_{20}\right)\mathbf{d}_{20} + \mathbf{G}_{20} \cdot \mathbf{c}_{20}\right] = 0, \\
\mathbf{c}_{10} \cdot \mathbf{b}_{10} = 0, \mathbf{c}_{20} \cdot \mathbf{b}_{20} = 0.
\end{gathered} \tag{4.57}$$

These 4 relations can be treated as equations relating the 10 unknowns $k_0, \mathbf{r}_{u0}, \mathbf{r}_{v0}, \mathbf{n}_0$ ($\mathbf{r}_{u0}, \mathbf{r}_{v0}$ and $\mathbf{n}_0$ are 3D vectors). Also $\mathbf{r}_{u0}, \mathbf{r}_{v0}$, and $\mathbf{n}_0$ form an orthonormal basis, which therefore can be completely described by the three Euler angles $\left(\alpha_0, \beta_0, \gamma_0\right)$:

$$\mathbf{r}_{u0} = Q_0 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{r}_{v0} = Q_0 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad \mathbf{n}_0 = Q_0 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix},$$

where Q_0 is the Euler rotation matrix. This shows that $\mathbf{r}_{u0}$, $\mathbf{r}_{v0}$, and $\mathbf{n}_0$ a three-parameter set depending on $(\alpha_0, \beta_0, \gamma_0)$. Thus the relations Eq. (4.57) can be treated as 4 equations with respect to the unknowns $k_0, \alpha_0, \beta_0, \gamma_0$, for given $\dot{s}_{20}$ or $k_0, \alpha_0, \beta_0, \dot{s}_{20}$ for given γ_0, and can be solved. Then

$$\rho_0 = \mathbf{r}_0 - u_0 \mathbf{r}_{u0} - v_0 \mathbf{r}_{v0}. \tag{4.58}$$

determines ρ_0 as soon as $\mathbf{r}_{u0}$, $\mathbf{r}_{v0}$, and $\mathbf{r}_0$ are specified. Furthermore, we can reduce the four equations above to one nonlinear equation, whose roots can be determined by conventional numerical methods [68].

We found that this equation has two solutions, and so the Euler angles have four possible values. By choosing the free parameter γ_0 (Orthographic case), we can set all the initial conditions needed for the inverse problem. The challenge is to get the best estimate of γ_0 so that the boundary condition specifying correspondence points (such as the corners) is achieved. This is called the *shooting method.* We do this by minimizing a cost function J:

$$J = \underset{\gamma_0}{\arg\min} \, ||(x_e, y_e) - \mathbf{F}(\mathbf{r}(t_{\max}; \gamma_0, \Gamma_1, \Gamma_2, \Gamma_1', \Gamma_2'))|| \, , \tag{4.59}$$

where (x_e, y_e) is the image coordinates of the 3D surface ending point (X_e, Y_e, Z_e) and $\mathbf{r}(t_{\max}; \gamma_0, \Gamma_1, \Gamma_2, \Gamma_1', \Gamma_2')$ is the last step of the 3D structure solution and $\mathbf{F}$ is the camera model function. It is clear that $\mathbf{F}(\mathbf{r}(t_{\max}; \gamma_0, \Gamma_1, \Gamma_2, \Gamma_1', \Gamma_2'))$ is the ending point of 3D surface calculated by the ODE solver. Therefore, we change the free parameter γ_0 until we can hit the ending corner or are within a specified tolerance of the ending point in the image plane. If the number of the correspondence points on the edge available exceeds the number of shooting parameters (say the 4 corners) a least-square approach can be used.

Ambiguities

As stated in the inverse problem, the method relies on the boundary information of the patch in the image plane. So, since some deformations can lead us to the same images of the boundary, we have ambiguities. In these cases we need to extract other useful cues such as texture or shading to resolve the ambiguities. This is the subject for future work.

4.3.7 Forward: generating flat page in 3D

Let us assume that an applicable surface be a plane in 3D. Then, we can represent a plane at any point (X, Y, Z) in the world coordinate system as:

$$aX + bY + cZ + 1 = 0, \tag{4.60}$$

where (a, b, c) is a normal vector to the surface. In more restricted representation, we can consider a patch Ω with known length and width (l and w). We can uniquely define Ω given the surface normal (a, b, c), on a corner $\mathbf{C}_0 = (X_0, Y_0, Z_0)$ and the rotation angle around the normal vector γ_0 (Figure 4.5). We can prove that these parameters are enough to calculate other corners $(\mathbf{C}_1, \mathbf{C}_2, \mathbf{C}_3)$.

Let us define a right-handed coordinate system on the patch 'body coordinate system' with the basis vectors of $(\mathbf{E}_1, \mathbf{E}_2, \mathbf{E}_1 \times \mathbf{E}_2)$. We show that given γ_0 and surface normal vector it is enough to calculate the rotation matrix Q_0 which transforms the

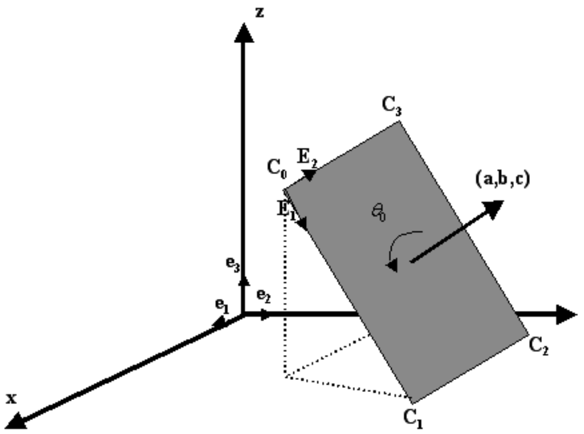


Figure 4.5: Patch in 3D

basis vector of the world coordinate system to the body coordinate system. Let us define the unit surface nomal vector n as:

$$\mathbf{n} = \frac{(\mathbf{a}, \mathbf{b}, \mathbf{c})}{||(\mathbf{a}, \mathbf{b}, \mathbf{c})||}. \tag{4.61}$$

and consider that the rotation matrix which transforms the world coordinate to the body coordinate system as $Q_0 = Q_0(\alpha, \beta, \gamma)$ where α, β and γ are Euler angles. $\mathbf{E}_1$ and $\mathbf{E}_2$ are the rotation of $\mathbf{e}_1$ and $\mathbf{e}_2$ in the world coordinate system by the rotation matrix $Q_0 = Q_0(\alpha, \beta, \gamma)$:

$$\mathbf{E}_1 = Q_0 \mathbf{e}_1, \quad \mathbf{E}_2 = Q_0 \mathbf{e}_2, \tag{4.62}$$

where $\mathbf{e}_1$ and $\mathbf{e}_2$ are $(1, 0, 0)^T$ and $(0, 1, 0)^T$ in the world coordinate system. We can compute α and β from the equation below, given that $\gamma = \gamma_0$:

$$\mathbf{E}_1 \times \mathbf{E}_2 = Q_0 \mathbf{e}_1 \times Q_0 \mathbf{e}_2 = \mathbf{n}. \tag{4.63}$$

Other corners are derived from the given corner as:

$$\mathbf{C}_1 = \mathbf{C}_0 + l\, \mathbf{E}_1, \quad \mathbf{C}_2 = C1 + w\, \mathbf{E}_2,\ \mathbf{C}_3 = \mathbf{C}_0 + w\, \mathbf{E}_2 \tag{4.64}$$

4.3.8 Inverse problem: special case

In this section, we consider the inverse for a flat surface. Here, the boundary of a flat surface in the image plane consist of lines. Therefore:

$$x^{''}(\tau_i) = y^{''}(\tau_i) = 0 \quad ; \; i = 1, 2 \tag{4.65}$$

The same conditions hold in the parametric plane. So,

$$u^{''}(\tau_i) = v^{''}(\tau_i) = 0 \quad ; \; i = 1, 2 \tag{4.66}$$

Hence, we can simplify Eqn. (4.54) into:

$$\begin{gathered}
s_1 = \dot{s}_{10} t, \; , \; \dot{s}_2 = \frac{\mathbf{f}_2 \cdot \mathbf{b}_2}{\mathbf{f}_1 \cdot \mathbf{b}_1} \, \dot{s}_{10}, \\
k = 0, \dot{\mathbf{r}}_u = 0, \dot{\mathbf{r}}_v = 0, \; \dot{\rho} = 0 \\
\dot{\tau}_2 = \dot{s}_2 \mathbf{c}_2 \cdot \mathbf{a}_2, \; \dot{\tau}_1 = \dot{s}_{10} \mathbf{c}_1 \cdot \mathbf{a}_1 \\
h_u = \frac{v_2 - v_1}{u_1 v_2 - u_2 v_1}, \; h_v = \frac{u_1 - u_2}{u_1 v_2 - u_2 v_1}, \\
\mathbf{a}_i \left(\tau_i, \mathbf{r}_i\right) = \frac{x_i' \nabla F_x \left(\mathbf{r}_1\right) + y_i' \nabla F_y \left(\mathbf{r}_i\right)}{x_i'^2 + y_i'^2}, \; \mathbf{b}_i \left(\tau_i, \mathbf{r}_i\right) = y_i' \nabla F_x \left(\mathbf{r}_i\right) - x_i' \nabla F_y \left(\mathbf{r}_i\right) = \mathbf{b}_i, \\
\mathbf{c}_i \left(s_i, \mathbf{r}_u, \mathbf{r}_v\right) = u_i' \mathbf{r}_u + v_i' \mathbf{r}_v, \; \mathbf{d}_i = 0, \quad \mathbf{e}_i = 0 \\
\mathbf{f}_i \left(s_i, \mathbf{r}_i\right) = \left(u_i' h_u + v_i' h_v\right) \mathbf{n} = \mathbf{f_i}, \; \mathbf{G}_i = 0.
\end{gathered} \tag{4.67}$$

These are the important conclusions from Eqn. (4.67):

- Tangent vectors ($\mathbf{r}_u(t) = \mathbf{r}_{u0}, \; \mathbf{r}_v(t) = \mathbf{r}_{v0}$) and surface normal ($n = \mathbf{r}_{u0} \times \mathbf{r}_{v0}$) are constant on any point on a flat surface.

