MEASUREMENT OF SKIN DEFORMATION USING FINGERPRINT FEATURE TRACKING

Vincent Levesque

Department of Electrical and Computer Engineering
McGill University, Montréal

November 2002

A Thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment
of the requirements for the degree of Master of Engineering

ABSTRACT

This thesis describes an experimental platform for the study of skin stretch of the human fingerpad during tactile exploration tasks. A digital camera records the sequence of patterns created by a fingertip as it slides over a transparent surface with simple geometrical features. Skin deformation is measured with high temporal and spatial resolution by tracking anatomical landmarks on the fingertip. Techniques adapted from the field of online fingerprinting are used to acquire high-contrast fingerprint images and extract salient features (pores, valley endings, and valley bifurcations). The performance of the method is evaluated with surfaces embossed with a ridge or valley and flat surfaces. This work is motivated by the need to provide meaningful ‘tactile movies’ for a tactile display using distributed lateral skin stretch.

RÉSUMÉ

Cette thèse décrit une plate-forme expérimentale pour l'étude de l'étirement de la peau du doigt humain durant l'exploration tactile. Une caméra digitale enregistre une séquence d'empreintes créées par un doigt glissant sur une surface transparente comportant des formes géométriques simples. La déformation de la peau est mesurée avec haute résolution temporelle et spatiale en traquant des points de repère anatomiques du doigt. Des techniques adaptées du domaine de l'analyse en-ligne d'empreintes digitales sont utilisées pour acquérir des images à fort contraste de l'empreinte digitale et pour extraire des traits caractéristiques saillants (pores, terminaisons de vallée et bifurcations de vallée). La performance de cette méthode est évaluée avec des surfaces comportant une crête ou un sillon et avec une surface plane. Ce travail est motivé par la nécessité de produire des « films tactiles » significatifs pour un afficheur tactile utilisant l'étirement latéral distribué de la peau.

ACKNOWLEDGEMENTS

This work would not have been possible without the encouragement and guidance of my supervisor, Prof. Vincent Hayward. His enthusiasm and optimism made this experience both rewarding and enjoyable. Most of the novel ideas and solutions found in this thesis are the result of our numerous stimulating discussions. His feedback and editorial comments were also invaluable for the writing of this thesis.

I also wish to thank my colleagues at McGill's haptics laboratory. Jérôme Pasquero has been a great friend and colleague, providing constant support and helpful suggestions. This work would not have been nearly as interesting without our collaboration. Gianni Campion offered expert advice and proposed numerous improvements to the calibration methodology that will likely be implemented in the near future. Many thanks also to Dr. Ian Sinclair for his help in the writing of this thesis. Thanks also to Q. Wang, S. Harashima, M. Mahvash, D. Yi, H. Dostmohamed, H.-Y. Yao, and S. Mathur for making the haptics lab an enjoyable and stimulating work environment.

I also wish to thank Mike Faulkner (Bock Optronics) and Seed Doric (Doric Lenses). Many thanks to Don Pavlasek and Jozsef Boka of the mechanical workshop (ECE department) for their dedicated work on mechanical parts used in this work. Eddie Del Campo of the physics machine shop also manufactured early prototypes of the transparent surfaces.

Finally, I wish to thank my parents, Suzanne and Viateur Levesque, and my sister, Danny Levesque, for their support throughout my Master's degree. My late grandfather, Fernand Parent, encouraged me to pursue graduate studies. I dedicate this thesis to him.

Financial support from the Institute for Robotics and Intelligent Systems (IRIS) and the Natural Sciences and Engineering Council of Canada (NSERC) is gratefully acknowledged.

CLAIM OF ORIGINALITY

To the author's knowledge, the following constitute original contributions of this thesis:

- An optical platform for the measurement of skin deformation through transparent surfaces.
- A technique for the measurement of skin stretch of the fingerpad using anatomical landmarks of the fingertip.
- A method to track fingerprint features between consecutive images in a sequence.
- A mapping algorithm converting local skin motion vectors to the position of skin contactors of a tactile display using distributed lateral skin stretch.
- A practical implementation of the above.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1 INTRODUCTION

1.1 Motivation

Tactile displays are devices capable of applying programmable mechanical stimuli to the fingerpad of a user. Tactile displays are conceptually similar to their visual counterparts (such as CRT monitors), which have become ubiquitous over the past century largely due to a good understanding of the sense of vision. Many visual illusions, such as the fusion of a discrete sequence of images into a continuous stream, have been exploited to enable the design of effective visual displays. Despite much research, touch is comparatively little understood. While the nature of the stimulus to which vision responds is well known, the exact nature of the relation between mechanical stimuli of the skin and tactile perception is the subject of debate among researchers. This lack of understanding, as well as numerous technical challenges, make the design of an effective tactile display an arduous task that has yet to be achieved satisfactorily.

The work that follows is motivated in part by the need to improve our understanding of the mechanical behaviour of the fingerpad and of the relation between mechanical stimuli and tactile perception. The immediate motivation, however, is the need to generate driving signals for the STReSS tactile display [25]. The STReSS is a tactile display that relies solely on distributed lateral skin stretch to cause tactile sensations. This novel concept was inspired by the observation that lateral skin stretch seems to be capable of giving the sensation of shape or texture. The driving signals of the tactile display will be referred to as a sequence of ‘tactile images’ or ‘tactile movie’ in this thesis. Generating a perceptually meaningful tactile movie is not trivial. One approach, akin to computer graphics in vision, consists of rendering a tactile movie using a model of fingertip deformations. This approach is limited by the absence of a definitive model of the dynamic properties of the fingerpad in contact with a surface. Another approach, akin to television systems, consists of observing and measuring fingerpad deformations during an exploration task. The recorded tactile movie can be played on the tactile display to reproduce the tactile sensation felt during the exploration task. This tactile movie acquisition/display method can ultimately be used to improve our understanding of the mechanical properties of the fingerpad and of tactile perception. The latter approach is adopted in this work.

1.2 Overview

This thesis proposes a technique to measure the deformations of the skin of a human fingerpad during a tactile exploration task. A digital camera records the patterns created by a fingertip as it moves across a transparent surface with geometrical features such as ridges and valleys. The skin deformation is measured with high temporal and spatial resolution by tracking anatomical landmarks on the fingertip. The technique draws from the expertise developed in the field of online fingerprinting for the acquisition and analysis of fingerprint images.

Images of the fingerprint are acquired with a right-angle prism using the principle of frustrated total internal reflection (FTIR). In the absence of contact, light shone through a face of the prism is reflected by its hypotenuse face. Where fingerprint ridges contact the hypotenuse face the reflection of light is prevented, thereby forming a dark, high-contrast pattern. The traditional prism-based fingerprint acquisition was adapted to the current application by adding a Lambertian diffuser on the entry face. This modification maintains reflection even with non-flat parallel ruled surfaces.

The resulting sequence of images is processed to extract fingerprint features. Fingerprint matching generally relies on minutia called ridge endings and ridge bifurcations. Pores are occasionally used to improve accuracy. Here all three types of feature are extracted using image processing techniques. The process consists of smoothing, binarizing and thinning the fingerprint pattern. The resulting skeleton is analyzed to locate the valley endings, valley bifurcations and pores. Syntactic processing is then applied to correct for artefacts that result from imperfect contact between the fingertip and the surface. Pore extraction can optionally be performed more accurately by connected-component analysis of the binary fingerprint.

The features are then tracked through the sequence of images using a simple matching algorithm. Two methods are proposed to study changes in local skin stretch using tracked features. A first method relies on the changes in triangle area or edge length of a Delaunay triangulation of tracked features. A second method relies on the local movement of patches of skin with respect to the fingertip. The latter method can be used to map patches of skin to the STReSS tactile display's contactors, thereby creating a mapping for 'tactile movies'.

An optical bench was assembled for the acquisition of images. Testing to date was done with three shapes: a flat surface, a surface with a small ridge and a surface with a small valley. A software application was written to perform image analysis and measurements. The application includes a fully featured graphical-user interface facilitating debugging, parameter adjustment, and measurement visualization.

1.3 Thesis Outline

Chapter 2 provides an overview of previous work relevant to this thesis. Chapter 3 describes in details the design and implementation of the fingertip image acquisition apparatus as well as some of its limitations. Chapter 4 describes the image processing algorithms used to extract fingerprint features. Chapter 5 describes the algorithms used to track features and extract skin stretch information. Chapter 6 provides the results of experiments that test the capabilities and limitations of the technique. Chapter 7 ends this thesis with a discussion of possible improvements as well as concluding remarks.

CHAPTER 2 LITTERATURE REVIEW

The work presented in this thesis touches upon numerous fields of research. This chapter attempts to provide an overview of these fields with a focus on the information relevant to this thesis. The literature review begins with a discussion of tactile displays. Follows an overview of current knowledge on fingertip properties and touch. The chapter closes with a brief overview of fingerprint recognition techniques, which are the basis for some of the image acquisition and image processing performed in this work.

2.1 Tactile Displays

Many tactile displays rely on displacement or pressure normal to the skin to convey tactile sensations (e.g. [19][22][24][35]). Another approach consists of relying on tangential displacement or lateral skin stretch. Although less intuitive, lateral stress may be sufficient to provide the sensation of a shape or texture. Figure 1 illustrates a convincing experiment that illustrates the principle [16]. Hold a comb in your left hand with the index touching the teeth of the comb as illustrated. As you slide a sharp object (such as a pen) on the side of the comb with your right hand, attend to the tactile sensation caused on your left-hand index. Although most of the pin movement occurs laterally, a moving bump is distinctly felt. In this case, lateral stress seems to be causing the illusion of normal indentation.

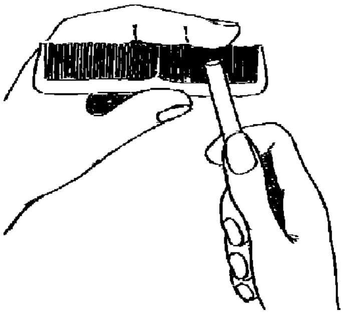
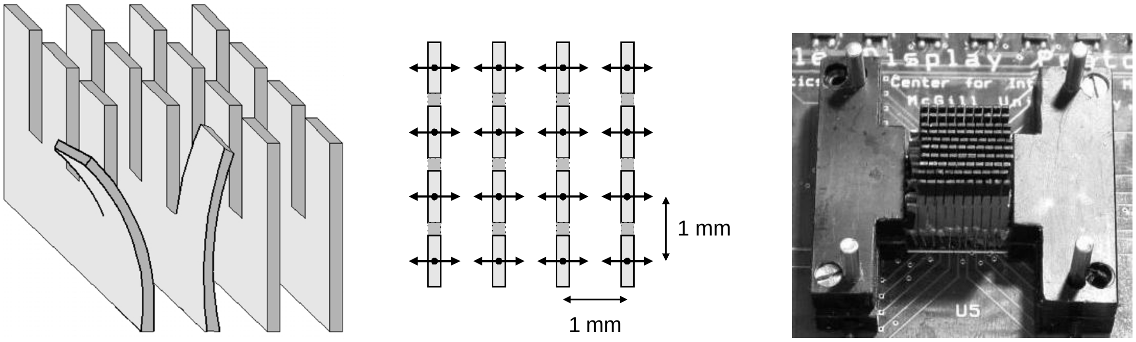


Figure 1. Comb analogy (from [16]).

McGill's Haptics Laboratory began efforts to design a tactile display using distributed lateral skin stretch in 1997. Hayward and Desmarais designed a first prototype consisting of a row of 8 skin contactors moving laterally under the effect of perpendicular piezoelectric actuator movement [10]. Encouraging results with the 1D tactile display led to the design of a second prototype by Hayward and Cruz-Hernández consisting of a 2D array of skin contactors using similar principles [16]. A third prototype is currently under development by Hayward and Pasquero [25]. This latest prototype consists of an arrangement of 10 comb-shaped sheets of piezoelectric material as illustrated in Figure 2. Each tooth of the combs can 'bend' in the direction perpendicular to the sheet. The piezoelectric teeth are also used as skin contactors. The result is an array of 10x10 skin contactors moving laterally in one dimension. The movement of each tooth can be controlled independently resulting in a programmable lateral strain field. Inter-contactor spacing is 1 mm and maximum displacement is approximately ±50 μm.



(a) (b) (c)

Figure 2. STReSS tactile display: (a) bending skin contactors (4x4), (b) array of skin contactors (4x4), and (c) picture (10x10).

The driving signals necessary to cause meaningful sensations with these devices are not obvious. Previous attempts used an empirical approach. A graphical user interface was written to control parameters of space- and time-varying signals [16]. The parameters and types of signals were varied in the hope of discovering interesting tactile sensations. Some non-intuitive perceptual phenomenon were observed using this technique. The work presented here uses an alternative approach that aims to create tactile movies from direct observation of fingerpad deformations during tactile exploration.

2.2 Fingertip Properties

2.2.1 Anatomy

Touch is experienced through skin, a complex organ that covers the entire surface of the body. While most skin is of the hairy type, the palms and the fingers of the hand are covered with glabrous skin. Our interest is in the latter. The skin of the fingertip “exhibits complex mechanical behavior such as nonlinear force-displacement relationship under indenting probes, anisotropy, and rate and time dependence” [32]. The fingertip consists of a bone covered with fatty tissue and two layers of skin. The outer layer is called epidermis and the inner layer is called dermis. Papillae at the interface between the two layers form ridges at the surface of the epidermis [20].

Four types of embedded mechanoreceptors give the glabrous skin its sensitivity. Table 1 describes the four mechanoreceptors along with their receptive field size, best stimulus and type of fibre. SA I and SA II mechanoreceptors are slowly adapting which means that they respond to constant stimulus. RA I and PC mechanoreceptors, on the other hand, quickly stop responding unless the stimulus varies. It is believed that the location of mechanoreceptors and the presence of papillae and ridges plays a part in the sensitivity of the skin [20]. The work presented in this thesis tracks features of the papillary ridges.

Table 1: Types of mechanoreceptor fibres (from [14]).

Type of fibre	**Receptive field size**	**Receptor structure**	**Best stimulus**
SA I	Small	Merkel receptors	Pressure
RA I	Small	Meissner corpuscle	Taps on skin
SA II	Large	Ruffini cylinder	Stretching of skin or movements of joints
PC	Large	Pacinian corpuscle	Rapid vibration

2.2.2 Measurements and Models

Much work was done in the recent past to observe, measure, and model the mechanical characteristics of the fingertip. This section gives a brief overview of some of this work.

Many of the existing fingertip models only consider deformations of the skin under a static load. The absence of a complete model of the dynamic properties of the fingertip in contact with surfaces justifies the deformation measurement approach taken in this work.

Srinivasan proposed a ‘waterbed’ model of the fingertip consisting of a thin membrane enclosing incompressible fluids [31]. The predictions of the model were compared with pictures of skin indentation under a line load. The model accurately predicted deformations of the human and primate fingertip near the load. Srinivasan and Dandekar later developed four models of the primate fingertip using finite element methods [32]. The most complex model assumes a cylindrical shape with a rigid fingernail covering a third of its surface, and a rigid bone in its interior. The models were evaluated using the result of previous experiments with line, rectangular bar and aperiodic grating loads. The models were found to be less accurate than the ‘waterbed’ model. They were shown, however, to predict neural responses of mechanoreceptors more accurately. Pawluk and Howe also studied the dynamic, distributed pressure response of the fingertip as it is loaded by a flat surface and developed a model based on their observations and measurements [26].

Dandekar and Srinivasan used videomicroscopy to observe and measure deformation of the skin under various loads [9]. The fingertip of a subject was covered with approximately 100-150 markers applied using a micro tip pen. Images of the fingertip were acquired as a load (such as a rectangular or cylindrical bar) was lowered on the fingertip in a controlled manner. Markers were located manually in the images and used to verify model predictions. Observations were limited to deformations resulting from static loads. Figure 3 shows an example of the technique. The MIT Touch Lab also performed experiments to study the dynamics of fingertip/object contact [39].

Using a different approach, Dorai, Ratha and Bolle relied on local motion data embedded into an MPEG-{1,2} video stream to detect distortion in a sequence of fingerprint images [11]. An MPEG-1 or MPEG-2 video stream contains information about the inter-frame movement of small blocks of pixels called macroblocks. They developed a technique that recovers the motion vector of most of the macroblocks directly from the compressed video stream. The resulting flow vector map is analysed to detect frames with significant distortion. Selecting a frame with low distortion improves the performance of fingerprint matching systems by reducing the effect of elastic distortions of the skin and variations in pressure.

Cappelli, Maio and Maltoni proposed a model of non-linear deformations occurring during fingerprint acquisition [8]. The contact area between the fingertip and the sensor was segmented in three regions as shown in Figure 4. The inner region, firmly pressed against the surface, is stationary. The outer region (shown in black), in light contact with the surface, moves freely with the fingertip. The intermediate region (shown in gray) stretches and compresses itself to join the inner and outer regions. A model that predicts deformations due to torque and traction was proposed. The model was successfully tested by comparing its predictions with actual fingerprint deformations.