- All characteristic lines are parallel in the parametric plane and in 3D. Therefore the characteristic line tangents are the same in the parametric plane ($\frac{h_v(t)}{h_u(t)} = \alpha$).

- Start and ending points are not adjacent in the parametric plane.

4.3.9 Homographic transformation

In this section, we want to prove that the mapping from any point in the *uv*-plane to any point in the *xy*-plane is homographic. Recall that:

$$\mathbf{r}(t) = u\mathbf{r}_{u0} + v\mathbf{r}_{v0} + \rho_0, \;\; u \in [u_{\min}, u_{\max}], \; v \in [v_{\min}, v_{\max}], \; uh_u(t) + vh_v(t) = 1. \tag{4.68}$$

we can re-write it in the matrix format as:

$$\mathbf{r} = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} \mathbf{r}_{u0} & \mathbf{r}_{v0} & \rho_0 \end{pmatrix} \begin{pmatrix} u \\ v \\ 1 \end{pmatrix}. \tag{4.69}$$

We can show Eqn. (4.69) in the homogenous coordinate:

$$\mathbf{X} = \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{pmatrix} \mathbf{r}_{u0} & \mathbf{r}_{v0} & \rho_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} u \\ v \\ 1 \end{pmatrix}. \tag{4.70}$$

We can relate image of a point $\mathbf{X}$ in the world coordinate in the image plane ($\mathbf{x}$) with:

$$\mathbf{x} = P\mathbf{X} = [\mathbf{K} \mid \mathbf{0}_{3\times 1}]\mathbf{X}, \tag{4.71}$$

where P is the 3×4 projection matrix. We can show that any point in *xy*-plane is related to the corresponding point in the *uv*-plane by a 3×3 homography matrix

H:

$$\mathbf{x} = P\mathbf{X} = P\begin{pmatrix} \mathbf{r}_{u0} & \mathbf{r}_{v0} & \rho_0 \\ 0 & 0 & 1 \end{pmatrix}\mathbf{u} = wH\mathbf{u}, \tag{4.72}$$

where $\mathbf{x} = (x, y, 1)^T$, $\mathbf{u} = (u, v, 1)^T$ and w is a scaling factor.

4.3.10 3D structure recovery from H

In the literature, we can compute H from 4 corners information up to scale (ref?). We can compute $(\mathbf{r}_{u0}, \mathbf{r}_{v0}, \rho_0)$ given the camera calibration matrix as follows:

$$[\mathbf{K} \mid \mathbf{0}_{3\times1}]\begin{pmatrix} \mathbf{r}_{u0} & \mathbf{r}_{v0} & \rho_0 \\ 0 & 0 & 1 \end{pmatrix} = wH, \tag{4.73}$$

then,

$$\mathbf{K}\begin{pmatrix} \mathbf{r}_{u0} & \mathbf{r}_{v0} & \rho_0 \end{pmatrix} = wH, \tag{4.74}$$

$$\begin{pmatrix} \mathbf{r}_{u0} & \mathbf{r}_{v0} & \rho_0 \end{pmatrix} = w\mathbf{K}^{-1}H, \tag{4.75}$$

we can compute scaling factor from the fact that the length tangent vectors $\mathbf{r}_{u0}$ and $\mathbf{r}_{u0}$ are one. Finally from the plane parameters $(\mathbf{r}_{u0}, \mathbf{r}_{v0}, \rho_0)$, 3D structure of a page is known (section forward-special case) given that:

$$\mathbf{E}_1 = \mathbf{r}_{u0},\ \mathbf{E}_2 = \mathbf{r}_{v0}, \mathbf{C}_0 = u_{\min}\mathbf{r}_{u0} + v_{\min}\mathbf{r}_{v0} + \rho_0,$$

$$w = u_{\max} - u_{\min},\ l = v_{\max} - v_{\min}. \tag{4.76}$$

4.4 Implementation Issues

4.4.1 Simple Validation of the Forward Problem

The purpose of this is to present and validate the new method. For this purpose we implemented the solution in algorithms. In the validation stage, we compared the results for warping to a 3D curve with the following analytical solution corresponding to a cylindrical surface

$$X = u - u_{\min},\ Y = N\cos\varphi\left(v\right),\ Z = N\sin\varphi\left(v\right),\ \varphi\left(v\right) = v/N. \tag{4.77}$$

To reproduce this surface we started our algorithm for warping with a 3D curve with the condition that in the (u, v)-plane the curve is a straight line, $u = u_{\min}$, and the fact that the corresponding 3D curve is

$$X(t) = 0, Y(t) = N\cos\varphi\left(t\right),\ Z(t) = N\sin\varphi\left(t\right). \tag{4.78}$$

For this surface we have the initial conditions for integration as $\mathbf{r}_{u0} = \left(-1, 0, 0\right)$, $\mathbf{r}_{v0} = (0, -\sin\varphi_0, \cos\varphi_0)$ with $\varphi_0 = v_{\min}/N$. We integrated the forward problem Eq. (4.38) numerically using an ODE solver from MATLAB, which was based on the 4^{th} order Runge-Kutta method. The results were identical to the analytical solution within the tolerance specified to the solver. We also checked that solution (4.39) is correct.

4.4.2 Forward Problem: Implementation Issues and Results

After initial tests we used the method of warping with 3D curves for generation of more complex applicable surfaces. The tests were performed both by straightforward numerical integration of ODE's (4.38) and using the analytical solution for rectangular patches (4.39). Both methods showed accurate and consistent results. To generate an example curve $\mathbf{R}(t)$ parameterized naturally, we specified another function $\widetilde{\mathbf{R}}(\theta)$ where θ is an arbitrary parameter and then used transform

$$\mathbf{R}(t) = \widetilde{\mathbf{R}}(\theta), \qquad \frac{dt}{d\theta} = \left| \frac{d\widetilde{\mathbf{R}}(\theta)}{d\theta} \right|, \tag{4.79}$$

which provides $\left|\dot{\mathbf{R}}\right| = 1$, and guarantees that t is the natural parameter. The function $\widetilde{\mathbf{R}}(\theta)$ used in tests was

$$\widetilde{\mathbf{R}}(\theta) = \left(P\left(\theta\right), N\cos\theta, N\sin\theta\right), \qquad P\left(\theta\right) = a_1\theta + a_2\theta^2 + a_3\theta^3 + a_4\theta^4, \tag{4.80}$$

and some other than polynomial dependencies $P\left(\theta\right)$ were tested as well. One of the examples of image warping with a 3D curve is presented in Figure 5.

For this case the boundary curve were selected in the form (4.80), with parameters $N = 200$, $a_1 = 20, a_2 = 10, a_3 = 10, a_4 = -10$ and we used Eqs (31) and (34) to generate the 3D structure and characteristics. In this example the characteristics for this surface are not parallel, which is clearly seen from the graph in the upper right corner of Fig. 5. The image of the portrait of Ginevra dé Bencia by Leonardo da Vinci, was fit into a rectangle in the uv-plane and warped with the generated surface. Further its orthographic projection was produced using pixel-by-pixel mapping of the obtained transform from the $\left(u, v\right)$ to the $\left(x, y\right)$. These pictures are also shown in Figure 5.

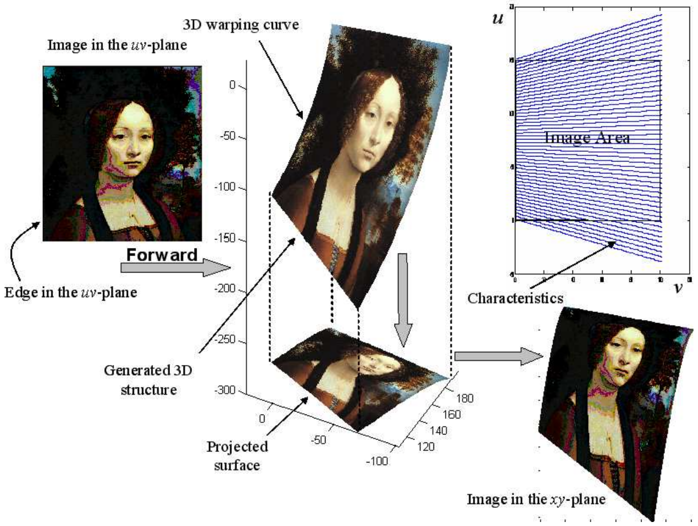


Figure 4.6: 'Forward' problem: given a plane sheet of paper, and a smooth 3-D open curve in Cartesian XYZ space. Our goal is to bend the paper so that one edge conforms to the specified curve. Using the analytical integration of the differential geometric equations specifying applicability we are able to achieve this. We can also achieve the same result not only for the straight line edge, but for an arbitrary 2-D curve in the uv-plane. The picture shown are actual computations.

4.4.3 Inverse Problem: Implementation Issues and Results

To check the validity of the unwarping procedure, we ran the 2D unwarping problem with synthetic input data on the patch boundaries and corner correspondence points obtained by the warping procedure. The output of the solver providing $h_u, h_v, \mathbf{r}_u, \mathbf{r}_v$, and ρ as functions of t coincided with these functions obtained by the 3D curve warping program within the tolerance specified for the ODE solver. The unwarped pixel-by-pixel images are shown in Figure 4.7 as the end point of the unwarping process in the xy-plane. We ran the algorithm for small fonts. The original image has the same font size everywhere and with the forward algorithm we warp the image. The unwarped image has uniform font size everywhere, lines are parallel and right angles are preserved. The output is noisy at the top of the output image, since in the image this information was lost. We make the following remarks about the implementation of the inverse problem:

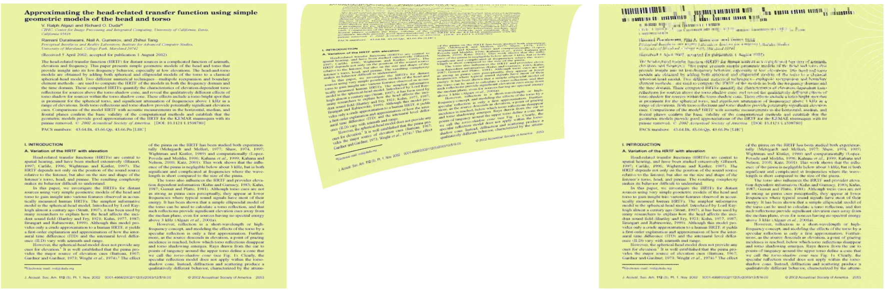

(a) (b) (c)

Figure 4.7: Inverse Problem for small font: a) original image b) warped by the forward $\tilde{\mathbf{R}}(\theta) = (a\theta(b - \theta^3), N cos\theta, N sin\theta)$ where $a = 10, b = 2, N = 200$ c) unwarped by the inverse problem

Global Parameterization: In the inverse problem, we march the ODE's with respect to the bounding contours in uv-plane and xy-plane. Therefore, for simplicity and modularity, we parameterize the contours using a global parameter η from η in [0,1] on the first boundary to $\eta = [3, 4]$ on the last. This parameterization gives us a simple and exact way of tracking the corners of the boundary contours and the correspondence between them.

ODE solver: To solve the ODE, we applied the Runge-Kutta solver of 4th and 5th order in MATLAB, except for the last edge of the ODE, where the problem was computationally stiff. For this, we solved the ODE using Gear's method [68].

Automatic Corner Detection by ODE solver: We need the corners in the image plane for the boundary of the patch to solve the inverse problem. As

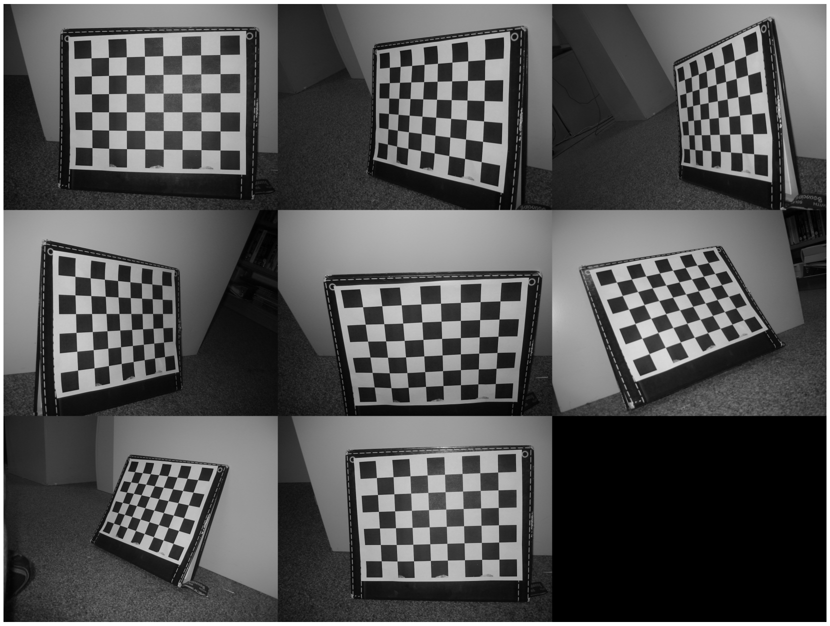


Figure 4.8: Images taken for the camera calibration parameter estimation

stated, the global natural parameterization of the curve in the image plane, gives us an easy and reliable feature for corner detection. Basically, the corner is reached when s_2 and τ_2 (global parameters of Γ_2' and Γ_2) are $1, 2$ and 3, respectively.

Camera Calibration: For 3D structure recovery, we need to have a camera model and the associated calibration matrix. Therefore, we used the MATLAB camera calibration toolbox at [7] to extract camera calibration matrix. To check the estimated parameters, we used the estimated matrix K in the simple case of the plane. In this case, we measure the angles of the page at corner and the length of each side in 3D using the camera calibration parameters. The results were as expected: a rectangle with the given length and width.

Chapter 5

3D Structure Recovery and Unwarping using texture information

5.1 Introduction

Texture is a phenomenon that is everywhere in the world, hard to define and easy to recognize. Texture is a regular or almost periodic pattern consisted of subelements (sometimes called textons). It can be viewed as larger numbers of small objects. Examples include grass, brush and marble ([29]). These subelements can be exactly the same (deterministic) or statistically the same (stochastic). Fig. 5.1 shows different texture patterns. Here, we are interested in treating documents as being composed of stochastic textured pattern. There are three standard problems in the texture literature:

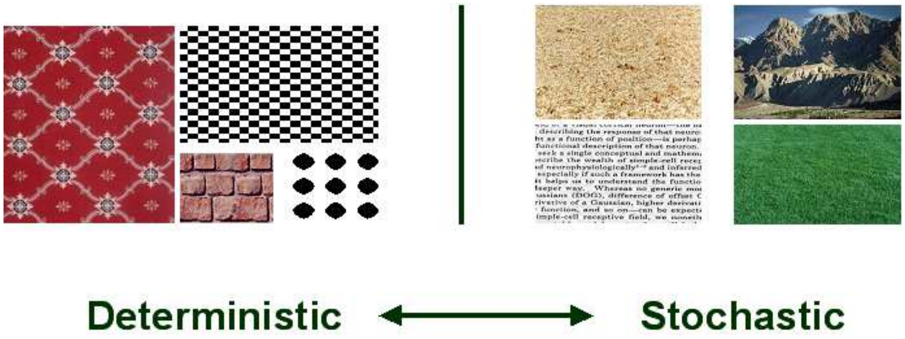


Figure 5.1: Textured pattern: Deterministic vs. Stochastic

- **Texture Segmentation** is the problem of splitting an image to small sub-elements within which the texture is constant. In the literature, the statistical properties of sub-elements are used as a feature. The goal is to obtain the size of the block for texture representation and to extract useful statistical properties to represent the extracted texture pattern. Examples are diverse but work most relevant to the thesis is text segmentation. In this problem, we identify the texture pattern as that which gives a similar response to the collection of wavelet filters e.g. Gabor filters(Chapter 3).

- **Texture Synthesis** is used to create a large texture pattern from small example images. This is done by modelling a texture pattern with a probabilistic framework from a set of image examples ([67]). We tried this method to model texture in document images but it fails. The reason is that a large amount of information on text images ignored if we use few filters to model the textual pattern.

- **Shape from Texture** attempts to recover a shape of the viewed object from two-dimensional image of a textured surface. There are two distinct areas of activity in the shape-from-texture problem. The first is focused at planar surfaces and studies aspects of perspective geometry using texture gradients. The second problem is to infer the shape of curved surfaces. Without a priori knowledge of the camera model, we can determine some shape properties e.g. local surface orientation from texture gradient. With a camera model, the problem is 3D structure recovery. In this thesis, we are interested in shape from texture and 3D structure recovery from single view for surfaces applicable to planes.

5.2 Previous works

The shape-from-texture problem originates from the work of Gibson at [32]. He introduced the term texture gradient as a feature for the 3D shape recovery. The difference between the images of two adjacent and similar texture patterns allows us to infer local orientation of a surface in 3D. Moreover, Marr [53] stated that texture information as providing a potentially useful data of the shape recovery.

Shape from texture proceeds along two main lines: (1) Measuring the texture gradient (2) 3D shape recovery and integrability. For problem (1) One approach is a global method, where it is assumed that the distribution of the texture is given e.g., isotropy [88] or homogeneity [1]. These assumptions on the texture model made us decide not to employ these methods for text patterns. Local methods recover some differential geometric properties at a point on a surface. There are surprising few works in the literature for recovering surface properties using local methods. This method is based on the work of Garding [30] and was expanded to various surfaces by [52]. A reformulation of the problem in terms of wavelet basis was done by [16]. Here, we chose the Malik and Rosenholtz algorithm [52] for further development.

These local methods provide local estimation of surface properties. These local estimates are not necessarily consistent. This problem is called '*integrability*'. In [46], a new approach to the reconstruction of surface normals using basis functions, referred as shapelets was presented. These shapelets are smooth in nature but it does not impose the applicable surface property. Here, we used their reconstruction method to reconstruct the 3D structure of the surface.

Our contribution is use of the applicable property for the 3D reconstruction of the surface from texture information. In our method we use the differential geometric properties for the integration of normal vectors. Such properties are isometry and vanishing Gaussian curvature. Here, we used local robust smoothing to satisfy this property.