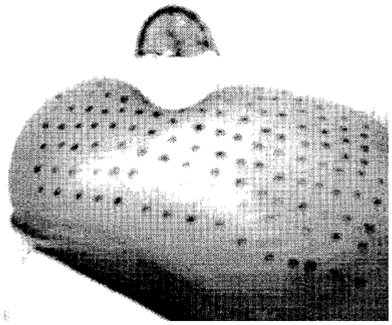


Figure 3. Cylinder indenting a fingertip (from [9]).

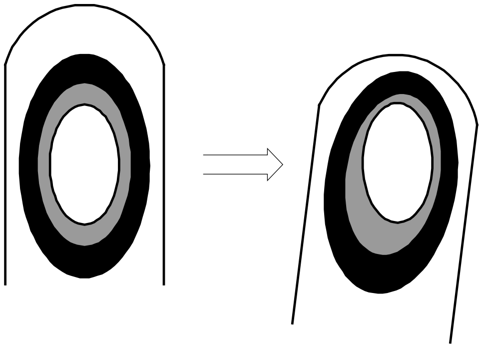


Figure 4. Fingerprint deformation model (adapted from [8]).

2.3 Fingerprinting Recognition

Fingerprint acquisition and matching are tedious tasks traditionally performed by trained professionals. In the past decades, however, the need for automation has led to research on the development of online fully- or semi-automated fingerprinting systems. In recent years, the demand for biometric applications such as access control has also increased the interest in online fingerprinting.

Online fingerprinting can be divided in two main problems: fingerprint verification and fingerprint identification. Fingerprint verification compares two fingerprints and assesses their similarity. Fingerprint identification consists of finding the closest match (or matches in the case of a semi-automated system) for a fingerprint given a fingerprint database. Access control applications typically fall in the verification category while law enforcement applications fall in the identification category. Our discussion will focus on fingerprint verification.

The block diagram of a typical fingerprint verification system is shown in Figure 5. The process begins with the acquisition of an image of the physical fingerprint. The fingerprint is then analysed to extract salient features. The identification information is used to fetch a corresponding set of features from a database. A matching algorithm compares the features of the input fingerprint with those of the enrolled fingerprint and determines the amount of similarity, leading to a decision (matching/not matching).

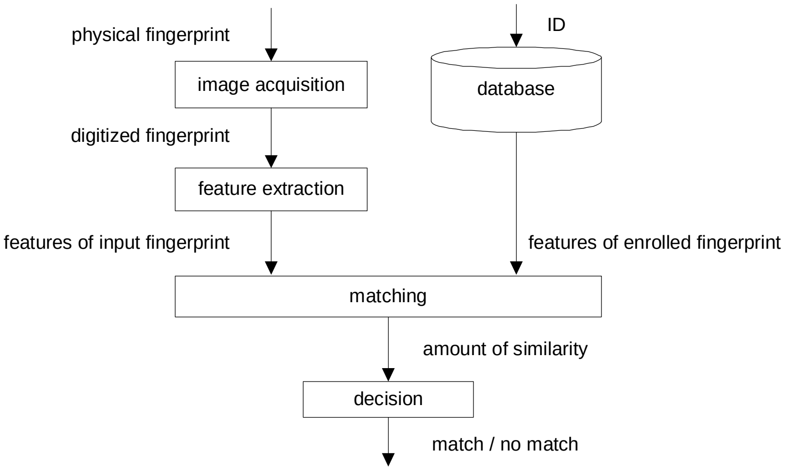


Figure 5. Block diagram of a typical fingerprint verification system (adapted from [15]).

The remainder of this section gives a brief overview of the different techniques used for each part of a fingerprint verification system. Section 2.3.1 describes fingerprint image acquisition techniques. Sections 2.3.2 and 2.3.3 describe typical feature extraction and fingerprint matching techniques respectively. Section 2.3.4 gives an overview of the use of pores in fingerprint verification. Readers are referred to [15][18] for more complete surveys of fingerprinting technologies.

2.3.1 Fingerprint Image Sensors

The availability of small, inexpensive and reliable fingerprint sensors is critical for online fingerprinting to gain wide acceptance. The problem has received much attention from academia and industry, resulting in numerous innovative solutions. A brief overview of available fingerprint capture devices is given here. Readers are referred to more comprehensive surveys for details [1] [18][38].

Optical techniques are the most common. A typical design relies on the principle of frustrated total internal reflection (FTIR) and consists of a light source, a right-angle prism and a camera [1] [18][38]. In the absence of contact, light enters one side of the prism and is totally reflected by its hypotenuse face toward the camera. When the fingertip is pressed against the hypotenuse face, the ridges of the fingerprint come in contact with the prism and prevent the total internal reflection from occurring. The result is a high-contrast pattern of black ridges over a white background. Various improvements have been proposed to address the main difficulties associated with this device, namely the size of the prism and the perspective view of the fingerprint. For example, a sheet prism can be used to reduce the size of the device [38], or holographic gratings can be used to eliminate the perspective projection [3].

Other types of optical sensors have also been proposed. Fujieda and Haga used a thin glass plate and a gradient-index fibre array to collect scattered light as opposed to reflected light [13]. Biometric Partners Inc.'s Touchless imaging technology captures the fingertip image directly without any contact between the fingertip and the sensor [7]. Ethentica commercializes an electro-optical sensor consisting of a thin sheet of polymer with embedded light-emitting and light-sensitive layers [12].

Solid-State sensors are manufactured on a single microelectronic chip. They can be divided in two main classes: thermal sensors and capacitive sensors. Thermal sensors use a pyroeletric material to sense the difference in temperature between fingerprint ridges and valleys. The fingerprint image quickly fades as the sensor reaches the same temperature as the fingertip. To overcome this problem, Atmel manufactures a thermal sensor as a thin strip and requires the user to swipe his finger on the device [2]. Capacitive sensors consist of an array of capacitor plates [1]. The skin of the fingertip acts as the second plate of the capacitors. The capacitance at each location is a function of the skin distance and thus indicates the presence of a ridge or a valley.

Ultrasonic sensors typically probe the fingertip using ultrasound emitters and receptors. Mechanical movement of the apparatus is generally required to scan the entire fingerprint. Ultrasound scanners were found to be expensive and slow [1]. It isn't clear whether this is still true of recent devices such as those sold by Ultra-Scan [34]. Other innovations, such as 3D laser scanning [5][18], may be forthcoming.

2.3.2 Feature Extraction

The feature extraction process analyses the fingerprint image to extract salient features for use in matching. Human experts usually rely on "local discontinuities in the ridge flow pattern" called minutiae [15]. Automated fingerprint verification systems typically use only two types of minutiae: ridge endings and ridge bifurcations (see Figure 6). Some characteristics of the minutiae, such as the orientation of nearby ridges, are often recorded to facilitate matching.

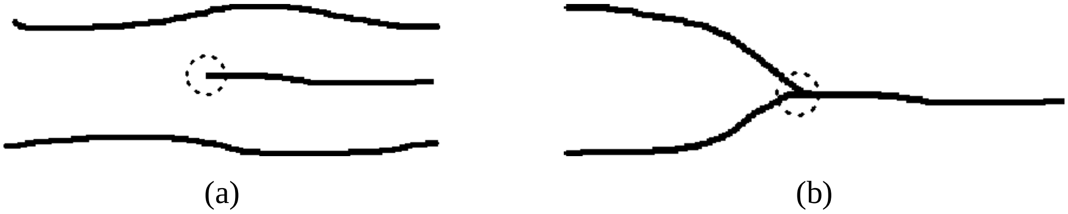

(a) (b)

Figure 6. Fingerprint minutiae: (a) ridge ending, and (b) ridge bifurcation.

Extracting minutiae reliably is impossible due to the imperfect contact between the fingertip and the sensor. Feature extraction systems use image processing operations that compensate the effect of poor image acquisition. A typical feature extraction system is shown in Figure 7. A direction map is often computed to estimate the orientation of ridges in small regions. The segmentation operation detects the part of the image that corresponds to the fingerprint. The variance or the local certainty of the direction map are often used to differentiate between background and fingerprint. The direction map is also often used to improve the greyscale or binary fingerprint by reinforcing the dominant ridge frequencies and orientations. Note however that this enhancement generally eliminates pores. The binarization operation is often performed with an adaptive threshold [30] or by finding the local maxima along the direction perpendicular to the local ridge orientation [17][18]. Thinning reduces the ridge pattern to a 1-pixel wide skeleton from which minutiae can easily be extracted. Imperfections in the skeleton often result in spurious minutiae which can be pruned using heuristic operations [15][30]. A registration point may also be extracted to facilitate the alignment of fingerprints for matching [15]. The registration point may be a singularity (a special pattern in the fingerprint) or any other distinctive feature.

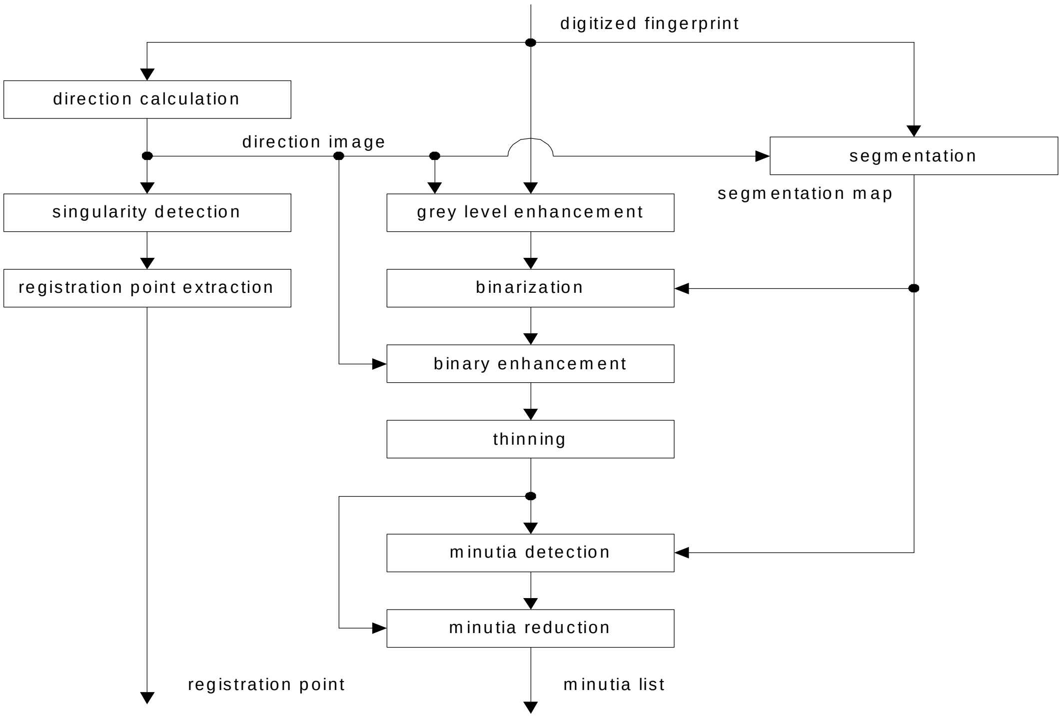


Figure 7. Block diagram of a typical feature extraction system (adapted from [15]).

2.3.3 Fingerprint Matching

Fingerprint matching entails comparing two fingerprints, or simplified fingerprint representations, to assess their degree of similarity. In a typical situation, the two fingerprints are acquired at different points in time, possibly with difference devices. The matching algorithm must handle translations and rotations of the fingerprint as well as a certain amount of non-linear deformation due to fingertip plasticity [17]. In the case of minutia-based algorithms, it is also necessary to handle spurious or missing minutiae resulting from imperfect imaging and/or feature extraction artefacts. For those reasons, fingerprint matching generally relies on local rather than global features. Fingerprint matching algorithms can generally be grouped in three categories: image-based, feature-based or a combination of the two [27]. Image-based techniques rely on the correlation of image segments for matching. The most common feature-based techniques rely on minutia matching. Although approximately 50 to 150 minutiae are present in an image, 10 matches are generally considered sufficient to determine that fingerprints are from the same finger [15]. The problem then becomes a graph or point pattern matching problem [15]. Extra information such as orientation of nearby ridges or structure of nearby minutiae is often used to improve matching accuracy. Other algorithms combine both approaches. For example, Stosz and Alyea matched salient image segments using image correlation techniques and confirmed matches using the local minutia structure [33]. Two minutia-based techniques are described next to give a better feel for the potential benefits of these techniques in the current work.

Bebis, Deaconu and Georgiopoulos used an approach based on the Delaunay triangulation for indexing and matching fingerprints in an identification context [4]. Features are built using a Delaunay triangulation of all minutia points. Each triangulation face is characterized by invariants based on edge lengths and internal angles. An indexing algorithm is then used to determine the best matches in a database. A match is evaluated by finding the similarity transformation that leads to the highest correspondence between the faces of the two fingerprints. The best matches are refined by computing the optimal affine transformation and evaluating the number of matching minutiae.

Ratha, Bolle, Pandit and Vaish matched fingerprints using a fingerprint representation called Minutia Adjacency Graph (MAG) [27]. Edges are formed between minutiae when the distance between them is less than a threshold. Each edge is characterized by the two vertices it connects, by its length and orientation, and by the number of ridges it crosses. Matching proceeds by first attempting to match the star-shaped neighbourhood of vertices. Reliable matches obtained with this method are then used to extend the matches with loosened rules.

2.3.4 Pores

Pores are small openings on the surface of the fingerprint ridges formed by the duct of sweat glands [29]. The width of pores varies from 88 to 220 μm [29] and their density is approximately 5 per mm^2 [30]. A minimum sensor resolution of approximately 31.5 pixel/mm is necessary to insure a sampling rate of twice the minimum width of a pore [33]. A typical sensor has a resolution of 19.7 pixel/mm [33].

Automated fingerprint matching techniques generally ignore or discard pores and rely exclusively on minutiae. Stosz and Alyea proposed the use of pores to increase matching accuracy [33]. A prism-based sensor acquires an image of 6.7 x 9.6 mm at a horizontal resolution of 95 pixels/mm and a vertical resolution of 50 pixels/mm. Pores and minutiae are extracted from the fingerprint image. The enrolment of fingerprints in the database proceeds by selecting and extracting segments of interest. The binary segments as well as the features they contain are stored and saved in a template. The matching algorithm attempts to correlate the segments stored in the template with the input fingerprint. The pore and minutia positions and orientations inside matched segments are used to confirm the match. Roddy and Stosz proposed a slightly modified technique [30].

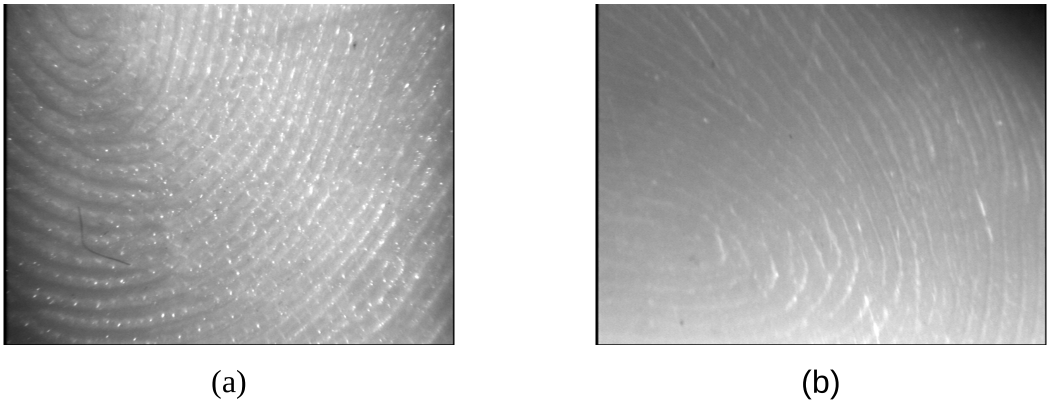
CHAPTER 3 FINGERTIP IMAGE ACQUISITION

This chapter describes a fingertip image acquisition platform for the measurement of skin deformation. It begins in section 3.1 with a re-evaluation of online fingerprinting technologies, leading to the selection of the prism-based fingerprint acquisition method. A detailed explanation of the principles and limitations of the method follows in section 3.2. Section 3.3 describes the platform assembled for experimental purposes. Section 3.4 and 3.5 describe the methodologies employed to correct geometric distortions and non-uniform lighting respectively.

3.1 Online Fingerprinting Devices

The online fingerprinting devices introduced in section 2.3.1 were designed for biometric applications with three requirements in mind: low cost, small size and high performance. The requirements of our application differ significantly. Small size and low cost are desirable but not critical. On the other hand, a high spatial resolution and a high frame rate are essential to locate pores precisely and capture high-frequency deformations of the skin. The device must also be capable of acquiring images through a surface. This section re-evaluates online fingerprinting technologies in light of these new requirements.

As shown in the next section, prism-based FTIR image acquisition can be adapted to our application with a few modifications and constraints on the contact surface. Improvements that reduce the size of the device, such as the use of a sheet prism, are of little interest given our requirements. Improvements that eliminate perspective projection, such as holographic gratings, are welcome but unlikely to be compatible with non-flat surfaces. Scattered light detection eliminates the perspective projection but is also likely to fail with non-flat surfaces. Moreover, our experiments and those of Fujieda and Haga have shown that scattered light detection results in specular reflections and low contrast images [13]. Figure 8(a) shows an example of scattered-light detection using a dove prism. Direct imaging of the fingerprint imposes few constraints on the contact surface but resulting images (see Figure 8(b)) tend to have a low contrast as well. It is unlikely that it could be used to locate small features such as pores. Tests showed that applying a high density of artificial tracking markers on the skin using ink or paint is impractical.