5.3 Theory

5.3.1 Review of the geometric framework

The basic underlying geometry is shown in Fig. 5.2. A surface S is mapped to a unit sphere view of Σ by a central projection model. The view is chosen spherical because it is uniform in each direction. This viewing model is the camera model assumption. In Fig. 5.2, F is a back projection of a point on the imaging view to a point on surface in 3D. The slant σ is defined to be the angle between the surface normal $\mathbf{N}$ and the viewing direction $\mathbf{p}$. Defining the tilt direction $\mathbf{t}$, the tangent plane of the viewing sphere at $\mathbf{p}$ to be a unit vector in the direction of the gradient of the distant function $r(\mathbf{p})$. We denote the back projection mapping from the viewing sphere to the surface as $F(\mathbf{p}) = \mathbf{r}(\mathbf{p}) = \mathbf{p}r(\mathbf{p})$ where $\mathbf{p}$ is a unit vector from the focal point to a point on the image sphere, and $r(\mathbf{p})$ is the distance along the visual ray from the focal point through $\mathbf{p}$ to the corresponding point $\mathbf{r} = F(\mathbf{p})$ on the surface S. The orientation of the surface can be defined in different ways. One convenient choice are the slant-tilt parameters (σ, τ) where $\sigma \in (0, \pi)$ is the slant, and $\tau \in (0, 2\pi)$ is the tilt. Let us define the direction of $\mathbf{b} = \mathbf{t} \times \mathbf{p}$. Then, $(\mathbf{t}, \mathbf{b})$ forms an orthonormal basis in the image plane at point $\mathbf{p}$. Furthermore, $(\mathbf{T}, \mathbf{B})$ are the back projections of vectors $(\mathbf{t}, \mathbf{b})$ on a surface at point $\mathbf{F}(\mathbf{p})$. Garding [30]

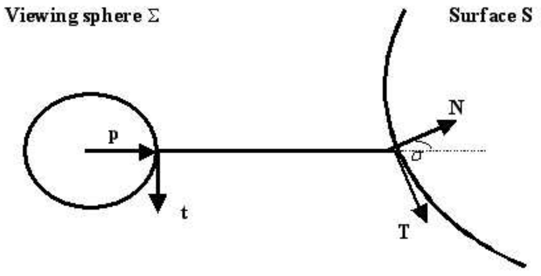


Figure 5.2: A smooth surface S is mapped to a unit sphere Σ centered at the focal point by central projection.

formulates $\mathbf{F}(\mathbf{p})$ in terms of $(\mathbf{T}, \mathbf{B})$ as

$$F(p) = \begin{pmatrix} \frac{r}{\cos\sigma} & 0 \\ 0 & r \end{pmatrix} = \begin{pmatrix} \frac{1}{m_p} & 0 \\ 0 & \frac{1}{M_p} \end{pmatrix}, \tag{5.1}$$

where r is the distance to a point on the surface to the center of viewing sphere, σ is slant angle, m_p and M_p are respectively the scaling of the minor and major axis in the tilt direction. Fig 5.3 the projection of a planar surface with circles of the same size painted on it. We can see the minor axis of each ellipse in the tilt direction. Major and minor axis notations are motivated by Fig. 5.3. In this figure, the local orientation of the plane makes circles be perceived as ellipses. We assumed that the surface is smooth therefore for any two adjacent points $\mathbf{P_1}$ and $\mathbf{P_2}$, the corresponding orthonormal basis $(\mathbf{T_1}, \mathbf{B_1})$ and $(\mathbf{T_2}, \mathbf{B_2})$ are rotated replica of each other. The transformation can be formulated by a rotation matrix $R(\delta_T)$. It can be shown that [30] for any two adjacent points $\mathbf{p_1}$ and $\mathbf{p_2}$ on the image sphere of Σ, the mapping is affine , and determined by a matrix A. Fig. 5.3.1 illustrates the affine mapping from $\mathbf{p_1}$ to $\mathbf{p_2}$ that is obtained if we back project $\mathbf{p_1}$ to $\mathbf{P_1}$, rotate the basis from $(\mathbf{T_1}, \mathbf{B_1})$ to $(\mathbf{T_2}, \mathbf{B_2})$ and project it to $\mathbf{p_2}$ from $\mathbf{P_2}$ and finally rotate the basis

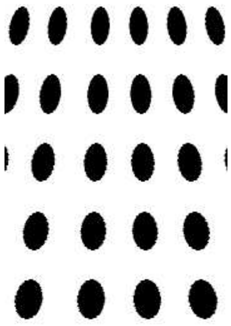


Figure 5.3: Projection of circle textured pattern.

in the image sphere by a rotation matrix $R(\delta_t)$. Therefore, A can be formulated for $\triangle\mathbf{t}, \triangle\mathbf{b} \rightarrow \mathbf{0}$ as:

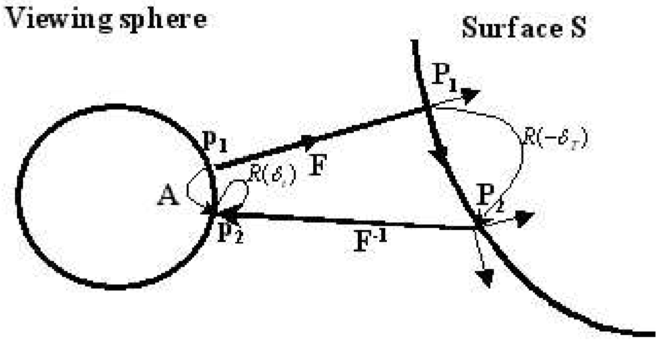


Figure 5.4: Affine transformation between two adjacent points $\mathbf{P_1}$ and $\mathbf{P_2}$ on a surface. The corresponding image are represented as $\mathbf{p_1}$ and $\mathbf{p_2}$. The mapping between two image points is affine matrix A. $R(\delta_T)$ is the rotation of the $(\mathbf{T}, \mathbf{B})$ between $\mathbf{P_1}$ and $\mathbf{P_2}$ and $R(\delta_t)$ rotates between the $(\mathbf{t}, \mathbf{b})$ basis from point p_1 to p_2.

$$\begin{aligned} A &= R(\delta_t)F^{-1}(\mathbf{P}_2)R(\delta_T)F(\mathbf{p}_1), \\ &= R(\delta_t)\begin{pmatrix} \cos\delta_T\frac{m_2}{m_1} & \sin\delta_T\frac{m_2}{M_1} \\ -\sin\delta_T\frac{M_2}{m_1} & \cos\delta_T\frac{M_2}{M_1} \end{pmatrix}, \end{aligned} \tag{5.2}$$

where

$$\begin{aligned} m_2 &= m_1 + \begin{pmatrix} \triangle\mathbf{t} & \triangle\mathbf{b} \end{pmatrix} \circ \nabla m, \\ M_2 &= M_1 + \begin{pmatrix} \triangle\mathbf{t} & \triangle\mathbf{b} \end{pmatrix} \circ \nabla M, \end{aligned} \tag{5.3}$$

m and M are the minor and major axis scaling at point $\mathbf{p}$ and ∇m and ∇M are the gradients. Garding [30] proved that the gradients of the minor and major axis can be represented in the $(\mathbf{t}, \mathbf{b})$ coordinates as:

$$\frac{\nabla m}{m} = -\tan\sigma \begin{pmatrix} 2 + r\kappa_t / \cos\sigma \\ \\ r\tau \end{pmatrix},$$

$$\frac{\nabla M}{M} = -\tan\sigma \begin{pmatrix} 1 \\ \\ 0 \end{pmatrix}. \tag{5.4}$$

where κ_t is the normal curvature in the $\mathbf{T}$ direction and τ is the geodesic torsion. Also, we can relate $\mathbf{p_1}$ to $\mathbf{p_2}$ by :

$$\mathbf{p_1} - \mathbf{p_2} = (\triangle\mathbf{t}, \triangle\mathbf{b})^T \tag{5.5}$$

We can rewrite Eqn. (5.2) using Eqns. (5.1) and (5.4):

$$A = R(\delta_t) \begin{pmatrix} k_m \cos\delta_T & k_m \sin\delta_T \cos\sigma \\ \\ -k_M \frac{\sin\delta_T}{\cos\sigma} & k_M \cos\delta_T \end{pmatrix}, \tag{5.6}$$

where

$$k_m = 1 + \begin{pmatrix} \triangle\mathbf{t} & \triangle\mathbf{b} \end{pmatrix} \circ \frac{\nabla m}{m},$$

$$k_M = 1 + \begin{pmatrix} \triangle\mathbf{t} & \triangle\mathbf{b} \end{pmatrix} \circ \frac{\nabla M}{M}. \tag{5.7}$$

The actual affine transformation is not necessarily in the $(\mathbf{t}, \mathbf{b})$ basis. Therefore, we have to change the basis: $\bar{A} = UAU^{-1}$. Here, U is a rotation matrix with tilt as the angle of rotation.

5.3.2 Affine estimation

In this section, we assume that two images I_1 and I_2 are related by an affine transformation:

$$I_2(\mathbf{x}) = I_1(A\mathbf{x}), \tag{5.8}$$

where x is the pixel location in $I_1$and A is an affine matrix. We can show that in the frequency domain, the Fourier transform of I_1 and I_2 are related by an affine matrix as well

$$F_2(\mathbf{x}) = \frac{1}{\det(A)} F_1(A^{-T}\mathbf{x}). \tag{5.9}$$

Thus if we can calculate the affine transformation in the frequency domain, we can consequently compute it in the spatial domain. There are advantages to working in the frequency domain: calculations are insensitive to small changes in the spatial domain and are fast because of the availability of the fast Fourier transform. Now, assume that we have the Fourier transforms of images I_1 and I_2. We use a differential method to estimate the affine matrix parameter. To illustrate the concept, we first consider the case in $1D$:

$$f_2(w) = f_1(aw), \tag{5.10}$$

then suppose $a = 1 + \triangle a$ where $\triangle a$ is small. From the Taylor expansion:

$$f_2(w) = f_1(w + w \triangle a) \approx f_1(w) + \frac{\partial f_1}{\partial w} w \triangle a, \tag{5.11}$$

Therefore, if f_1 and f_2 are given, we can approximately solve for affine parameter $\triangle a$. We can extend the method to $2D$. Assume that the two spectograms are related by affine matrix $A = I + \triangle A$. Then the difference equation is:

$$F_2(\mathbf{w}) - F_1(\mathbf{w}) \approx \nabla F_1 \circ \triangle A \mathbf{w}, \tag{5.12}$$

The only unknown is $\triangle A$ which is a 2×2 matrix. If we write the elements of $\triangle A$ as a_{ij}, we can rewrite Eqn. 5.12 as:

$$F_2(\mathbf{w}) - F_1(\mathbf{w}) \approx \begin{pmatrix} \frac{\partial F_1}{\partial w_1} w_1 & \frac{\partial F_1}{\partial w_1} w_2 & \frac{\partial F_2}{\partial w_2} w_1 & \frac{\partial F_2}{\partial w_2} w_2 \end{pmatrix} \begin{pmatrix} a_{11} \\ a_{12} \\ a_{21} \\ a_{22} \end{pmatrix}, \tag{5.13}$$

where each partial derivative is computed at frequency $\mathbf{w}$. This is a standard linear matrix equation $\mathbf{Ax} = \mathbf{b}$. We have many independent equations if the $2D$ signals are rich in content. To find the solution to the affine matrix only four independent equations are enough. In this formulation, we can have more than four points and the solution is given by the least square solutions. The method is explicitly addressed in [52].

5.3.3 Shape recovery

To estimate the texture gradient map at a point $\mathbf{p}_1$, we calculate the Fourier transforms of the block centering at that point and for neighboring points in a number of different directions. Therefore, $\mathbf{p}_2$ is given by:

$$\mathbf{p}_2 = \mathbf{p}_1 + d(\cos\theta_i, \sin\theta_i)^T, \tag{5.14}$$

where d the distance to be chosen and θ_i the orientation angle. In this section , we apply the shape recovery algorithm addressed in [52] by Malik and Rosenholtz. In this method, the tilt and slant parameters are computed from the estimation of affine matrix for each direction. Recall that for the orientation angle θ_i, the affine matrix can be formulated as:

$$\bar{A}_i = UA_iU^{-1} = UR(\delta_t^i)\begin{pmatrix} k_m^i \cos\delta_T & k_m^i \sin\delta_T^i \cos\sigma \\ -k_M^i \frac{\sin\delta_T^i}{\cos\sigma} & k_M^i \cos\delta_T^i \end{pmatrix} U^{-1}$$

where $U = R(\theta_t)$ and θ_t is the tilt direction. The three stages of the shape-recovery algorithm are:

1. Compute the singular values of $\bar{A}_i$ denoted as s_1^i and s_2^i. It can be shown that:

$$\begin{aligned} trace(\bar{A}_i^T \bar{A}_i) &= trace(A_i^T A_i), \\ \det(\bar{A}_i^T \bar{A}_i) &= \det(A_i^T A_i). \end{aligned} \qquad (5.15)$$

We drop superscripts i to avoid confusion. Therefore, we can claim that:

$$\begin{aligned} (s_1+s_2)^2 &= trace(\bar{A}_i^T \bar{A}_i) = trace(A_i^T A_i) \approx (k_M + k_m)^2, \\ s_1^2 s_2^2 &= \det(\bar{A}_i^T \bar{A}_i) = \det(A_i^T A_i) \approx k_M^2 k_m^2, \end{aligned} \qquad (5.16)$$

where $\triangle\mathbf{t}_i, \triangle\mathbf{b}_i \rightarrow \mathbf{0}$. So, we can approximate $s_1 + s_2 \approx k_M + k_m$ and $s_1 s_2 \approx k_M k_m$. Therefore, we can estimate k_M and k_m from the singular values of the estimate of $\bar{A}$. As we stated before, the major scaling is in the minor axis direction, k_M is close to one. This is true for planar surfaces and for many surfaces with positive curvature and torsion.

2. Estimate the tilt and slant angles from the major axis gradient. from Eqns. (5.7) and (?) that $k_M^i - 1 = -\tan\sigma \triangle t_i$. Here, we still do not know the tilt direction and so forth $\triangle t_i$. Let us represent the major axis gradient as $(t_x, t_y)^T$ in the standard basis, (x, y). This vector has length $|-\tan\sigma|$ and it is in the

tilt direction. Then,

$$\begin{pmatrix} \triangle x_1 & \triangle y_1 \\ \triangle x_2 & \triangle y_2 \\ . & . \\ \triangle x_n & \triangle y_n \end{pmatrix} \begin{pmatrix} t_x \\ t_y \end{pmatrix} = \begin{pmatrix} k_M^1 - 1 \\ k_M^2 - 1 \\ . \\ k_M^n - 1 \end{pmatrix}, \tag{5.17}$$

where $\triangle x_i = d\cos\theta_i$ and $\triangle y_i = d\sin\theta_i$.

3. Smooth the surface normals robustly. The surface normals calculated by the shape-from-texture method are likely to be noisy and inconsistent. This inconsistency in the sense of local smoothness of the surface, relegates the needle map estimate for curved surface usage. Therefore, to improve the consistency of the needle map and hence the surface local smoothness criteria, we need to impose an iterative scheme to smoothen out the normals. We used the algorithm at [90]. This smoothing is a robust smoothing method whereas the smoothing process can be gauged by the choice of the kernel. Here, we choose the smoothness penalty as:

$$I = \int\int \{\rho_\sigma(||\frac{\partial \mathbf{n}}{\partial x}||) + \rho_\sigma(||\frac{\partial \mathbf{n}}{\partial y}||)\} dxdy, \tag{5.18}$$

where $\rho_\sigma(\eta)$ is the robust error kernel. Our choice of the kernel is:

$$\rho_\sigma(\eta) = \frac{\sigma}{\pi} \log\cosh(\frac{\pi\eta}{\sigma}). \tag{5.19}$$

This kernel is the log-cosh sigmoidal derivatives M-estimator. Applying variational calculus the update equation for the surface normals which minimizes the smoothness penalty I is:

$$\begin{aligned}\mathbf{n}^{k+1}_{i,j} = {} & ||\frac{\partial \mathbf{n}^k_{i,j}}{\partial x}||^{-1}\tanh(\frac{\pi}{\sigma}||\frac{\partial \mathbf{n}^k_{i,j}}{\partial x}||)(\mathbf{n}^k_{i+1,j}-\mathbf{n}^k_{i-1,j}) + (\frac{\pi}{\sigma}||\frac{\partial \mathbf{n}^k_{i,j}}{\partial x}||^{-2}\sec h^2(\frac{\pi}{\sigma}||\frac{\partial \mathbf{n}^k_{i,j}}{\partial x}||)- \\ & ||\frac{\partial \mathbf{n}^k_{i,j}}{\partial x}||^{-3}\tanh(\frac{\pi}{\sigma}||\frac{\partial \mathbf{n}^k_{i,j}}{\partial x}||))(\frac{\partial \mathbf{n}^k_{i,j}}{\partial x}\cdot\frac{\partial^2 \mathbf{n}^k_{i,j}}{\partial x^2})\frac{\partial \mathbf{n}^k_{i,j}}{\partial x} + ||\frac{\partial \mathbf{n}^k_{i,j}}{\partial y}||^{-1}\times \\ & \tanh(\frac{\pi}{\sigma}||\frac{\partial \mathbf{n}^k_{i,j}}{\partial y}||)(\mathbf{n}^k_{i,j+1}-\mathbf{n}^k_{i,j-1}) + (\frac{\pi}{\sigma}||\frac{\partial \mathbf{n}^k_{i,j}}{\partial x}||^{-2}\sec h^2(\frac{\pi}{\sigma}||\frac{\partial \mathbf{n}^k_{i,j}}{\partial y}||) \\ & -||\frac{\partial \mathbf{n}^k_{i,j}}{\partial y}||^{-3}\tanh(\frac{\pi}{\sigma}||\frac{\partial \mathbf{n}^k_{i,j}}{\partial y}||))(\frac{\partial \mathbf{n}^k_{i,j}}{\partial y}\cdot\frac{\partial^2 \mathbf{n}^k_{i,j}}{\partial y^2})\frac{\partial \mathbf{n}^k_{i,j}}{\partial y}. \end{aligned} \tag{5.20}$$

where $\mathbf{n}^k_{i,j}$ is the estimate of the surface normals at the i^{th} row and j^{th} column at the iteration k of the smoothing process.

5.4 Discussion and Results

Here, we show the results of the shape-from-texture algorithm for different surfaces: positive Gaussian curvature and zero Gaussian curvature. For all results, the parameters stated in section 5.3: d= 7 , number of orientations= 8, block-size= 64 and image-size= 256 × 256. We robustly smoothed the surface normal needle map in 25 iterations.

In Fig. 5.5, we show the results of the algorithm for a golf ball. In this example the tilt-slant parameters estimations are accurate because we have a deterministic textured pattern and we picked the affine parameters using 3 features in the frequency domain. To overcome inconsistencies in the sense of surface normals, we applied the robust smoothing to the needle-map field.