(a)

(b)

Figure 8. Imaging techniques: (a) scattered light detection, and (b) direct imaging.

The feasibility of shaping Ethentica’s polymer-based fingerprint sensor onto a simple embossed surface isn’t clear from available documentation. The maximum resolution and frame rate of the sensor are also unknown. Similarly, solid-state sensors are unlikely to be useable with non-flat surfaces. The direct contact between the fingertip and a capacitive sensor implies that its surface would need to be embossed. This is likely to be very difficult and costly, if at all possible. Although thermal sensors are capable of imaging through thin, uniform films such as gloves, it is unlikely that they could image through an arbitrary surface. Moreover thermal sensors are currently manufactured only as thin strips. Ultrasonic sensors are able to image through an opaque surface but their cost, complexity and frame rate is unclear. Bicz et al. proposed an ultrasonic fingerprint sensor capable of operating at a few frames per seconds with hopes of reaching 25 frames per second in the near future [6]. The frame rate of Ultra-Scan’s technology is not known at this time [34].

The prism-based optical approach was selected for its ease of implementation. Although other technologies may prove superior, their use would require a significant research effort with uncertain results.

3.2 Prism-Based Fingertip Image Acquisition

Prism-based fingerprinting was introduced briefly in section 2.3.1. A typical apparatus consists of a collimated light source, a right-angle prism and a camera as shown in Figure 9. Fingerprint valleys and pores leave an air gap between the fingertip and the hypotenuse face of the prism, resulting in the reflection of light rays. Ridges, on the other hand, come in contact with the prism and scatter light due to the principle of frustrated total internal reflection (FTIR) [36]. The camera sees a dark ridge pattern over a bright background. The fingertip is imaged at a 45° angle, resulting in a compression of the fingerprint in one dimension by a factor of $\sqrt{2}$. The perspective projection also requires the imaging system to provide a depth of field of at least 0.7 times the width of the field of view.

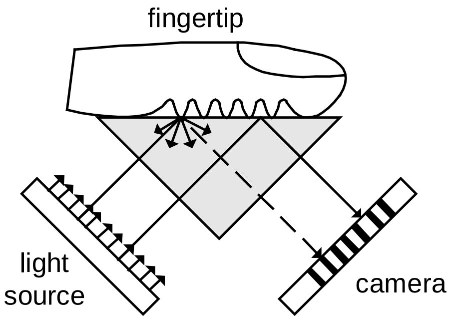


Figure 9. Prism-based fingerprinting.

FTIR prism-based imaging fails for non-flat surfaces. The surface's local gradient may be such that light strikes at an angle below the critical angle of the prism material and escapes the prism (Figure 10(a)). If striking above the critical angle, light is likely to be reflected away from the camera (Figure 10(b)). This results in dark patterns over regions of the surface with non-zero gradient (Figure 10(c)). This problem can be solved by insuring that for all positions on a surface, there is a ray coming from the light source that is reflected towards the camera. This can be accomplished by replacing the collimated light source with a Lambertian light source. A near Lambertian light source can be obtained by placing a diffuser on the entry face of the prism.

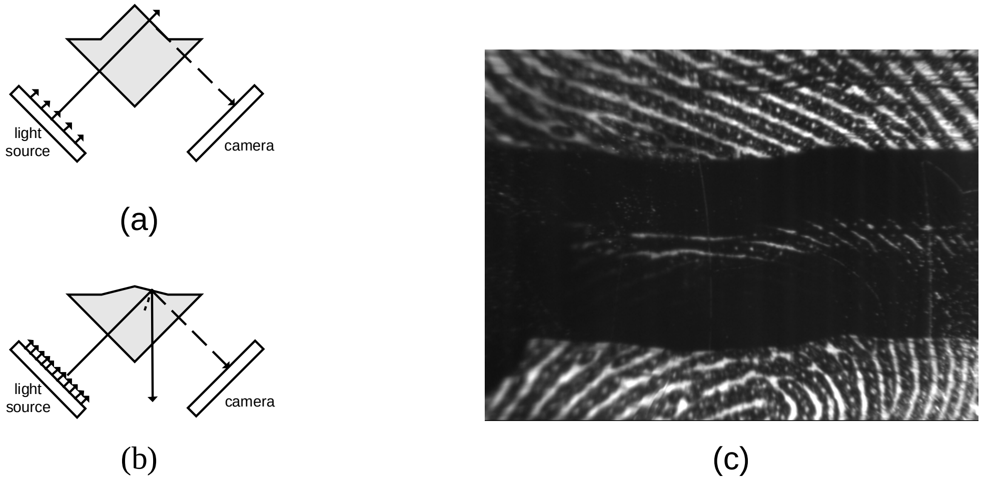


Figure 10. Failure of FTIR with a shape: (a) large slope, (b) small slope, and (c) fingerprint imaged through a ridge without a diffuser.

The size of the diffuser limits the angles at which light can strike the surface and thus imposes constraints on the surface gradients. Readers are referred to Appendix A for a detailed discussion. Assuming a right-angle prism with square faces of side *D* made of BK7 glass (see Figure 11), it can be shown that only a null horizontal gradient can insure reflection of the light satisfactorily. The surfaces are thus restricted to y-varying parallel ruled surfaces. It can then be shown that the vertical component of the surface gradient, denoted ∇ for simplicity, must fall in the range given by Expression (3.3) in order to insure that light is reflected toward the camera. Figure 12 illustrates the upper and lower bounds on the vertical gradient as a function of position.

$$\nabla_{\min}(x,y) = -\tan\left(\frac{1}{2}\arctan\frac{D+2y}{D+\sqrt{2}x}\right) \tag{3.1}$$

$$\nabla_{\max}(x,y) = +\tan\left(\frac{1}{2}\arctan\frac{D-2y}{D+\sqrt{2}x}\right) \tag{3.2}$$

$$\nabla_{\min}(x,y) < \nabla(y) < \nabla_{\max}(x,y) \tag{3.3}$$

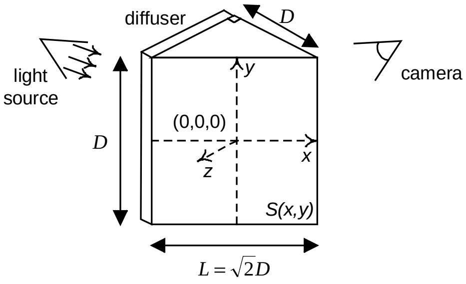


Figure 11. Notation for shape constraints.

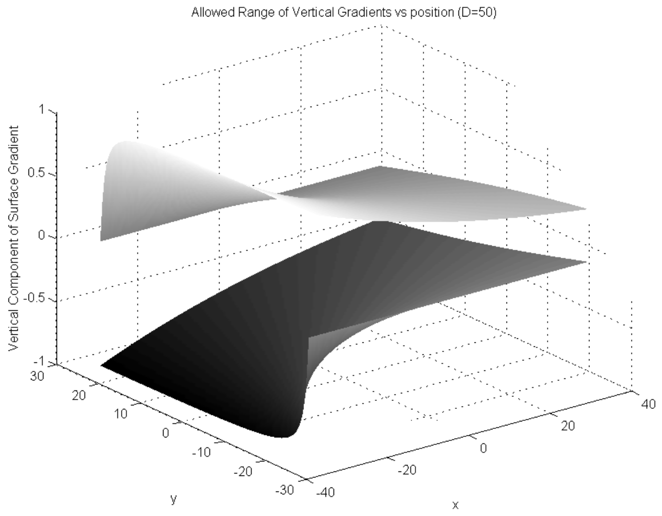


Figure 12. Allowed range of vertical gradients versus position.

The best flexibility is obtained when the field of view of the camera is centred vertically but displaced towards the entry face (low values of *x*). Furthermore it can be shown that the surface gradient is then limited by the constraints prevailing at the rightmost limit of the camera's field of view. Assuming a square field of view of side l, the limits on the surface gradient are given by Expressions (3.4) and (3.5).

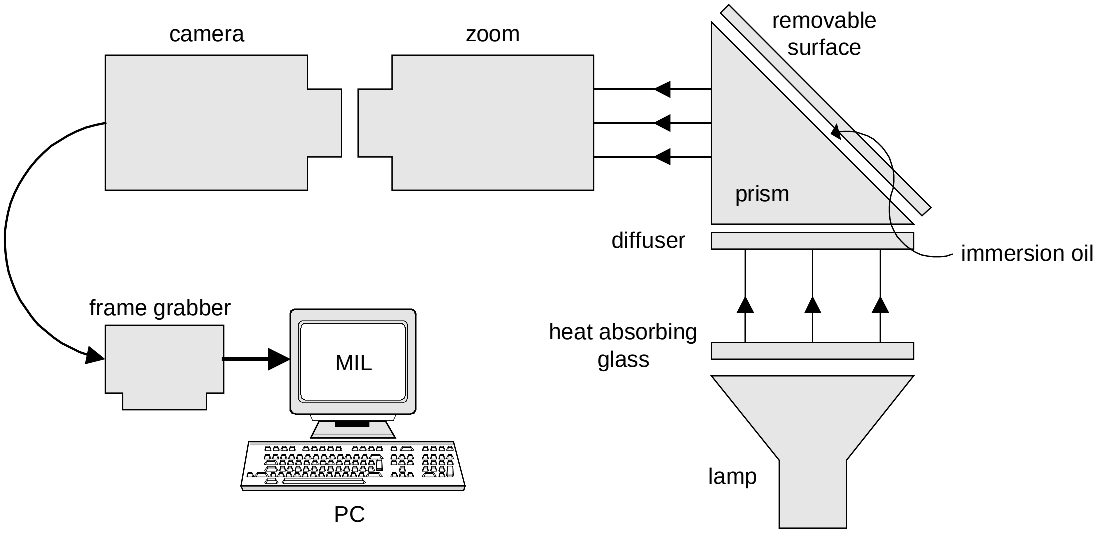
$$\nabla_{\min}(x,y)\big|_{x=l-L/2} = -\tan\left(\frac{1}{2}\arctan\frac{D+2y}{\sqrt{2}l}\right) \tag{3.4}$$

$$\nabla_{\max}(x,y)\big|_{x=l-L/2} = +\tan\left(\frac{1}{2}\arctan\frac{D-2y}{\sqrt{2}l}\right) \tag{3.5}$$

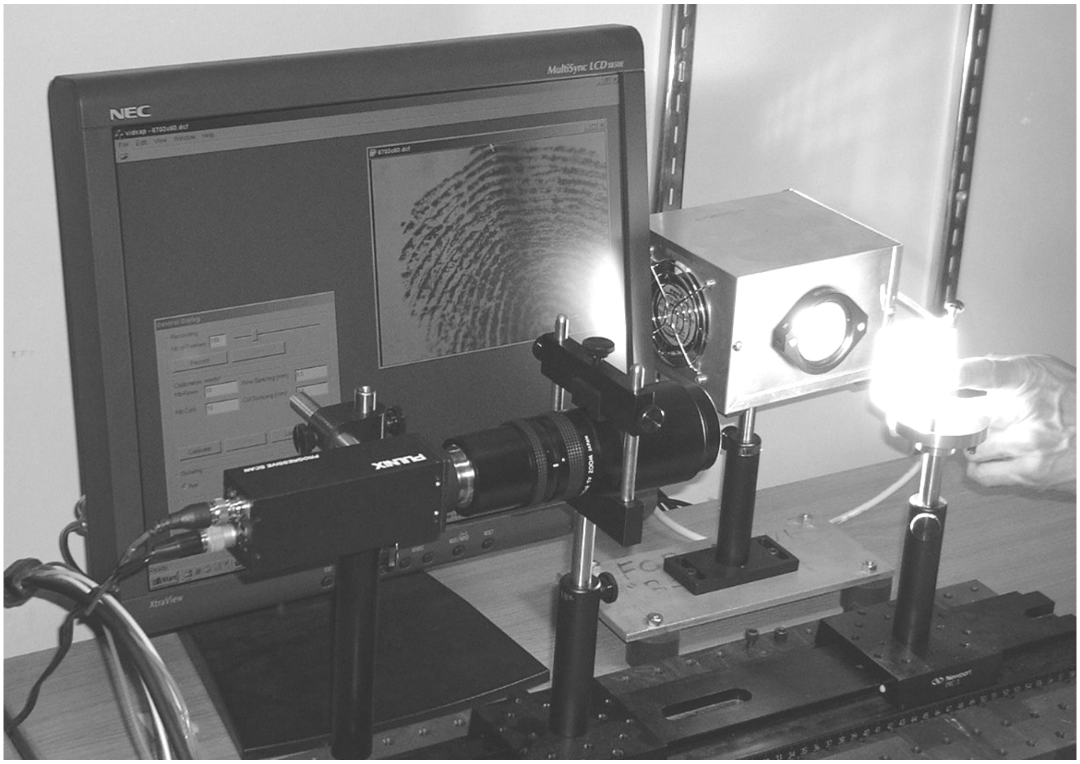
The optimal range occurs at y = 0 and the worst ranges at the upper and lower vertical boundaries. Assuming a field of view with $l = 20$ mm and a prism of size $D = 50$ mm, the limits on the vertical gradient are at worst –0.67 to 0.43 mm/mm (at $y = 10$ mm) and –0.43 to 0.67 mm/mm (at $y = -10$ mm), and at best -0.58 to 0.58 mm/mm (at $y = 0$ mm). This allows for a sufficiently large range of experiments.

3.3 Experimental Platform

An experimental platform was assembled using the concepts explained in the previous section. The main components of the platform are shown in Figure 13.



(a)



(b)

Figure 13. Experimental platform: (a) illustration, and (b) picture.

The right-angle prism measures 50 mm x 50 mm x 70 mm and is made of BK7 glass. Experiments showed that ground glass diffusers do not scatter light sufficiently well to approximate a Lambertian light source. An opal diffuser was selected for its high diffusion capabilities. Two parallel ruled surfaces were machined onto the surface of thin BK7 glass plates: a surface with a ridge (Figure 14(a)) and a surface with a valley (Figure 14(b)). The ridge and valley have a Gaussian profile with a height of 0.5 mm and a width of 3 mm (Figure 14(c)). The surfaces fit the specifications described in the previous section. Plates are joined with the hypotenuse face of the prism using an index matching liquid that minimizes reflections at the interface between plates and the prism (Immersion Oil Type A, Cargille Laboratories inc., NJ). The hypotenuse face of the prism is also used directly as a flat contact surface.

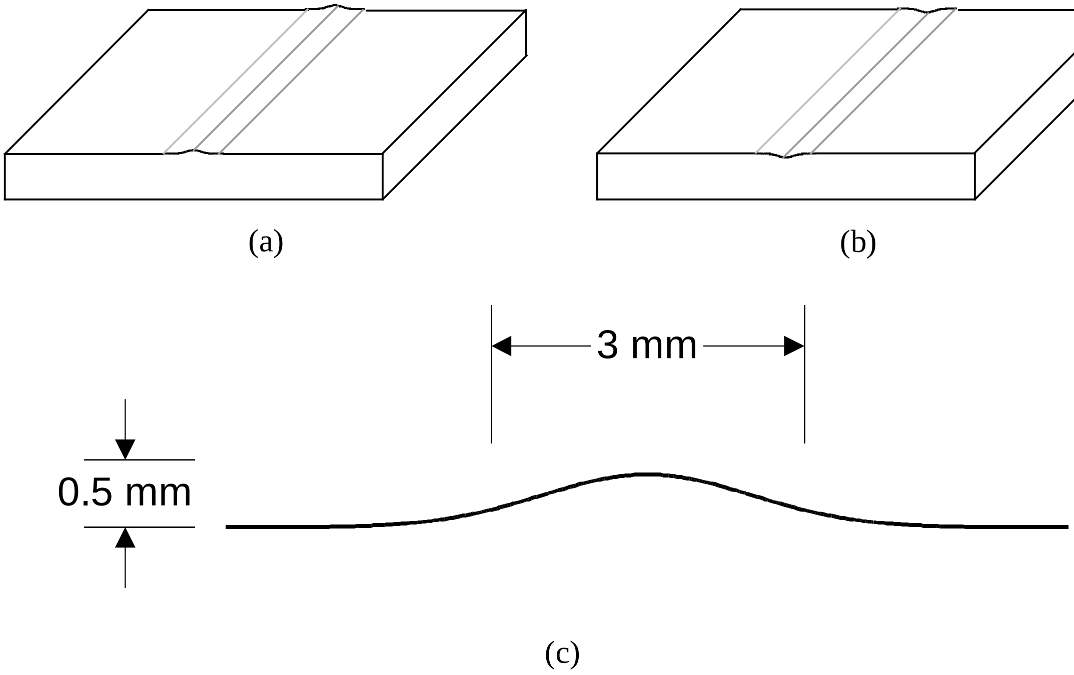


Figure 14. Parallel ruled contact surfaces: (a) ridge (actual size), (b) valley (actual size), and (c) Gaussian shape of ridge and valley.