In Fig. 5.6 and 5.7, we show the result of our algorithm on surfaces with zero Gaussian curvature. The first example is a flat surface and the second example is a

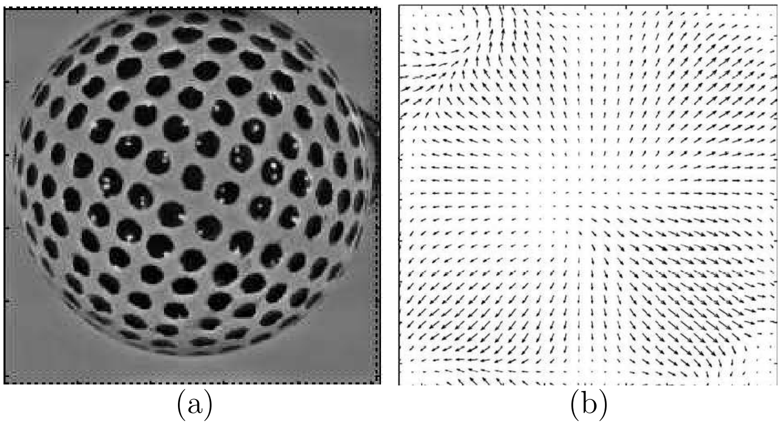
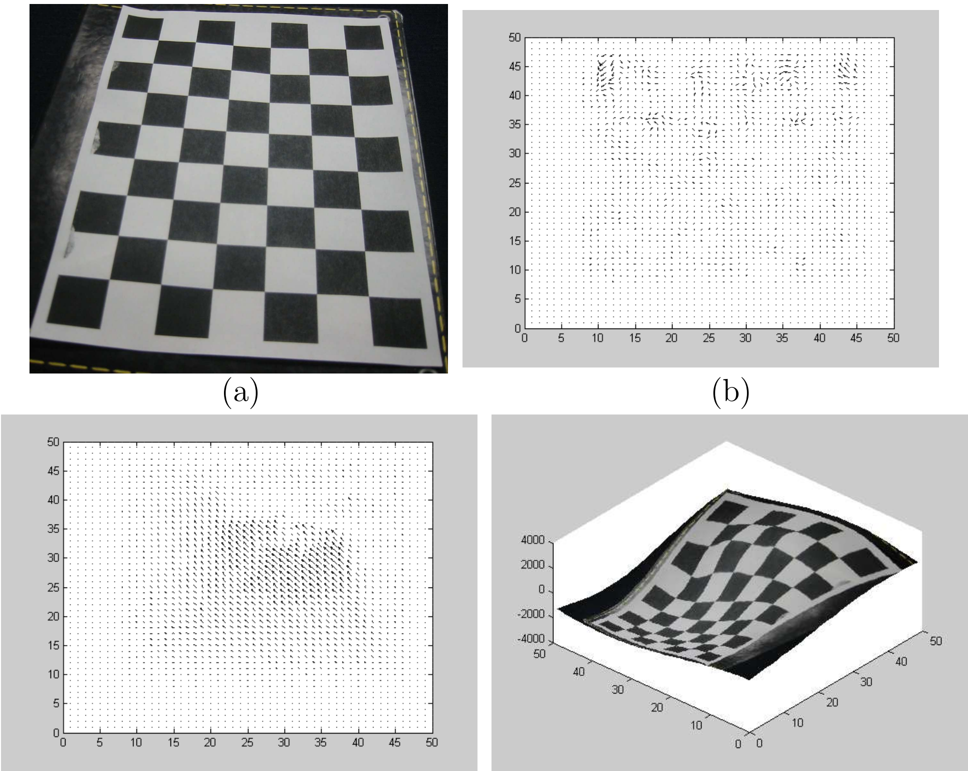


Figure 5.5: (a) Original image of a golf ball (Gaussian curvature is nonzero) (b) Normal vector field after robust smoothing

worked surface with deformation applicable to a plane. Both input images are taken by a digital camera Canon $S200$-PowerShot. In these examples, the camera model is (viewing model) not a spherical central projection model. Also, we automatically estimated the affine parameters at each chosen point on the surface which is not as precise as manual method of picking points in the frequency domain. Furthermore, in the reconstruction stage, we used the Kavosi method [46] for the 3D reconstruction using a set of smooth Gaussian kernels. Therefore, the results are smoothed out even if the normal surface needle map were noisy. However, in Fig. 5.6, we can infer the smooth increasing nature of the tilted plane at the center of the surface. Moreover, in Fig. 5.7, we can claim the nature of the bending. So far, we tried the automatic affine estimation module which is not very precise and consequently affect the results.

In the future, we will work in the accuracy of the affine estimation module which has direct impact on the quality of surface local orientation results. Also, we

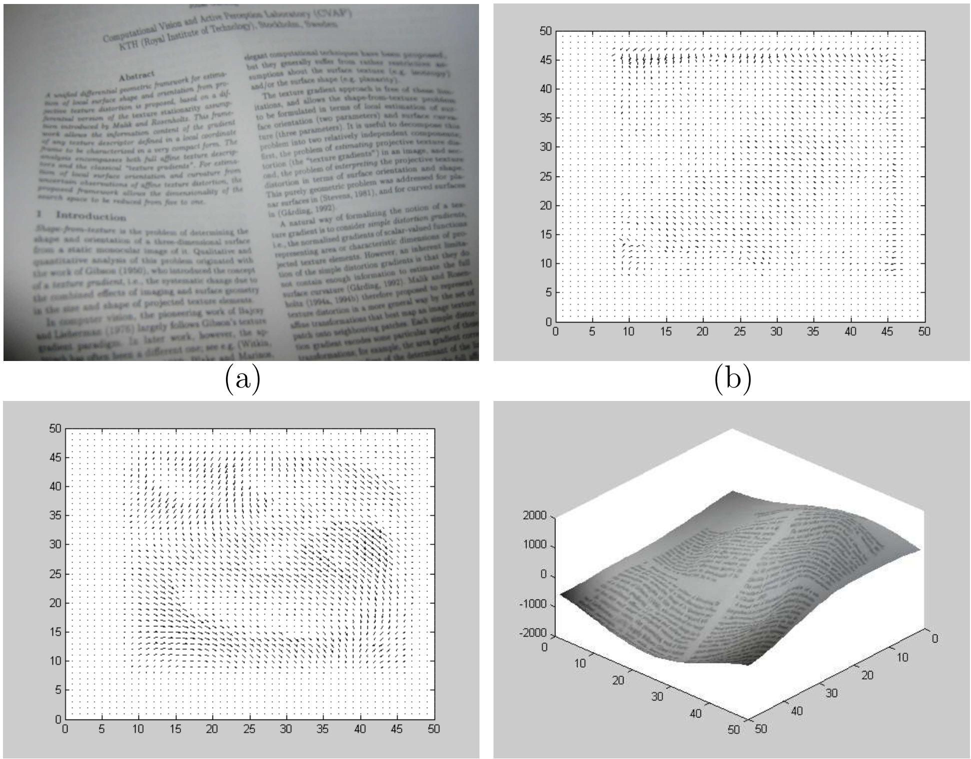


(a) (b)

(c) (d)

Figure 5.6: (a) Original image of a flat paper (b) Normal vector field from shape-from-texture algorithm (c) Robustly smoothed normal vector field (d) 3D reconstruction

think with a more elaborate objective function for the robust smoothing method, we can improve significantly the results for the applicable surface. The additional term can be the direction of the vectors in the needle-map field. As stated in chapter 4, at any point on the surface there is a direction which has zero curvature (back to zero Gaussian curvature property). On this direction, surface normals are parallel as stated in 4.



(a) (b)

(c) (d)

Figure 5.7: (a) Original image of a bent paper (b) Normal vector field from shape-from-texture algorithm (c) Robustly smoothed normal vector field (d) 3D reconstruction

Chapter 6

Conclusion and Future work

In this dissertation, we first described the development of a prototype device for scene text acquisition and processing for the visually impaired to access textual information in the environment. This integrated system uses video, image processing, optical-character-recognition (OCR) and text-to-speech (TTS). The video provides a sequence of low resolution images in which text must be detected, rectified and converted into high resolution rectangular blocks that are capable of being analyzed via off-the-shelf OCR. To achieve this, various problems related to feature detection, mosaicing, auto-focus, zoom, and systems integration were addressed in the development of the system.

Next we employ the video-based interface for a case that the scene text is printed on plane. Several modules for pre-processing and analysis were augmented e.g., text segmentation, enhancement and pre-processing the video content and metric rectification. we presented a system to extract textual and graphical information in lecture presentations or poster/slides using video and image processing, optical-character-recognition (OCR) and pattern recognition. Related computer vision problems were introduced and solved. The results were promising and efficient for the video-based interface. The indexed output is represented in structured format; text , graph and importance of text in the content. So, a video of lecture or slide/poster can be compressed without losing any key information, and still be small enough to be retrieve in on-line environments. The ability to capture and process textual information access by camera-based scanning system has many applications e.g. mobile text reader for the visually impaired, sign detection and translation, document and conference archiving, semantic extraction and so on.

Next, we considered a more general class of worked paper surfaces with scene text printed on them. Such surfaces are applicable to planes. This thesis presents, to our knowledge, the first occasion that differential geometry has been used quantitatively in the recovery of structure from images. A theory and method for warping and unwarping images for applicable surfaces based on patch boundary information and solution of nonlinear PDEs of differential geometry was developed. The method is fast, accurate and correspondence free (except for a few boundary points). We see many useful applications of this method for virtual reality simulations, computer vision, and graphics; e.g. 3D reconstruction, animation, object classification, OCR, etc. While the purpose of this study was developing and testing of the method itself, ongoing work is related both to theoretical studies and to development of practical algorithms. This includes more detailed studies of the properties of the obtained equations, problems of camera calibration, boundary extraction, sensitivity analysis, efficient minimization procedures, and unwarping of images acquired by a camera, where our particular interest is in undoing the curl distortion of pages with printed text.

We next developed the shape-from-texture method to complement the method above to infer the 3D structure. We showed that for the consistency of normal vector field, we need to add extra conditions based on the surface model. Such conditions are isometry and zero Gaussian curvature of the surface.

The novel contributions are: first, it is shown that certain linear and non-linear clues (contour knowledge information) are sufficient to recover the 3D structure of scene text; second, that with a priori of a page layout information, we can reconstruct a fronto-parallel view of a deformed page from differential geometric properties of a surface; third, that with a known cameral model we can recover 3D structure of a bent surface; forth, we present an integrated framework for analysis and rectification of scene texts from single views in general format; fifth, we provide the comparison with shape from texture approach and finally this work can be integrated as a visual prostheses for the visually impaired.

We expect our work to have many applications in computer vision and computer graphics. The applications are diverse e.g. a generalized scanning device, digital flattening of creased documents, 3D reconstruction problem when correspondence fails, 3D reconstruction of single old photos, bending and creasing virtual paper, object classification, semantic extraction, scene description and so on. The direction for the future work is listed as following:

6.1 Image enhancement and restoration

Let us assume that the images taken by a a camera is blurred. The statement is that given a degraded image of printed text can we restore the image such that image looks as if it is taken from an in-focus camera? Here, we will point out the direction for this problem. In the thesis, the solution to this problem is beneficial to the improvement of the OCR output accuracy. In more general definition, the degradation process can be modeled as:

$$y = H(x) + \eta \tag{6.1}$$

where x is the original image, y is the blurred image, H is the PSF (point spread function) that degrades the original image and

eta is an additive noise. These are the vivid research areas involved:

- **PSF modeling:** We assume that the degradation is out-of-focus blur. In general, we can classify the image restoration schemes with respect to the PSF. If we have the PSF of a camera for this application, it is called '*classical image restoration*'. There are different methods e.g. Inverse filtering, Wiener Filtering, Least-square filtering and so on. Without the prior knowledge of PSF, the method is called '*blind image restoration*, e.g. Direct de-convolution, Recursive filtering, Neural network, Wavelet filters, etc. [3].

- **Statistical properties of the additive noise:** For simplicity, we can assume that it is white Gaussian noise.

- **Prior knowledge on the original image:** The prior knowledge improves the quality of the restoration scheme e.g. Markov Random Fields (MRF) [31].

- **Temporal data:** If we have a sequence of the low-resolution images(y's), we can restore the super-resolution image (x) and estimate the H [11, 18].

- **PDE-based technique:** In this method [45], we model the degradation process as:

$$I_N = I + \eta, \tag{6.2}$$

where to minimize:

$$\inf_{I \in BV(\Omega)} \int_{\Omega} \{(I - I_N)^2 + \alpha\phi(\frac{||\nabla I||}{\delta})\} d\Omega, \tag{6.3}$$

where α and δ are two constants, I_N is the noisy image, and ϕ is a function still to be defined. Notice that if $\phi(x) = x^2$, we recognize the Tikhonov regularization term. This method is well known to smooth the image isotropically without preserving discontinuities in intensity.

6.2 3D structure recovery from differential geometric properties

This problem stated in Chapter 4 can be extended to the real applications. So far, our algorithm is able to perform on the inverse problem using the shooting method bases on one parameter. As we claimed, we have to find the minimum of the objective function J in Eqn. (4.59). This function is not smooth enough for the real images, thus the results on the real images were not accurate. We will augment the extra initial constraint to the initial constraint equation set. Such condition is the smoothness of the rate of change of the characteristic slope at the starting point:

$$\dot{m}(0) = \frac{-h_u(0)}{h_v(0)} = 0 \tag{6.4}$$

Also, the major step for the future work of the 3D structure recovery is the sensitivity analysis of our method.

6.3 Knowledge-driven OCR

Database of text in the environment: We will gather a database consisting of digitized samples of reading material for each task and characterize the distributions of print parameters (e.g., size, font, contrast, color, background pattern, etc.) for each task. it has already been constructed a similar database for U.S. newspapers [22] and for product labels [8]. This database is necessary for development of the knowledge-driven OCR for tasks relevant to the target population.

Contextual Dictionaries: The words that appear in daily activities for the visually impaired to read the scene text are are from a very restricted vocabulary. We propose to use the domain knowledge to improve the recognition accuracy of the OCR subsystem. The knowledge will be represented as dictionaries and thesauri.

6.4 User interface

The proposed system in the real application for the visually impaired need a user interface capable of performing the analysis of the scene text. Therefore, the future work can be dedicated to set up a video-based interface consisting of a head-mounted camera with the loudspeaker on person's ears and a laptop for the processing part instead of the lab set-up we already built up. In this context, we can add hardware for the user input to the system e.g. a mouse or a joystick.

BIBLIOGRAPHY

[1] Aloimonos Y., Shape from texture, *Biological Cybernetics* vol. 58, pp 345-360, 1988.

[2] American Foundation for the Blind *http://www.afb.org*

[3] Jain A.K. , Fundamental of Digital Image Processing, *Prentice Hall Inc.*, 1989.

[4] Arkenstone Company. *http://www.arkenstone.org* provides a software (OPEN-Book) to turn your computer system into a scanning and reading machine, offering blind and vision-impaired individuals access to printed materials.

[5] Bookshare.org is a service for the visually impaired that provides access to books via a rights-managed subscription service.

[6] Bovik A., The Handbook of Image and Video Processing In *Academic Press*, 2000.

[7] Bouguet J.Y., Camera Calibration toolbox in MATLAB and C, In *http://www.vision.caltech.edu/bouguetj.*

[8] Braudway, S.M. and Massof R.W., Visual requirements of reading: Distributions of print sizes for consumer product labels, *Investigative Ophthalmology and Visual Science Supplement,35:1554*,1994.

[9] Brown L.G.,A Survey of Image Registration Techniques In *ACM Computing surveys, VOL 24 pp 325-376*, 1992.

[10] Brown M.S. and Seales W.B., Document restoration using 3D shape: A general deskewing algorithm for arbitrarily warped documents In *International Conference on Computer Vision, ICCV 2001*, 2001.

[11] Capel D. and Zisserman A.(2000) Super-resolution Enhancement of Text Image Sequence In: International Conference in Pattern Recognition, Vol I.

[12] Cheng L. and Robinson J., Dealing with Speed and Robustness Issues for Video-Based Registration on a wearable Computing Platform In *IEEE Cs Press, Los Alamitos, California, pp 84-91*, 1998.

[13] Clarke J.C. , Carlsson S. and Zisserman A., Detecting and tracking linear features efficiently In *Proceedings of the British Machine Vision Conference*, 1996.

[14] Clark P. and Mirmehdi M. (2001) Estimating the orientation and recovery of text planes in a single image In: Proceedings of the British Machine Vision Conference.

[15] Clark P. and Mirmehdi M. (2002) Recognising Text in real scenes In: International Journal of Document Analysis and Recognition (IJDAR), pp 243–257.

[16] Clerc M. and Mallat S. (1999) Shape from texture deformations In: Int. Conf. Computer Vision, pp 405–410.

[17] Comaniciu D. and Meer P (1999) Mean Shift Analysis and Applications In: IEEE International Conference on Computer Vision, pp 1197–1203.

[18] Cortijo F.J., Villena S., R. Molina, and A.K. Katsaggelos, Bayesian Super resolution of Text Image Sequences from Low-resolution Observations, *ISSPA 2003, vol. I, 421-424, Paris (France)*, July 2003.

[19] Davis L.S. and Duriswami R., Textual information access for the visually impaired. *Proposal funded by NSF*, April 2000.

[20] Devernay F. (1995) A non-maxima suppression method for edge detection with sub-pixel accuracy Technical report RR 2724, INRIA.

[21] Dellaert F., Seitz S., Thorpe C., and Thrun S., Structure from Motion without Correspondence. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'00)* , June 2000.

[22] DeMarco L.M. and Massof R.W. , Visual requirements of reading: Distributions of print sizes in U.S. newspapers In *Journal of Visual Impairment and Blindness, 91:9-13*, 1997.

[23] Do Cormo M., Differential Geometry of Curves and Surfaces. *Prentice Hall*, 1976.

[24] Doermann D., Liang J. and Li H.(2003) Progress in Camera-based Document Image Analysis In: 7th International Conference on Document Analysis and Recognition Volume I, pp 606–617.

[25] Faugeras O., Three Dimensional Computer Vision. *MIT press*, 1990.

[26] Faugeras O. (1995) Stratification of 3-D vision: projetcive, affine, and metric representations In: Optical Society of America Vol 12, No. 3. pp 465–484

[27] Ferreira S., Thillou C., Gosselin B.(2003) From Picture to speech: an innovative OCR application for embedded environment In: Proc. of the 14th ProRISC workshop on Circuits, Systems and Signal Processing (ProRISC 2003).

[28] Foroosh H. (Shekarforoush), Zerubia J. and Berthod M. (2002) Extension of Phase Correlation to Sub-pixel Registration In: IEEE Trans. Image Processing, vol. 11, Issue 3.pp. 188-200

[29] Forsyth D.A. and Ponce J.(2003) Computer Vision: A Modern Approach, Prentice Hall.

[30] Garding J., Surface orientation and curvature from differential texture distortion In *International Conference on Computer Vision, ICCV95*, June 1995.

[31] Geman D. and Reynolds G., Constrained Restoration and the Recovery of Discontinuities, *IEEE-PAMI,vol 14, No 3*, pp 367–383,1992.

[32] Gibson J., The perception of the Visual world. In *Boston: Houghton Mifflin*, 1950.

[33] Gray A., Modern Diffrential Geomtery of curves and surfaces with MATHEMATICA In *CRC Press*, 1998.

[34] Gumerov N. , Zandifar A., Duraiswami R. and Davis L.S. (2004) Structure of Applicable Surfaces from Single Views European Conference on Computer Vision (ECCV2004).

[35] Hartley R.I., Theory and practice of projection rectification In *International Journal of Computer Vision, Vol.2, No. 35, pp 1-16*, Nov 1999.

[36] Hartley R. and Zissermann A. (2000) Multiple View Geometry in Computer Vision, Cambridge Press.