Insufficient depth of field results in out of focus regions on the sides of the fingerprint due to the perspective view. The depth of field can be increased by reducing the aperture of the zoom, which in turn limits the amount of light that reaches the imaging sensor of the camera. Since the diffuser significantly attenuates the illumination, a powerful light source is necessary to maintain an acceptable depth of field. A light source was assembled using a 250W/120V halogen lamp. Heat absorbing glass was placed in front of the lamp to avoid heating up the prism.

A monochrome progressive scan CCD camera (TM-6703, Pulnix America, Inc., Sunnyvale, CA) was selected for its high resolution and high frame rate: 8-bit pixel depth with a resolution of 640 x 484 at 60 Hz, 640 x 242 at 120 Hz, and 640 x 100 at 220 Hz. The camera is oriented sideways to compensate for the horizontal compression of the fingertip[1]. A zoom (Zoom 7000, Navitar inc., Rochester, NY) allows the imaging system to focus on a region of approximately 10 mm x 10 mm at a working distance of 15 cm. The field of view can be reduced to approximately 5 mm x 5 mm with extension tubes. Images are acquired in real-time by a frame grabber (Meteor-II/Multi-Channel, Matrox Electronic Systems Ltd., Dorval, Canada) connected to a personal computer. Images are acquired and processed with software built using the Matrox Imaging Library (MIL) version 6.1, the Microsoft Foundation Class (MFC) library (Microsoft Corp., Redmond, WA), and the Computational Geometry Algorithms Library (CGAL, http://www.cgal.org).

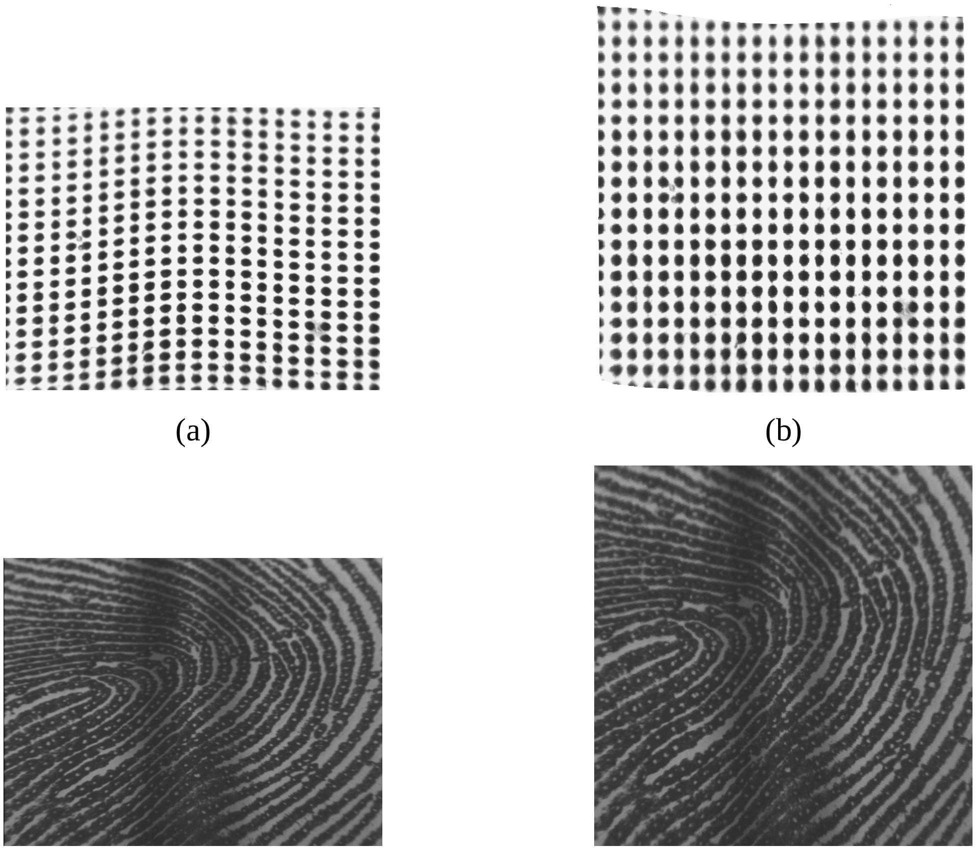
3.4 Geometric Calibration

The perspective view of the fingerprint and the surface shape cause geometric distortions in fingerprint images. These are corrected by imaging a known geometric pattern. A precise calibration grid consisting of dots spaced by 0.5 mm is printed on a sheet of transparency film. When applied to the contact surface with a thin film of immersion oil, the dots contact the surface and prevent the reflection of the light. The resulting pattern is analysed to correct the perspective projection and 'unroll' the contact surface. It thus takes into account the height of the shape and insures that measurements relate to the surface itself and not to its projection on a flat plane. The geometric calibration also determines pixel dimensions in distance units. This measurement is used to parameterize

---

[1] Images presented in this document are oriented as captured by the camera (i.e. rotated by 90º with respect to the prism).

image processing algorithms and provide skin stretch measurements and tactile display actuator positions in real-word units. Figure 15 shows an example of geometric calibration.



(a) (b)

(c) (d)

Figure 15. Geometric calibration over a ridge: calibration grid (a) before and (b) after calibration, fingerprint image (c) before and (d) after calibration.

3.5 Illumination Compensation

A diffuser improves the illumination uniformity of the halogen lamp but does not create a perfectly Lambertian light source. As a result, the energy reflected toward the camera varies with the position. This is especially true with high surface gradients, as light reflected in the camera originates from the outer edges of the diffuser where lighting may be of lower intensity. This problem is corrected by normalizing the intensity of fingerprint valleys in captured images. An illumination map *IM(x,y)* is computed by evaluating the maximum intensity for each pixel of an image *I(x,y)* over a sequence of *n* images.

$$(3.6) \quad IM(x,y) = \max_{0<i<n-1} I_i(x,y)$$

The illumination map is then low-pass filtered to eliminate small variations. A scaling factor is computed for each location by dividing the maximum greyscale value (255 in the case of 8-bit images) by the illumination map intensity. The original image is then scaled pixel-wise and clipped to the maximum greyscale value if necessary.

$$(3.7) \quad I'(x,y) = \min\left(\frac{255 \cdot I(x,y)}{LPF(IM(x,y))}, 255\right)$$

Figure 16 shows the illumination compensation of a fingerprint image acquired through a surface with a ridge. Notice that the sides of the ridge are initially darker than the rest of the image.

(a) (b) (c)

Figure 16. Illumination compensation: (a) original image, (b) illumination map, and (c) compensated image.

CHAPTER 4 FEATURE EXTRACTION

This chapter describes the extraction of features from fingerprint images. The feature extraction process is largely based on the work of Roddy and Stosz [30]. The process is illustrated in Figure 17. It begins with a smoothing operation, which is followed by a background detection operation that locates the fingertip. This is followed by a binarization operation that reduces the fingerprint to a black and white pattern. Small blobs are then detected and marked as pores. Thinning reduces the remaining pattern to a skeleton from which valley endings, valley bifurcations, and pores are extracted. Feature editing applies heuristic operations that correct common feature extraction errors. Each operation will be discussed in more details in this chapter.

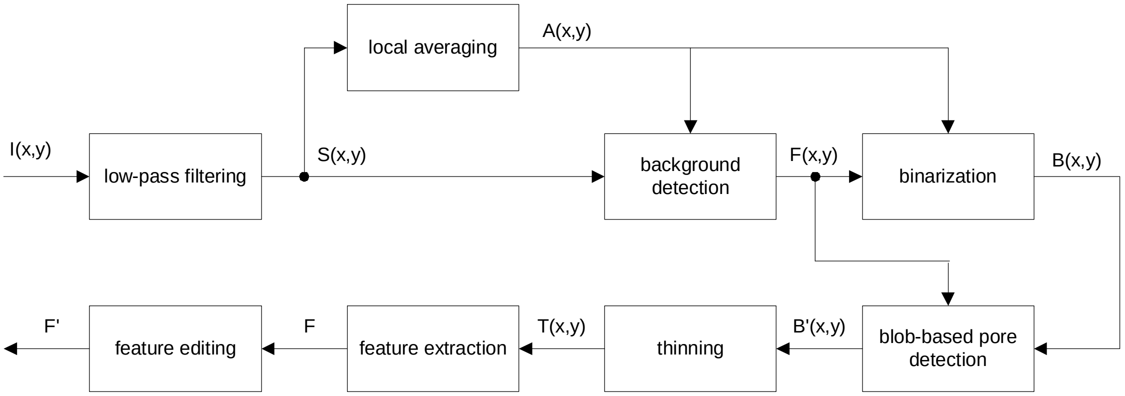


Figure 17. Feature extraction block diagram.

4.1 Low-Pass Filtering

Low-pass filtering eliminates low-frequency noise. The process consists of convolving an image *I(x,y)* with a Gaussian smoothing kernel *K(x,y)* as in equation (4.1). A standard deviation of 1 and a kernel diameter of 3 are generally sufficient for good results.

$$S(x,y) = \frac{I(x,y) \otimes K(x,y)}{\sum_{x,y} K(x,y)} \tag{4.1}$$

4.2 Local Average

The mean greyscale value in a sub-window of an image is useful for many purposes, including fingertip detection and adaptive binarization. In order to be meaningful, the averaging window must be large enough to include at least one ridge/valley cycle. This results in a window of a width in the order of a hundred pixels for a typical fingerprint. To reduce the amount of computation, the local average is often computed only for a number of windows [30]. By reusing computational results, it is possible to go further and compute the local average for every pixel in an image without incurring a significant time penalty, even for large averaging windows. See Appendix B for details on the algorithm used here. Figure 18 shows an example of local average computation. A averaging width of 2 mm is generally sufficient for good results.

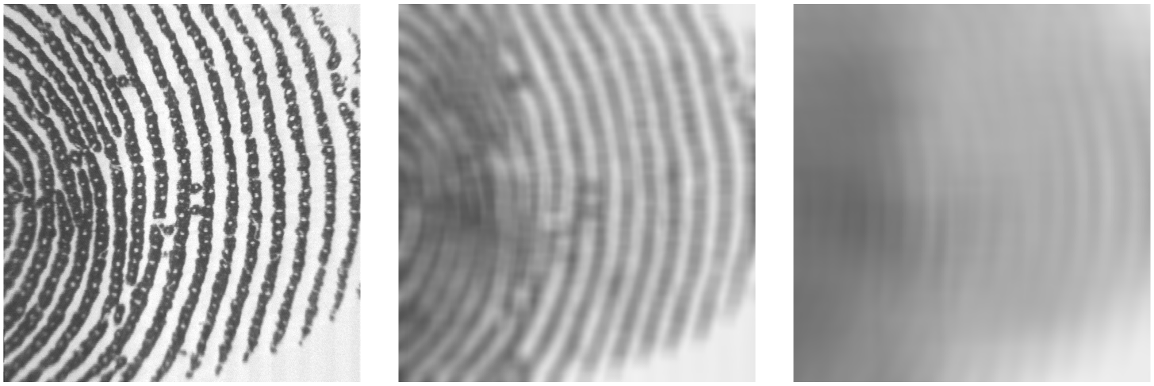

(a) (b) (c)

Figure 18. Local average: (a) fingerprint image, (b) local average map (width of 0.5 mm), and (c) local average map (width of 2 mm).

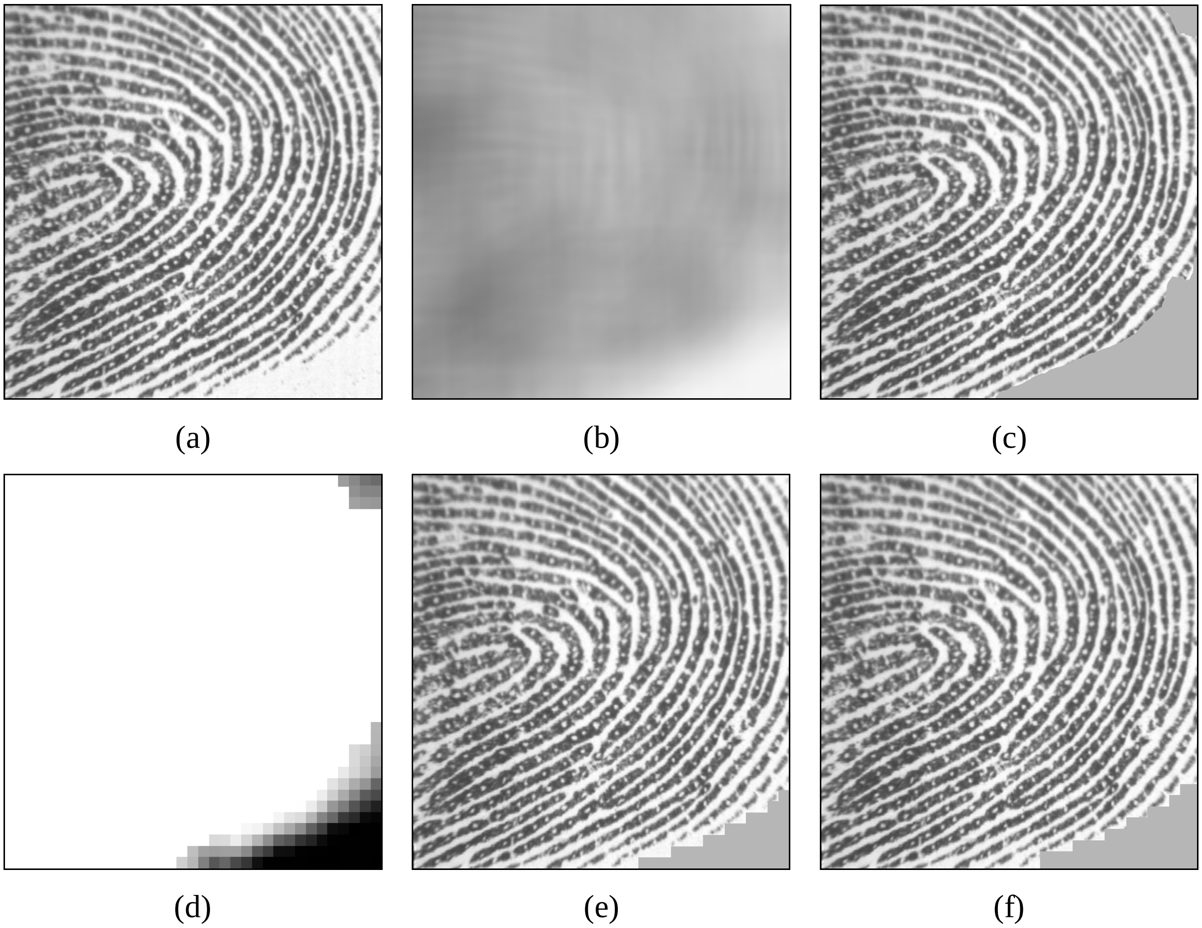
4.3 Background Detection

The fingertip movement tends to partially expose the contact surface, which results in a bright background with faint residual fingerprint traces. The background must be distinguished from fingerprint valleys to avoid the detection of false patterns in the binarization and thinning processes. The local mean can be computed efficiently using the algorithm described in the previous section and used to detect the background. The background is almost uniformly white and has a high local mean. The fingerprint, on the other hand, has both dark and bright regions resulting in a lower local mean. Positions with a local mean above a certain threshold are be assumed to belong to the background. The local mean is usually sufficient to detect the background accurately.

Accuracy can be improved by using the local variance [23]. Fingerprints have a high variance due to the ridge-valley cycles. The background, on the other hand, has a low variance due to its regularity. Positions with a local variance below a certain threshold are assumed to belong to the background. The local variance in a square window of width n is computed with Equation (4.2) where $\mu(x,y)$ is the local mean in a window of the same size [37].

$$\sigma^2(x,y) = \frac{\sum_{j=y-n/2}^{y+n/2} \sum_{i=x-n/2}^{x+n/2} \left(S(i,j) - \mu(x,y)\right)^2}{n^2} \qquad (4.2)$$

Unlike the local mean, the local variance is too expensive to be computed at every position. Computations are minimized by dividing the image in a grid and computing the variance in each square. The process is further optimized by neglecting the variance in squares where the local mean is below a threshold. The number of rows and columns is determined dynamically based on the number of background pixels detected using the local mean. A grid leading to approximately a hundred variance computations is generally reasonable and sufficiently accurate. The final background corresponds to positions for which both the local mean and the local variance agree. A few morphological dilation operations close holes and bring the background closer to the fingertip. The background is removed by setting all its pixels to black. Figure 19 illustrates the background detection process starting from the original image and ending in a background mask. Notice that the local average incorrectly marks the upper-right corner as being part of the background (see Figure 19(c)). The variance analysis eliminates the error (Figure 19(e)). Note also that variances that are not computed are set to the maximum in Figure 19(d).