[37] Jain A.K. and Bhattacharjee S. (1992) Text segmentation using Gabor filters for automatic document processing In: Machine Vision and Applications archive Volume 5 , Issue 3 1992, pp 169–184.

[38] Jain A.K., Fundamentals of Digital Image Processing In *Prentice Hall*, 1989.

[39] Kang-Sun Ch., Jun-Suk L. and Sung-Jae K., New Auto-Focusing Technique using the Frequency Selective Weighted Median filter for video cameras In *IEEE transactions on Consumer Electronics*, Aug. 1999 pp 1127–1131.

[40] Kanungo T., Haralick R., and Phillips I., Nonlinear Local and Global Document Degradation Models In *Int'l. Journal of Imaging Systems and Technology, Vol. 5, No. 4, pp 220-230*, Sep 1994.

[41] Kergosien Y.L., Gotoda H. and Kunii T.L., Bending and creasing virtual paper In *IEEE Computer graphics and applications, Vol. 14, No. 1, pp 40-48*, Jan 1994.

[42] Koenderink J.J., What Does the Occluding Contour Tell us About Solid Shape? *Perception,* 13: 321-330, 1984.

[43] Koenderink J.J., *Solid Shape*, MIT Press, 1990.

[44] Korn G.A. and Korn T.M., Mathematical Handbook for scientists and engineers In *Dover Publications, Inc.*, 1968.

[45] Kornprobst P., Deriche R. and Aubert G., Nonlinear operators in image restoration, *CVPR 1997.*

[46] Kovesi P., Shapelets correlated with surface normals produce surfaces In *TR 03-002 University of Western Australia* , 2003.

[47] Kuglin C. and Hines D.(1975) The phase correlation image alignment method In: Proc. of Int. Conf. Cybernetics Society, Vol. 12. pp 163–165

[48] Liebowitz D. and Zisserman A. (1998) Metric Rectification for Perspective Images of Planes In: IEEE Computer Vision and Pattern Recognition Conference. pp 482–488

[49] Liebowitz D. (2001) Camera Calibration and Reconstrcution of Geomtery from Images In: Doctor of Philospphy Dissertation. Universirty of Oxford

[50] Lienhart R. and Wernicke A. (2002) Localizing and segmenting text in images and videos In: IEEE Transaction on Circuits and Systems for Video Tech. Vol12, N0. 4, pp 256–268.

[51] Mann S., Brook P. and Fogarty S., Goals for supporting the education needs of the visually impaired with an integrated reading device. *Digital proceedings*, 1999.

[52] Malick J. and Rosenholtz R., Computing local surface orientation and shape from texture for curved surfaces, *Int. J. Computer Vision*, 1997.

[53] Marr D., Vision: A Computational investigation into the human representation and processing of visual information, *Freeman*, 1982.

[54] Massof R.W., A Systems Model for Low Vision Rehabilitation. I. Basic Concepts., *Optometry and Vision Science, 1995, vol 72 (10), pp 725-736*, pp 149-168, 1995.

[55] Massof R.W., A Systems Model for Low Vision Rehabilitation. II. Measurement of Vision Disabilities,*Optometry and Vision Science, 1998, vol 75 (5), pp 349-373*, 1998.

[56] McInerney T. and Terzopoulos D., Deformable Models in Medical Image Analysis: A Survey, *Medical Image Analysis, 1(2), 1996*, pp 91-108.

[57] McIvor A. M., Robust 3D Surface Property Estimation In *Second Asian Conference on Computer Vision, Vol. 2, pp 275-279*, Dec 1995.

[58] Mirmehdi M., Clark P. and Lam J., Extracting Low Resolution Text with an active Camera for OCR In *Proceedings of the IX Spanish Symposium on Pattern Recognition and Image Processing, pp 43-48*, 2001.

[59] Mirmehdi M., Palmer P.L. and Kittler J. (1997) Towards Optimal Zoom for Automatic Target Recognition In: Proc. of 10th SCIA, Vol. I.pp 447–453

[60] Penna M.A., Non-rigid Motion Analysis: Isometric Motion In *CVGIP: Image Understanding, Vol. 56, No. 3, pp 366-380*, Nov 1992.

[61] Pilu M., Extraction of illusory linear clues in perspectively skewed documents In *IEEE Computer Vision and Pattern Recognition Conference*, Dec 2001.

[62] Pilu M., Undoing Page Curl Distortion Using Applicable Surfaces In *Proc. IEEE Conf Comouter Vision Pattern Recognition*, Dec 2001.

[63] National Fedreation of the Blind Newsline *http://www.nfb.org/newsline1.htm* is a service that provides telephone access to content of newspapers and magazines.

[64] Newman W., Dance C. , Taylor A., Taylor S., Taylor M. and Aldhous T., CamWorks: A Video-based Tool for Efficient Capture from Paper Source Documents In: Procedeeings of ICMCS. pp 647–653.

[65] Open Computer Vision (OpenCV) Library developed by Intel Image Processing http://www.intel.com/mrl/research/opencv.

[66] Poleman C.J.and Kanade T., A Paraperspective Factorization Method For Shape And Motion Recovery, *PAMI(19)*1997.

[67] Portilla J. and Simoncelli E.P., A Parametric Texture Model based on Joint Statistics of Complex Wavelet Coefficients, *Int. Journal of Computer Vision. 40(1)* pp 49–71, 2000.

[68] Press W.A., Teukolsky S.A., Vetterling W.T. and Flannery B. P., *Numerical Recipes in C,* Cambridge University Press, 1993.

[69] Press W.H., Teukolsky S.A., Vetterling W.T. and Flannery B.P. . Numerical Recipes in C : The Art of Scientific Computing In *Cambridge University Press*, Jan 1993.

[70] Reddy B.S. and ChatterjiB.N., An fft-based technique for translation, rotation and scale invariant image registration In *IEEE Transaction on Image Processing*, 1996.

[71] Riberio E. and Hancock E.R., 3-D planar orientation from texture: estimating vanishing points from local spectral analysis In *Proceedings of the British Machine Vision Conference*, 1989.

[72] Rosenholtz R. and Malik J., Computing Local Surface Orientation and Shape from Texture for Curved Surfaces,*International Journal of Computer Vision*, vol. 23 pp 149-168, 1997.

[73] Sauvola J. and Pietkainen M., Page Segmentation and Classification using fast Feature and Connectivity Analysis In *Proc. of the 3rd International Conference on Document Analysis and Recognition, Montreal, Canada pp 1127-1131*, 1997.

[74] Scansoft2000 (OCR software) http://www.scansoft.com/devkit/docimage.asp.

[75] Scholl G. and Schur R., Measures of Psychological, Vocational and Education Functioning in the Blind and Visually Handicapped In *American Foundation for Blind(AFB)*, 1976.

[76] Seeger M. and Dance Ch.(2001), Binarising Camera Images for OCR In: Procedeeings of 6th ICDAR. pp 54-59.

[77] Szeliski R., Image Mosaicing for Tele-reality Applications In *IEEE Workshop on Applications of Computer Vision, pp 44-53*, 1994.

[78] Taylor M.J. and Dance C.R. (1998) Enhancement of Document Images from Cameras In: Proc. IS & T/SPIE EIDR V. pp230–241

[79] Tekalp A.M., Digital Video Processing In *Prentice Hall*, 1995.

[80] Torras C.. Computer Vision: Theory and Industrial Applications In *Spring-Verlag*, 1992.

[81] Torkkola K., Discriminative features for document classification In: Proceedings. 16th International on Pattern Recognition, Volume I. pp 472–475

[82] Trier O.D. and Taxt T. (1995) Evaluation of Binarization Methods for Document Images In: PAMI Vol 17, No.3. pp 312–315

[83] Van Hateren J.H. and Van der Schaaf A., Independent component filters of natural images compared with simple cells in the primary visual cortex In: Proc. R. Soc. Lond. B. pp 359–366

[84] Wallick M.N., Lobo N.D.V. and Shah M., Computer Vision Framework for Analyzing Projections from Video of Lectures In: Proceedings of the ISCA 9th International Conference for Intellegent Systems.

[85] Wallick M.N., Lobo N.D.V. and Shah M., A System for Placing Videotaped and Digital Lectures Online In: IEEE 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing (ISIMP).

[86] Wang Y.F., Mitchie A. and Aggarwal J.K., Computation of surface orientation and structure of objects using grid coding In *IEEE trans. on pattren analysis and machine intelligence, Vol. 9, No. 1, pp 129-137*, Jan 1987.

[87] Whitham G.B., *Linear and Nonlinear Waves*, New-York: Wiley, 1974.

[88] Witkin A.P., Recovering Surface Shape and Orientation from Texture, In *Artificial Intelligence, vol. 17*, pp 17-45, 1981.

[89] Wolberg G., Digital Image Warping, *Wiley-IEEE press*, 1990.

[90] Worthington Ph. L. and Hancock E.R., Needle map recovery using robust regulizer In *Image and Vision Computing, Elsevier Vol. 17, pp 545-557* , 1999.

[91] Wu V., Manmatha R. and Riseman E.M. (1999) extFinder: An Automatic System to Detect and Recognize Text in Images In: PAMI(21), No. 11, November 1999, pp. 1224-1229.

[92] You Y., Lee J. and Chen Ch., Determining location and orientation of a labeled cylinder using point pair estimation algorithm In *International Journal of Pattern Recognition and Artificial Intelligence, Vol. 8, No. 1, pp 351-371*, 1994.

[93] Zandifar A., Chahine A., Duraiswami R. and Davis L.S. (2002) Video-based Interface to Textual Information for the Visually Impaired In: IEEE Computer Society, Internation Conference on Multimodal Interfaces (ICMI). pp 325–330.