(a) (b) (c)

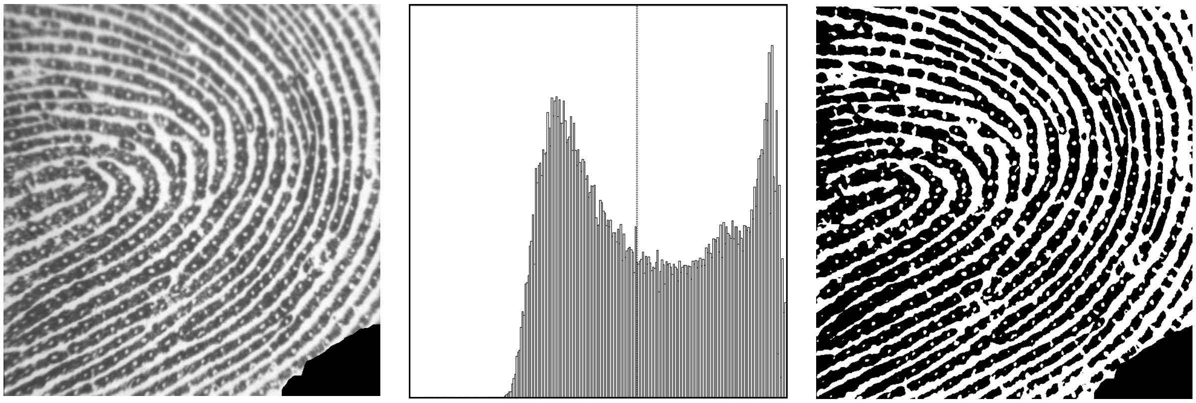
(d) (e) (f)

Figure 19. Background detection: (a) input image, (b) local average, (c) background detected using local average, (d) variance, (e) background detected using variance, and (f) background.

4.4 Binarization

The binarization consists of converting an 8-bit grayscale image to a 1-bit binary image. Superfluous information is eliminated by determining which pixels belong to a ridge and which pixels belong to a valley or pore. The difficulty is in finding a thresholding algorithm capable of making this decision reliably. This section discusses numerous methods that were tried.

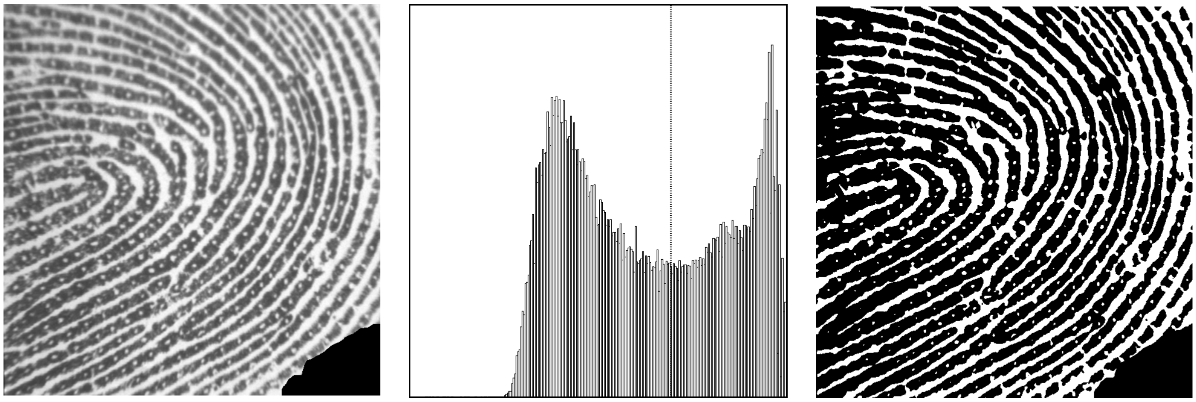
The simplest approach consists of selecting manually a single threshold to be applied everywhere (Figure 20). This method requires a significant amount of tweaking since optimal thresholds differ from one sequence to the next. Variations in fingerpad pressure in a single frame or in a sequence of frames often make it impossible to find a single, reliable threshold.



(a) (b) (c)

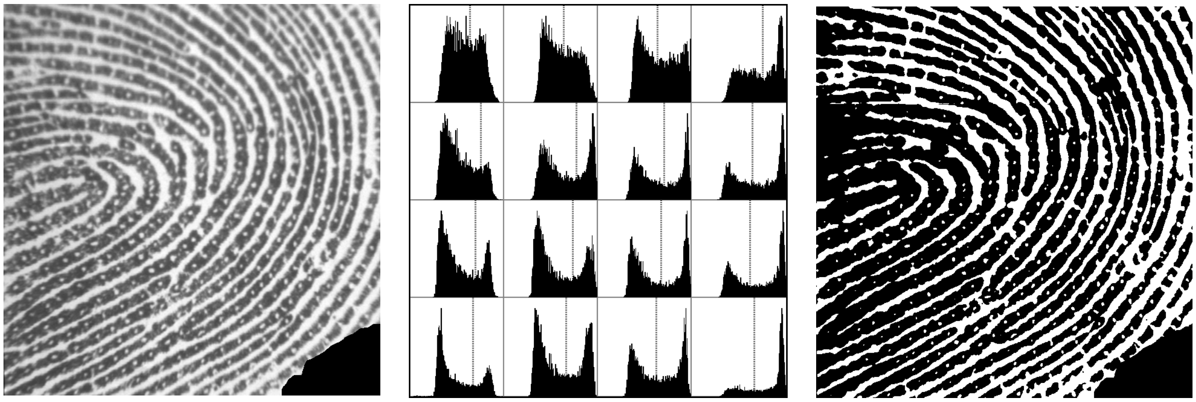
Figure 20. Binarization with user-selected threshold: (a) input image, (b) histogram with threshold, and (c) binarized image.

Another option is to use MIL’s proprietary threshold selection procedures (Figure 21). Although the details are not public, MIL probably analyses the histogram of the image and selects a threshold that falls between high peaks. An improvement consists of analysing a number of windows separately in order to have local thresholds (Figure 22).



(a) (b) (c)

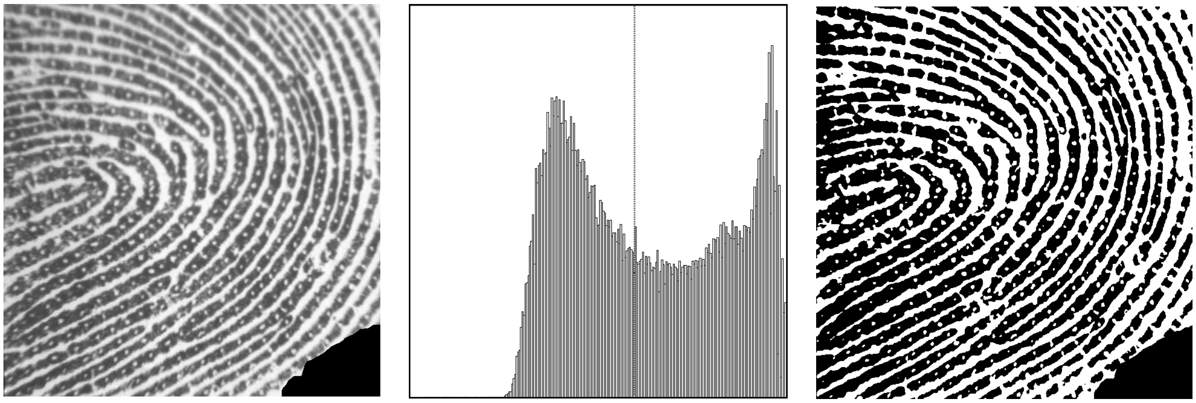
Figure 21. Binarization with MIL’s threshold selection method: (a) input image, (b) histogram with threshold, and (b) binary image.



(a) (b) (c)

Figure 22. Binarization with MIL’s threshold selection method (4x4): (a) input image, (b) histograms with thresholds, and (c) binary image.

Another solution consists of selecting the mean intensity as the binarization threshold (Figure 23). Notice that parts of the image are not binarized correctly. This is likely due to non-uniform fingerprint pressure resulting in variations in valley intensity. To take these differences into account, the local average map computed in section 4.2 can be used as an adaptive pixel-wise threshold (Figure 24). This is similar to the adaptive thresholding method used in [30]. Scaling the local average map by a user-defined factor occasionally improves the binarization. The local average method doesn’t yield a perfect fingerprint but is generally the most reliable method.



(a) (b) (c)

Figure 23. Binarization using mean greyscale level: (a) input image, (b) histogram with threshold, and (c) binary image.


(a) (b) (c)

Figure 24. Binarization using local average: (a) input image, (b) local average map, and (c) binary image.

4.5 Pore Detection

Most of the pores can be detected directly from the binary fingerprint image. A pore is a round feature with a diameter varying between 88 and 220 µm [29]. Taking into account the imperfections in image acquisition and binarization, it is assumed that any connected component (or blob) of the binary fingerprint is a pore if its diameter is less than 0.25 mm. A blob is defined as a set of white pixels in which every pixel is 4-connected to at least one other pixel in the set. The position of a pore P is determined by computing its centre of mass using greyscale intensity values from the foreground image as in equation (4.3).

$$(4.3) \quad (x, y)_P = \left(\frac{\sum_{i,j \in P} iF(i,j)}{\sum_{i,j \in P} F(i,j)}, \frac{\sum_{i,j \in P} jF(i,j)}{\sum_{i,j \in P} F(i,j)} \right)$$

This pore detection step is optional but is recommended in order to improve the accuracy of the pore's position. If not detected here, pores are detected during the skeleton analysis phase. In order to combine the result of this pore detection with the next operations seamlessly, pores are reduced to a single pixel at their centre of mass. Figure 25 illustrates the pore detection process.

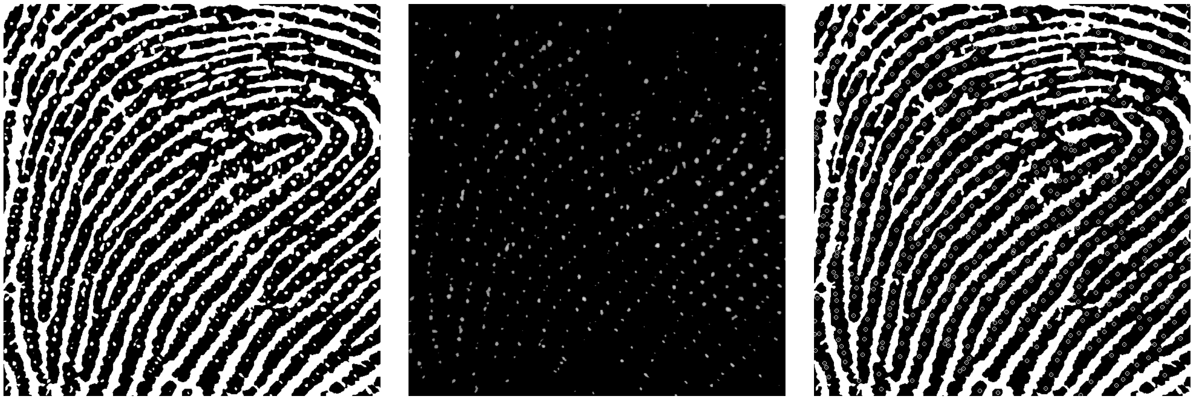

(a) (b) (c)

Figure 25. Pore detection: (a) binary image, (b) selected blobs in greyscale, and (c) binary image with pores shown as circles.

4.6 Thinning

The width of a fingerprint pattern carries no useful information for our immediate purpose and is eliminated by a thinning operation. Thinning algorithms reduce connected patterns to a width of a single pixel while maintaining their topology. A good thinning algorithm is also resistant to contour noise and produces a skeleton that falls approximately on the medial axis of the pattern. While numerous algorithms exist, many trade reliability for speed and result in the deletion or disconnection of some patterns. The selected algorithm is a modified version of the Zhang-Suen thinning algorithm [28]. Post-processing is applied to reduce the skeleton to a near minimal 8-connected pattern.

Zhang-Suen thinning repeatedly deletes contour points respecting a number of conditions until a 1-pixel wide 8-connected skeleton is obtained. The algorithm alternates between two passes, each selecting points for deletion based on different criteria. Consider a point P and its 8-neighborood as shown in Figure 26(a). Figure 26(b) shows the four conditions marking points for deletion in the first and second passes. Condition a) depends on the number of non-zero neighbours $N(P)$. Condition b) depends on the number of zero-to-one transitions $T(P)$ in a round-trip around the 8-neighborood. The modifications proposed by Ritter and Wilson avoid the erosion of square blocks of four pixels [28]. The Zhang-Suen algorithm can be implemented efficiently using look-up tables.

p_8	p_1	P_2
p_7	P	p_3
p_6	p_5	p_4

(a)

First Pass	Second Pass
a) $2 \le N(P) \le 6$	
b) $T(P) = 1$	
c) $p_1 \cdot p_3 \cdot p_5 = 0$	c') $p_1 \cdot p_3 \cdot p_7 = 0$
d) $p_3 \cdot p_5 = 0$	d') $p_1 \cdot p_7 = 0$

(b)

Figure 26. Zhang-Suen thinning (modified version): (a) 8-Neighborood of pixel P, and (b) thinning rules.

The 8-connected skeleton resulting from the modified Zhang-Suen thinning algorithm is not minimal. Numerous pixels can be deleted without breaking the connectivity of the patterns and without moving end points of lines. Most notable are the wide staircases resulting from diagonal lines (Figure 27), which cause the minutiae detection algorithm to fail. A simple remedy consists of performing a pixel deletions pass with each of the deletion kernels shown in Figure 27(a). The center pixel of these kernels can be deleted without affecting the connectivity of the skeleton since all "don't care" pixels have another white pixel as a neighbour. Figure 28 illustrates the thinning of a binary fingerprint image.

(a) (b) (c)

Figure 27. Simplification of 8-connected patterns: (a) four deletion kernels, (b) pattern before simplification, and (c) pattern after simplification.

(a) (b)

Figure 28. Thinning example: (a) binary image, and (b) thinned image.

4.7 Feature Extraction

Fingerprint matching typically relies on features of the ridge patterns called minutiae. Minutiae can be divided in two classes: ridge endings and ridge bifurcations. Occasionally a third type of feature corresponding to sweat pores is also used [30][33]. Remembering that our fingerprint skeleton corresponds to valleys rather than ridges, this section describes the extraction of valley endings, valley bifurcations and pores.

The number of 8-neighbors of skeleton points determines their classification. Pores have no neighbours, valley endings have a single neighbour and valley bifurcations have three or more neighbours. All other skeleton points belong to a valley and are not considered to be features. Pixel classification is illustrated in Figure 29. The notation: circle, square, triangle, is used through the end of this thesis. A valley bifurcation may result in a group of connected valley bifurcation pixels (Figure 29), in which case its position is the centre of mass of its pixels.

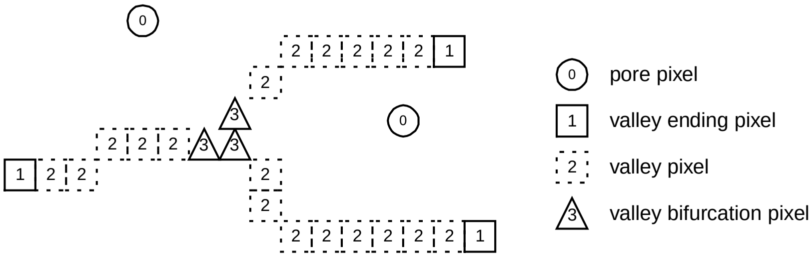
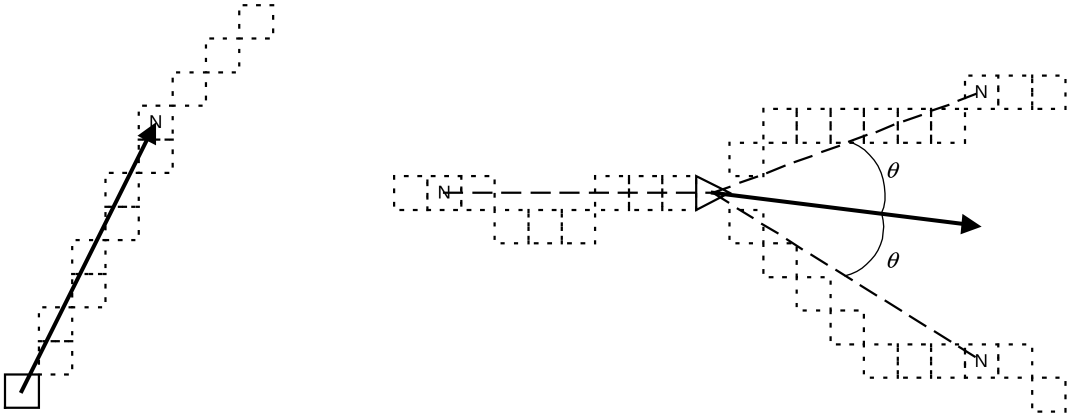


Figure 29. Skeleton pixel classification using number of neighbours.

The orientation of nearby valleys is used to obtain distinguishing minutia characteristics. The orientation of a valley is determined by tracing a vector from a minutia to the *N*’th pixel of the valley. The orientation of a valley ending is given by the orientation of its only valley (see Figure 30(a)). The orientation of a valley bifurcation is selected so as to bisect the smallest angle between nearby valleys (see Figure 30(b)). This definition differs slightly from the typical orientation definition found in the literature [15][30]. In some situations the available information may be insufficient or not sufficiently coherent to determine the orientation of a minutia. For example, a ridge bifurcation may join more than three ridges and very short valleys may not allow meaningful orientation computations. Pores have no reliable distinguishing features. Figure 31 shows an example of feature extraction in a small portion of a fingerprint.



(a) (b)

Figure 30. Minutiae orientation: (a) valley ending orientation, and (b) valley bifurcation orientation.

(a) (b) (c)

Figure 31. Feature extraction: (a) greyscale image, (b) skeleton (after blob-based pore detection), and (c) extracted features with minutiae orientation.

4.8 Feature Editing

The image acquisition process and subsequent pre-processing operations often result in fingerprint skeleton artefacts. These artefacts in turn result in spurious features, incorrect feature labels, or missing features. Four common artefacts are shown in Figure 32. Short valleys, short spurs and bridges should typically be replaced by pores. Broken valleys, on the other hand, should be healed by bridging their gap. This section describes syntactic editing rules applied to the fingerprint skeleton to eliminate most of these artefacts. Most skeleton processing rules are taken from [30] with a few modifications. This section also discusses two filtering operations that reject unreliable features. The first operation discards regions of high feature density. The second operation rejects feature near the outer border of the fingerprint.

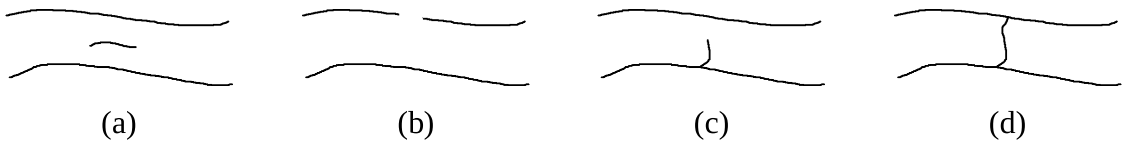


Figure 32. Artefacts: (a) short valley, (b) broken valley, (c) spur, and (d) bridge.

4.8.1 Short Valleys

This section proposes an alternative to the blob-based pore detection method described in section 4.5. When blob-based pore detection is disabled, pores are typically reduced to short lines. Short valleys bounded by two end points are thus replaced by pores. Considering the size of a pore, a maximum distance of approximately 0.2 mm is reasonable. Figure 33 and Figure 34 illustrate the removal of short valley and the decision process. Figure 35 shows an example of short valley removal.

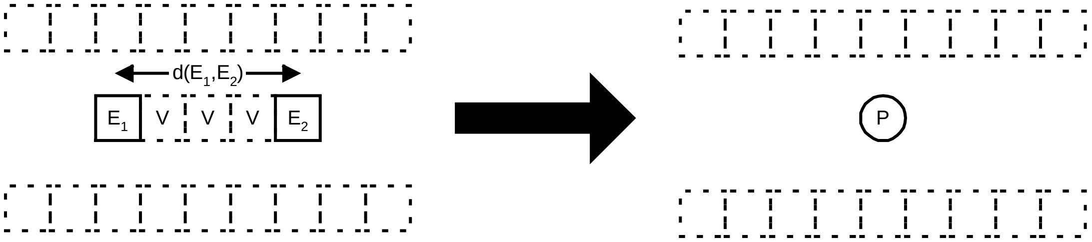


Figure 33. Short valley removal illustration.

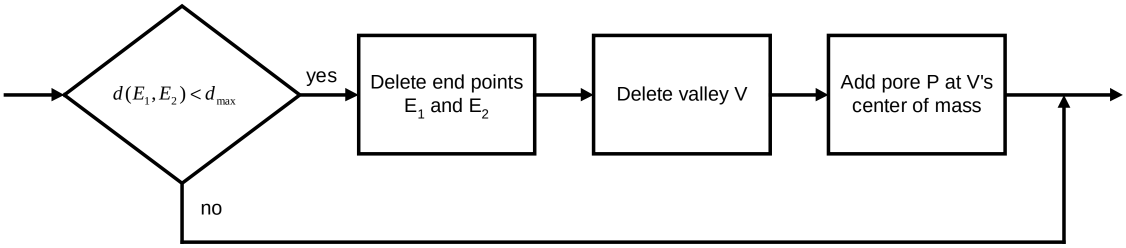
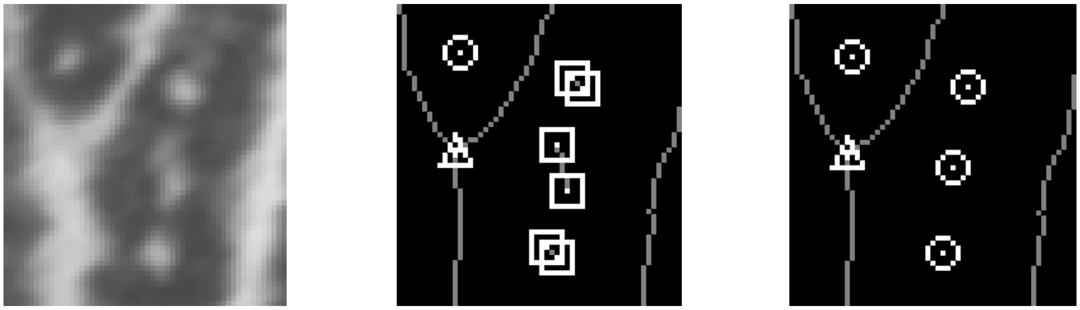


Figure 34. Short valley removal decision flowchart.



(a) (b) (c)

Figure 35. Replacement of short valleys by pores: (a) greyscale image, (b) features before corrections, and (c) features after corrections.

4.8.2 Broken Valleys

The image acquisition and subsequent pre-processing steps often result in broken valleys and thus in the apparition of spurious ridge endings. Intuitively, small gaps between valleys should be bridged unless there is a conflict with other valleys. The valley healing algorithm examines all pairs of valley endings and evaluates the suitability of bridging the gap between them. Consider end points E_1 and E_2 as shown in Figure 36.

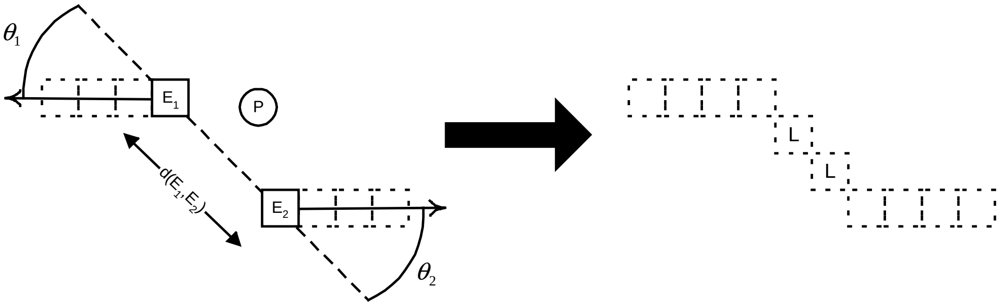


Figure 36. Valley healing illustration.

In order to be considered, the distance between the two end points must be lower than a certain threshold. The angles θ_1 and θ_2 subtended by the bridging line and the minutiae orientations must also be lower than a threshold. A maximum distance of 0.6 mm and a maximum angular deviation of 30° were found to be reasonable. Pairs satisfying these conditions are given a confidence rating according to equation (4.4) and added to a list of potential healing pairs. The candidate selection process is described in Figure 37.

$$(4.4) \quad C(E_1, E_2) = \left(\frac{d_{\max} - d(E_1, E_2)}{d_{\max}} \right) \bullet \left(\frac{\theta_{\max} - \theta_1}{\theta_{\max}} \right) \bullet \left(\frac{\theta_{\max} - \theta_2}{\theta_{\max}} \right)$$

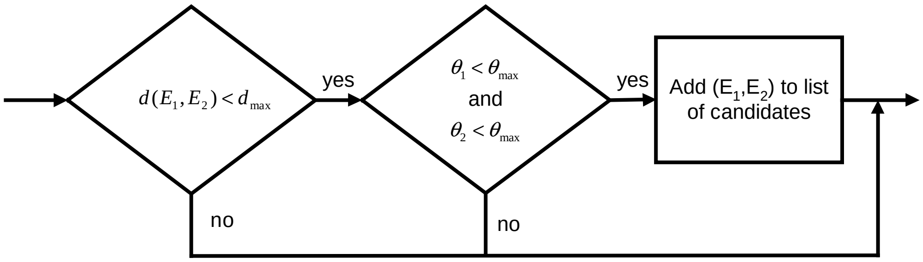


Figure 37. Selection of valley healing candidates flowchart.

The healing algorithm proceeds in order of decreasing confidence rating. Healing proceeds by deleting the pair of end points and drawing a straight valley between them. A end point may be healed only once. Candidate pairs are also rejected if the bridging line is in contact with points other than the deleted end points. This prevents the healing process from changing the connectivity of the valley pattern beyond the two joined valleys. Finally, pores that fall in the 8-neighborood of the bridging line are assumed to be part of the newly healed valley and are deleted. The valley healing process is summarized in Figure 38 and illustrated in Figure 39.

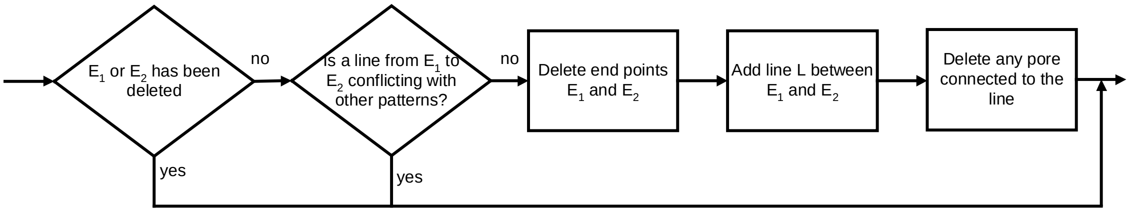


Figure 38. Valley healing flowchart.

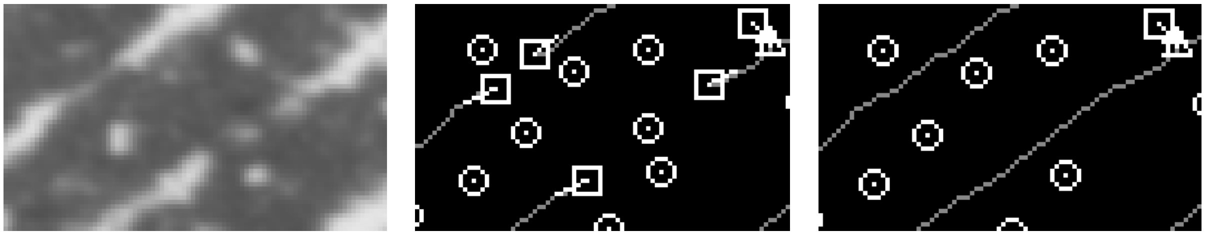

(a) (b) (c)

Figure 39. Valley healing: (a) greyscale image, (b) features before corrections, and (c) features after corrections.

4.8.3 Short Spurs

Image acquisition and pre-processing often causes short spurs or protrusions along the edges of valleys. Some of the spurs are due to the effect of thinning curved valley boundaries while others are due to the proximity of pores. This section describes processing rules that eliminate most of the spurs and restore as many pores as possible. Spurs consist of short valleys bounded by a branch point on one side and an end point on the other. Spurs are removed only if their length is less than a threshold and if they are approximately perpendicular to the valley. Consider the spur formed by *B* and *E* in Figure 40. Perpendicularity is evaluated by computing the angles θ_1 and θ_2 between the spur and the two sides of the valley. The spur is ignored if the angles differ from 90° by more than a certain threshold value. This prevents the deletion of valid valley endings that are close to a branch point.

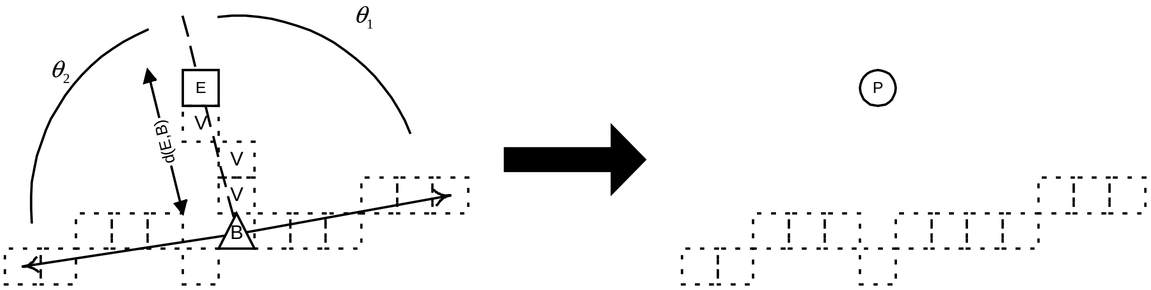


Figure 40. Short spur removal illustration.

Spurs respecting these conditions are deleted by removing the end point and the connecting valley. The valley bifurcation is re-evaluated, leading to possible deletion or re-classification of its constituent pixels. If the spur is longer than a certain threshold, a pore is added at the position of the end point. The spur removal process is summarized in Figure 41. Figure 42 shows an example of spur correction. Notice that the two upper spurs result in the addition of two pores while the two lower spurs do not. A maximum angular deviation of 60°, a minimum length of 0.1 mm and a maximum length of 1 mm were found to provide acceptable results.

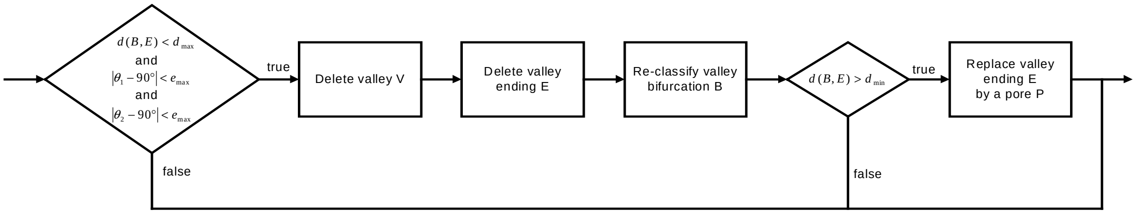
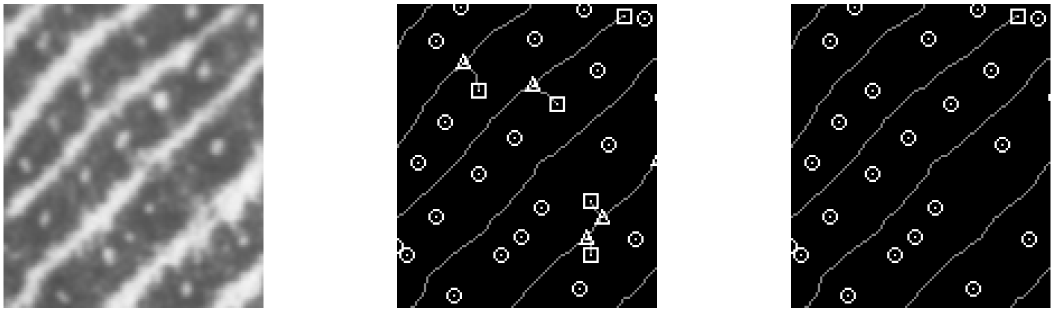


Figure 41. Short spur removal decision flowchart.



(a) (b) (c)

Figure 42. Short spur removal: (a) greyscale image, (b) features before corrections, and (c) features after corrections.

4.8.4 Bridges

Pores occasionally cause a bridge to form itself between two adjacent valleys. Syntactic rules can be applied to remove these bridges and replaced them by a pore. A bridge is a short valley segment bounded on both sides by branch points. A bridge is not deleted unless its length falls in a given range. It must also not be deleted if it isn't roughly perpendicular to the two nearby valleys. The perpendicularity is evaluated on both sides using the same technique as for spurs. A bridge is not deleted unless the four angles θ_1 to

θ_4 differ from 90° by less than a given threshold. If a bridge respects all these conditions, the valley segment is deleted, the two valley bifurcation are re-evaluated and a pore is added at the centre of mass of the bridge. The bridge removal process is illustrated in Figure 44. Figure 45 shows an example of bridge removal. A length between 0.1 mm and 1 mm, and a maximum angular deviation of 60° were found to be reasonable.

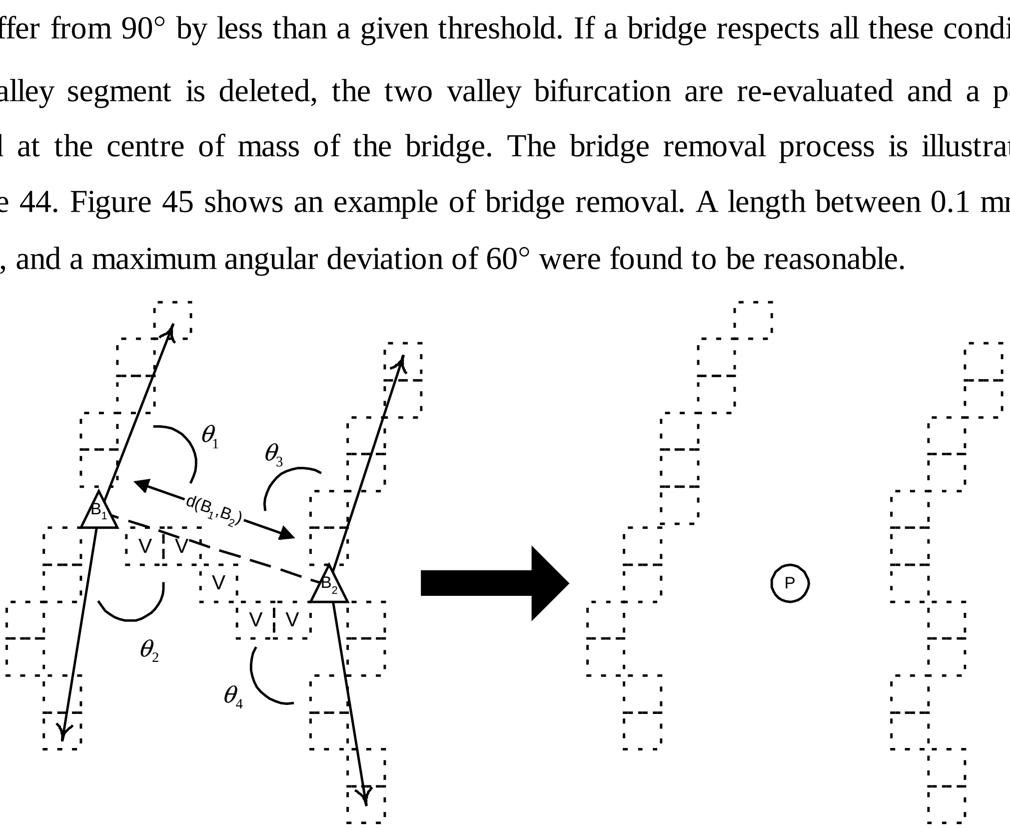


Figure 43. Bridge removal illustration.

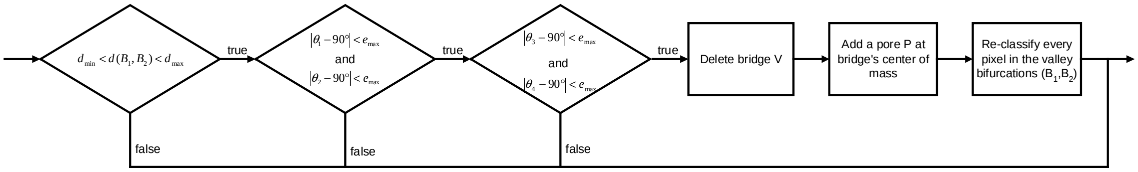
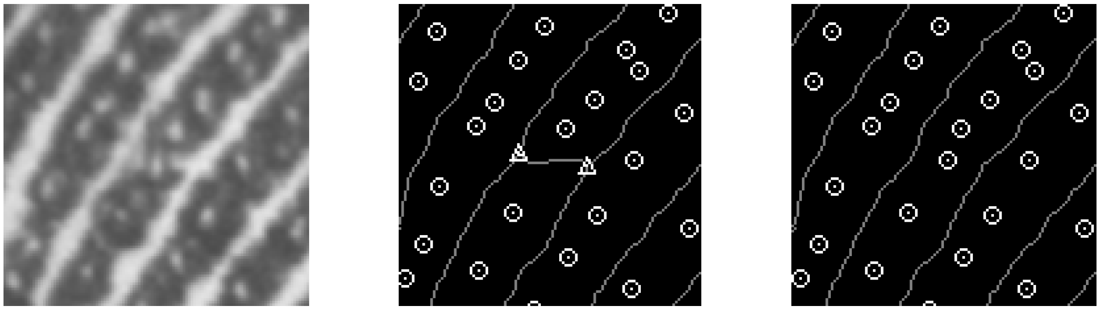


Figure 44. Bridge removal decision flowchart.



(a) (b) (c)

Figure 45. Bridge removal: (a) greyscale image, (b) features before corrections, and (c) features after corrections.

4.8.5 Feature Density

High feature density is generally the result of incorrect feature extraction. Feature density in the neighbourhood of a feature is thus a good measure of its reliability. It is evaluated by counting the number of features in a circular window of a given radius. All enclosed features are deleted if their number is greater than a threshold. This operation is performed in parallel for all features. Features density is evaluated separately for all three types of features. Figure 46 illustrates the operation with a radius of 0.25 mm and a maximum number of features of 3.

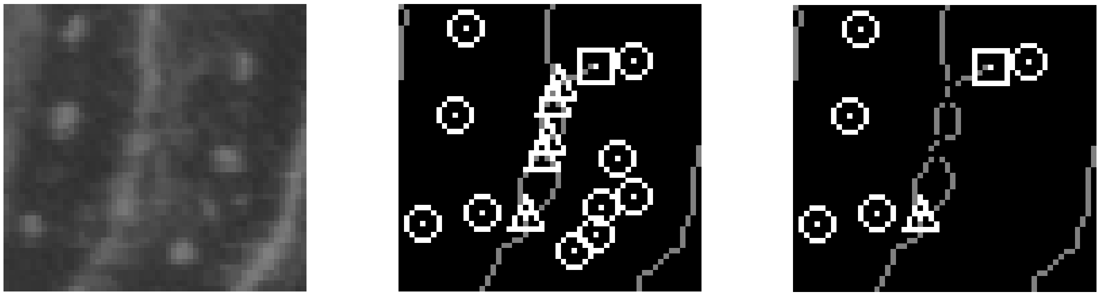
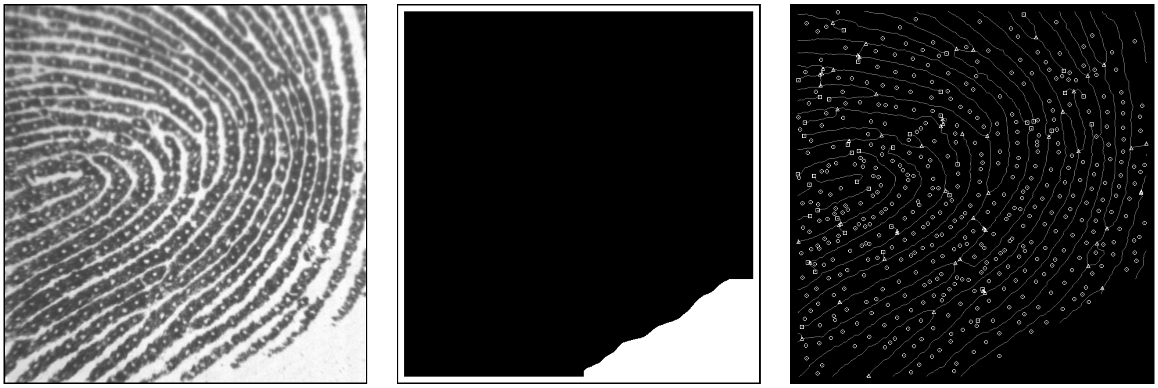

(a) (b) (c)

Figure 46. Feature density filtering: (a) greyscale image, (b) features before corrections, and (c) features after corrections.

4.8.6 Fingerprint Border

The edges of the image as well as the loss of contact between the fingertip and the surface cause false features, especially end points, to be detected. These false features appear at the edges of the image and at the interface between the fingerprint and the background. A deletion mask is grown from the background detected in section 4.3. A 1-pixel border is added to the mask to take into account the edge of the image. The mask is then expanded by a number of dilations. Any feature that falls beneath the mask is deleted. Figure 47 illustrates the process.



(a) (b) (c)

Figure 47. Deletion of bordering features: (a) grayscale image, (b) expanded deletion mask, and (c) features after corrections.

4.8.7 Summary

The feature editing routines described in this section generally have a significant impact on the quality of the extracted features. Figure 48 and Figure 49 show an example of feature extraction and feature editing through all the steps involved. Even though syntactic processing occasionally makes mistakes, the overall effect is generally beneficial. Note that blob-based pore detection was disabled here to illustrate the effect of short valley removal.

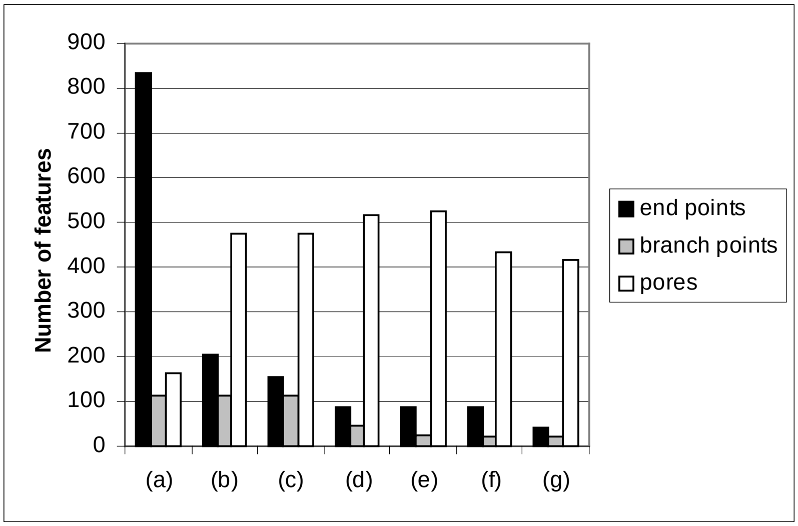
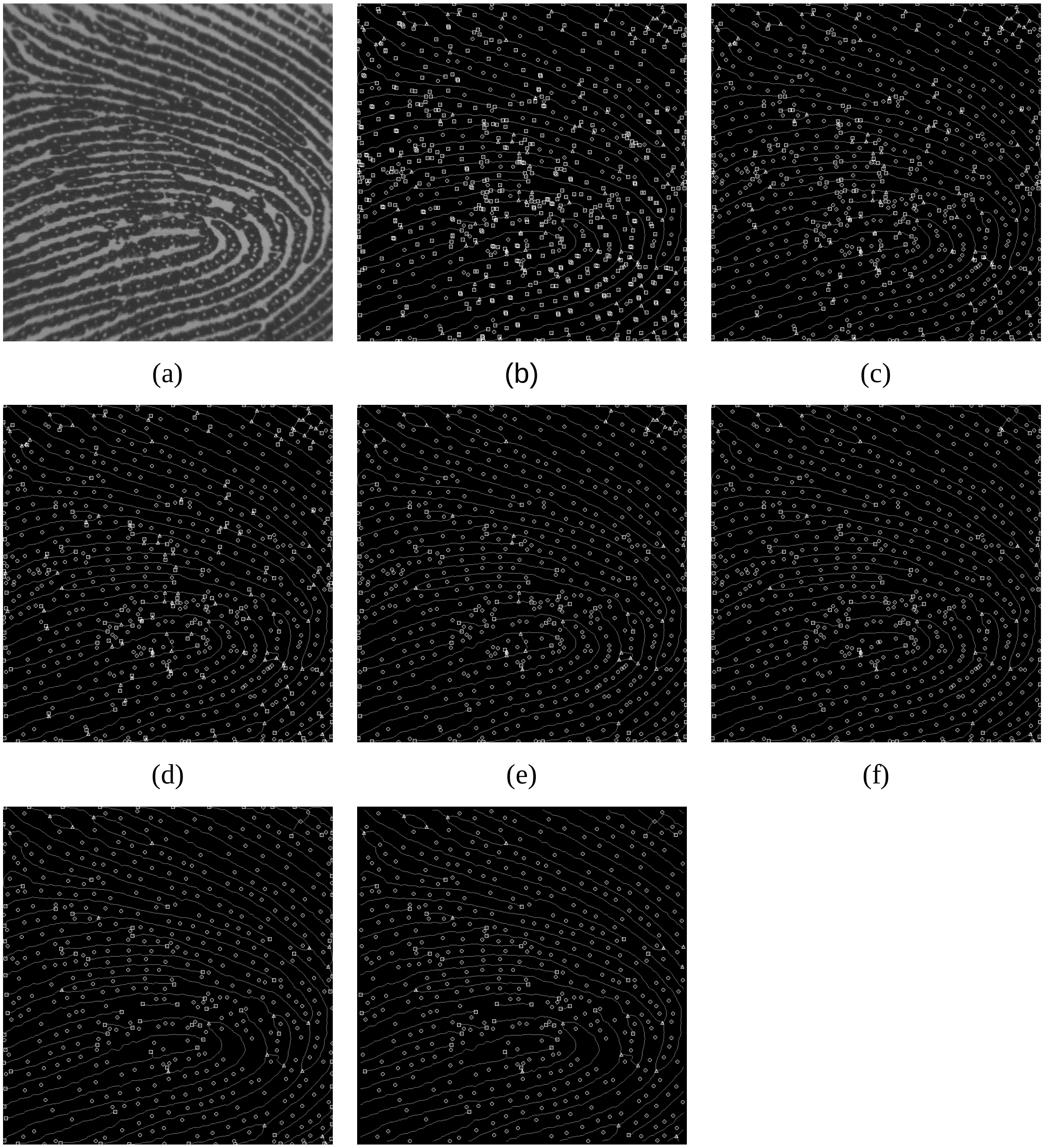


Figure 48. Effect of feature editing on feature count: (a) initial, (b) short valley removal, (c) valley healing, (d) short spur removal, (e) bridge removal, (f) feature density filtering, and (g) border removal.



(a) (b) (c)

(d) (e) (f)

(g) (h)

Figure 49. Feature editing: (a) greyscale image, (b) initial features, (c) short valley removal, (d) valley healing, (e) short spur removal, (f) bridge removal, (g) feature density filtering, and (h) border removal.

CHAPTER 5 SKIN STRETCH MEASUREMENT

This section describes the method used to derive skin stretch information from features extracted from a sequence of fingerprint images. Section 5.1 explains how features are tracked through consecutive frames. Section 5.2 describes the tracking of the fingertip. Section 5.3 explains how tracked features are used to study local skin stretch changes. Finally, section 5.4 describes the mapping from skin stretch measurements to tactile movies.

5.1 Feature Tracking

Feature tracking consists of two steps. The first step matches as many features as possible in pairs of consecutive frames. The second step assembles matches into feature trajectories.

5.1.1 Feature Matching

Feature matching attempts to associate features between pairs of successive frames. It is assumed that the image acquisition rate is sufficiently high to insure that feature displacements are much shorter than inter-feature distances. Matching is performed by searching for the best match near a feature's expected position as illustrated in Figure 50(a). The displacement of a feature is predicted from its previous displacement, if available. The average of all predicted displacements is used for features for which the previous position is unknown. A circular search window of a predefined radius is placed at the feature's expected position. Any feature of the same type (valley ending, valley bifurcation or pore) is considered a candidate match. In the example shown, features A and B are candidate matches while C and D are not.

Each candidate match is given a confidence rating that decreases with the distance from the feature's expected position and with the orientation error as illustrated in Figure 50(b). The confidence rating is given by Equation (5.1). The orientation component is null for pores, and minutiae with unknown orientation.

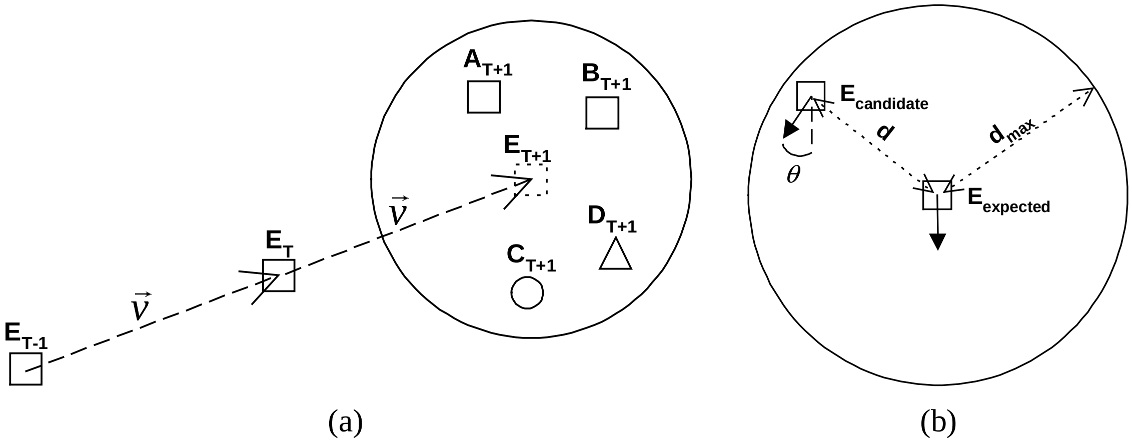
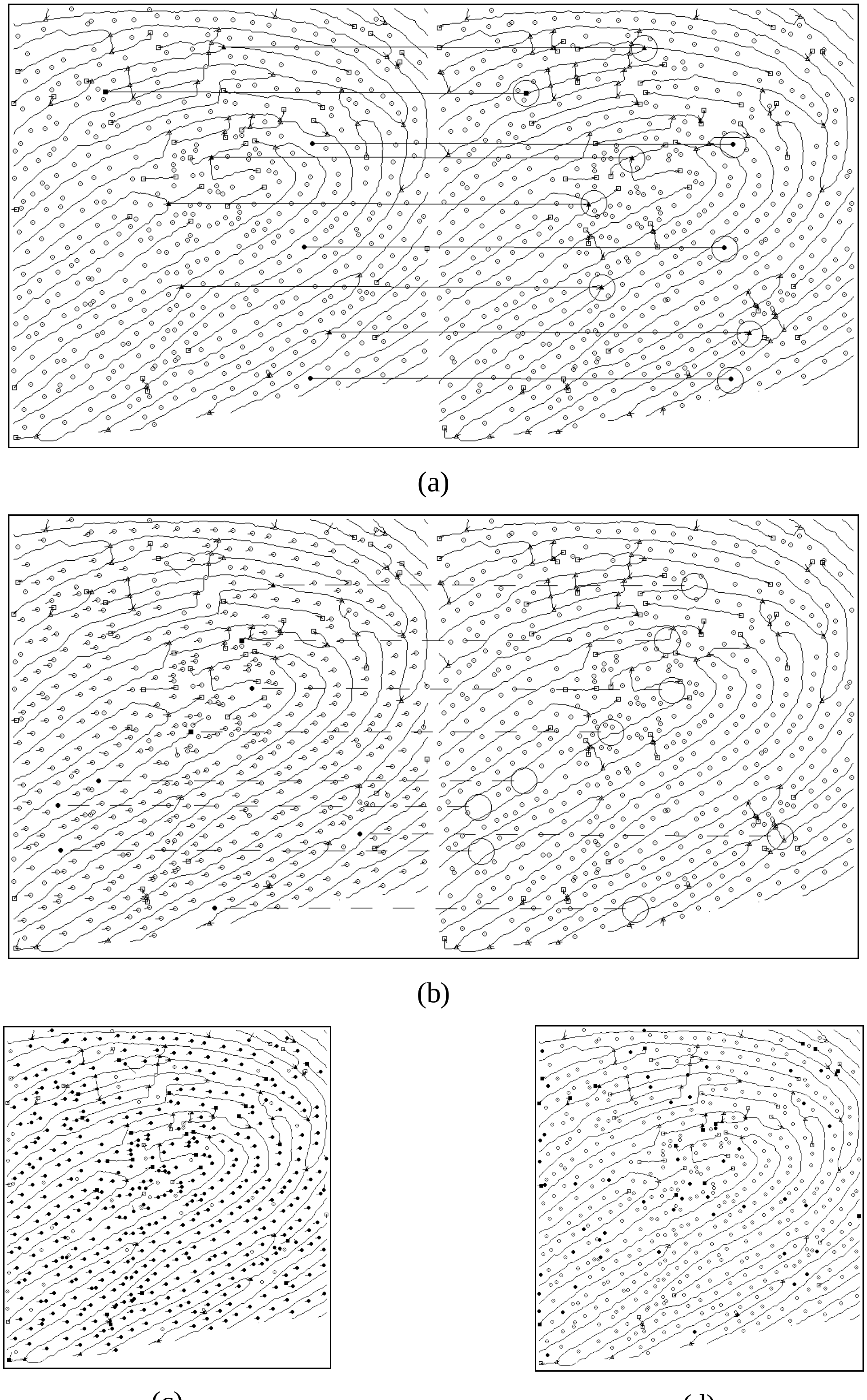


Figure 50. Feature matching illustration: (a) candidate match selection, and (b) candidate match rating.

$$C(F) = \left(\frac{180° - \theta}{180°}\right) + \left(\frac{d_{\max} - d}{d_{\max}}\right) \tag{5.1}$$

A list of candidate matches is generated for each tracked feature in a frame. The complete list of candidate match pairs is sorted and matches are finalized in order of decreasing confidence. A feature from the current or next frame may be matched only once. The algorithm maximizes the sum of confidence ratings.

The correctness of matches is evaluated qualitatively using interactive software showing two frames of features side-by-side. The user clicks on a feature on the left-hand side to see the corresponding search window and match, if any, on the right-hand side. Figure 51(a)-(b) show examples of successful and unsuccessful matching attempts. Figure 51(c)-(d) show the proportion of matched versus unmatched features. Figure 51(c) also shows short lines indicating the previous position of features. Although some errors are identifiable, most movements appear correct. A search window radius of 0.3 mm is generally reasonable.



(a)

(b)

(c)

(d)

Figure 51. Matching example: (a) selected matched features, (b) selected unmatched features, (c) matched features and (d) unmatched features.

5.1.2 Feature Tracking

Imperfections in fingerprint pre-processing and feature extraction invariably results in spurious and missing features, as well as errors in feature positions. Features also leave the field of view as the fingertip moves across the surface. As a result, the matching algorithm is generally capable of tracking features continuously only for a number of frames. No attempt is made to keep track of features through discontinuities. The result is a set of disjoint feature trajectories starting and ending at different frames. The temporal length of a trajectory is a good indicator of the feature's reliability. To improve the quality of measurements, features trajectories that do not span a minimal number of frames are rejected. Figure 52 shows an example of feature trajectory filtering. Rejecting trajectories shorter than 30 frames is generally reasonable.

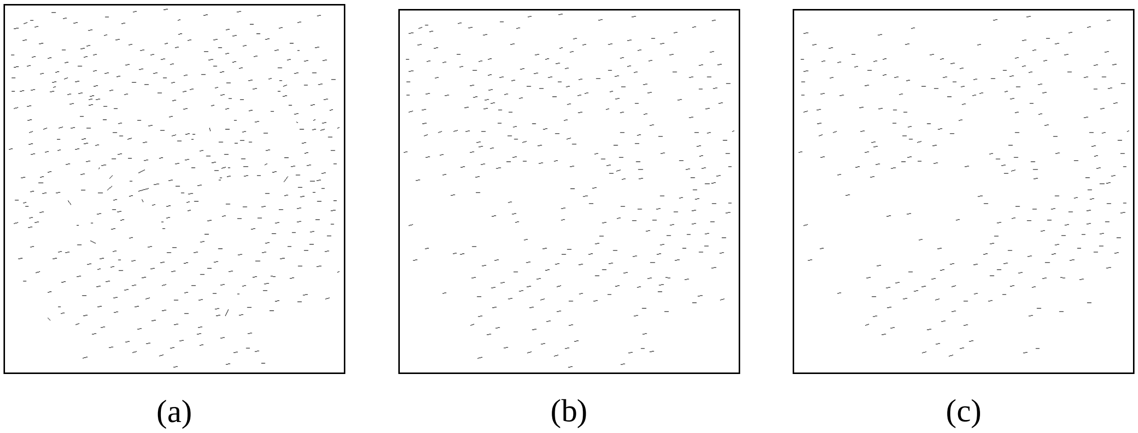

(a) (b) (c)

Figure 52. Effect of filtering short trajectories on feature flow: (a) no filtering, (b) minimum length of 30 frames, and (c) minimum length of 60 frames.

The discrete nature of the image grid as well as minor errors introduced by the feature extraction process often result in jagged feature trajectories. This problem is corrected by smoothing trajectories, resulting in sub-pixel floating-point feature coordinates. The smoothing operation consists of a convolution of a feature path *P(T)* with a Gaussian smoothing kernel *K(T)* as in Equation (5.3). The amount of smoothing depends on the filter's radius and standard deviation. Excessive smoothing may eliminate the very discontinuities that we are trying to measure. Figure 53 shows an example of trajectory smoothing.

(5.2) $$K(T) = e^{-\frac{T^2}{2\sigma^2}}$$

(5.3) $$S(T) = \left(\sum_{i=-R}^{+R} P_x(T+i) \bullet K(i), \sum_{i=-R}^{+R} P_y(T+i) \bullet K(i) \right)$$

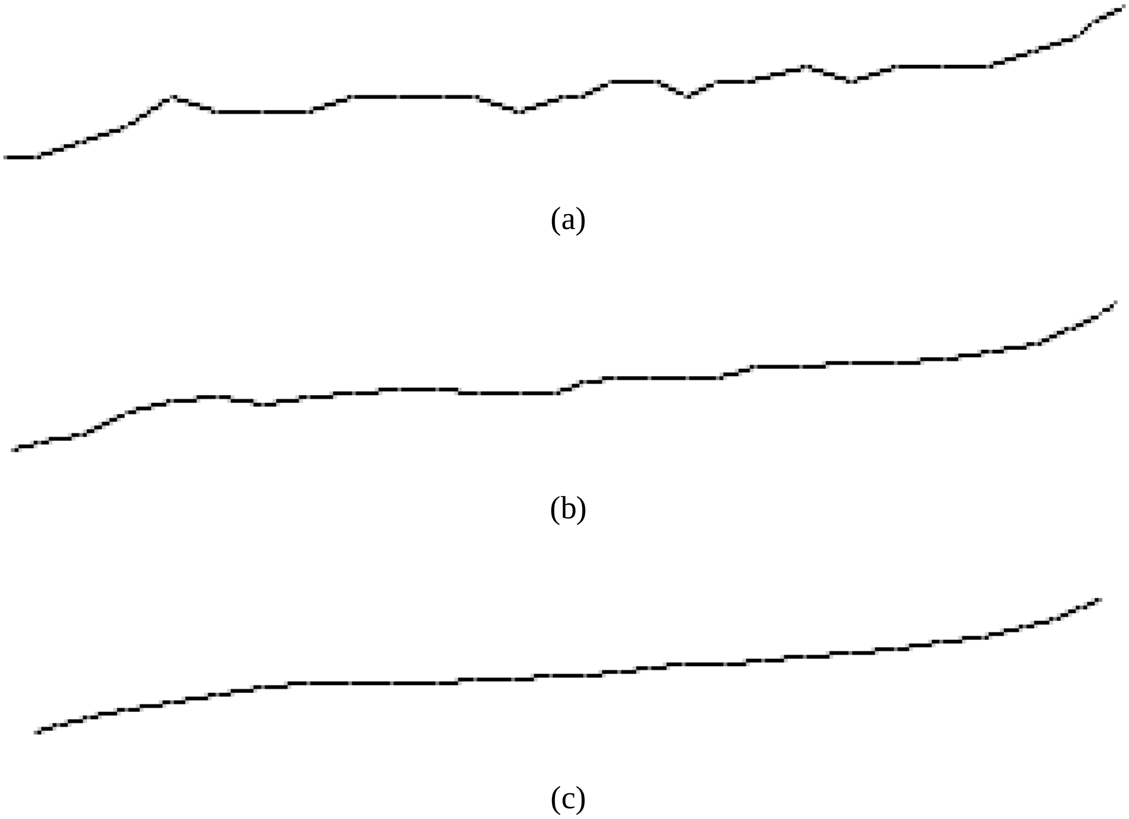


Figure 53. Trajectory followed by a tracked feature (successive positions in grey): (a) initial trajectory, (b) smoothed trajectory (diameter = 3, standard deviation = 1), and (c) smoothed trajectory (diameter = 7, standard deviation = 5).

5.2 Fingertip Tracking

The frame of reference of feature position measurements is the fixed surface over which the fingertip moves. A tactile display, on the other hand, applies a stress field to a fixed finger. Fingertip tracking is thus necessary to obtain a frame of reference that moves with the fingertip. The displacement of the fingertip is estimated by computing the average of tracked features displacements in a frame as illustrated in Figure 54.

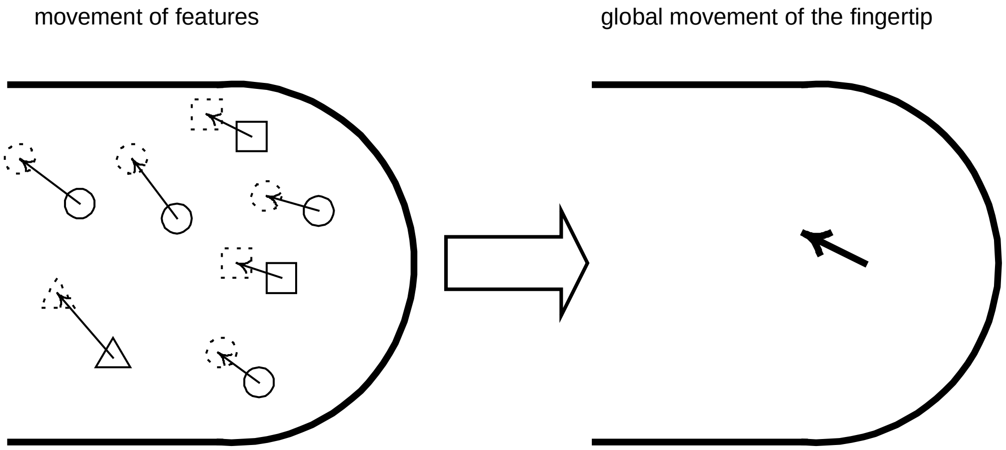
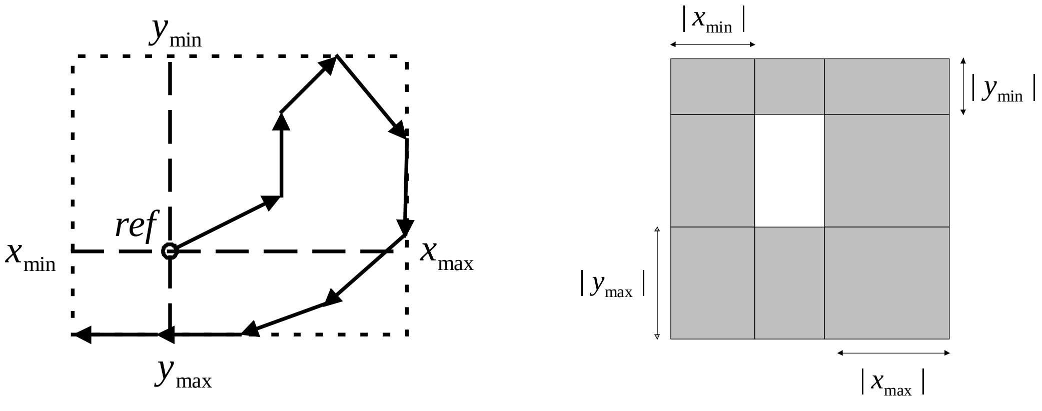


Figure 54. Fingertip tracking.

A tactile display's active area must also be mapped to a segment of the fingertip. The tactile display must be positioned to maximize the visibility of the associated patch of skin. The optimal position is determined by locating and tracking the part of the fingertip that is visible through an entire sequence of images, or visibility window. The visibility window is found by examining the trajectory followed by the fingertip. The extrema of the trajectory are illustrated in Figure 55(a). The initial position and dimensions of the window are shown in Figure 55(b). The window then follows the trajectory of the fingertip. The skin patch associated with the tactile display should track the visibility window to optimize the measurement coverage.



(a) (b)

Figure 55. Computation of the visibility window: (a) trajectory followed by the fingertip, and (b) part of the initial image that will be visible through the entire sequence.

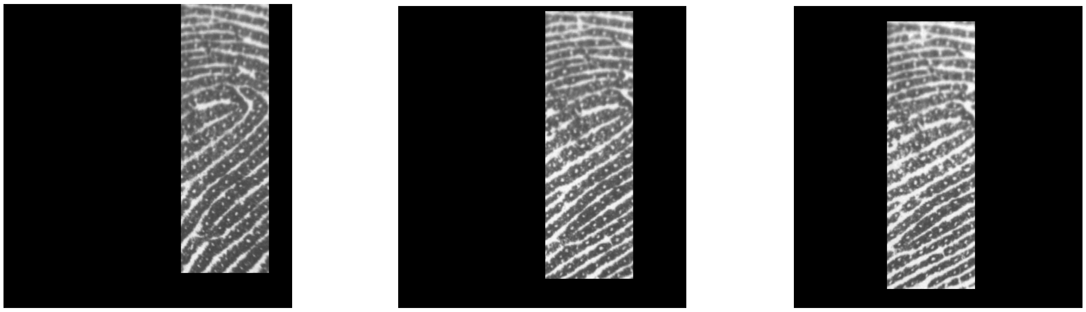

(a) (b) (c)

Figure 56. Visibility window example: (a) frame 0, (b) frame 50, and (c) frame 75.

5.3 Skin Stretch Measurement

Sections 5.1 and 5.2 described the algorithms used to track the fingertip and its features. This section discusses two methods used to extract skin stretch information from the tracked features. The first method relies on relative changes in triangle area or edge length of a Delaunay triangulation of tracked features. The second method relies on the relative movement of small patches of skin.

5.3.1 Triangulation

Changes in local skin stretch can be estimated by observing changes in a triangulation of tracked features. Consider a pair of consecutive frames. The subset of features of the first frame that are tracked in the second frame is used to construct a Delaunay triangulation. A Delaunay triangulation has the property that “the circumscribing circle of any face of the triangulation contains no point of the set in its interior” [40]. A typical feature triangulation is shown in Figure 57. The triangulation is maintained in the second frame. The change in local skin stretch is evaluated using one of two methods. The first method consists of computing the relative change in area of each face as illustrated in Figure 58(a). The alternative consists of measuring the change of edge lengths as illustrated in Figure 58(c). Each pair of images is analysed, yielding a map of relative changes in skin stretch over time.

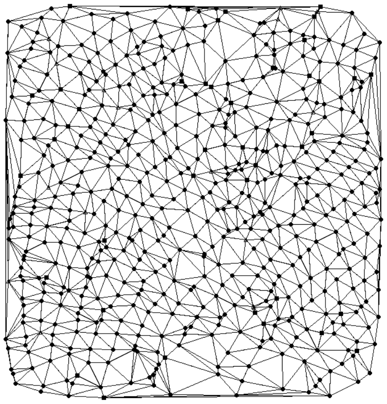
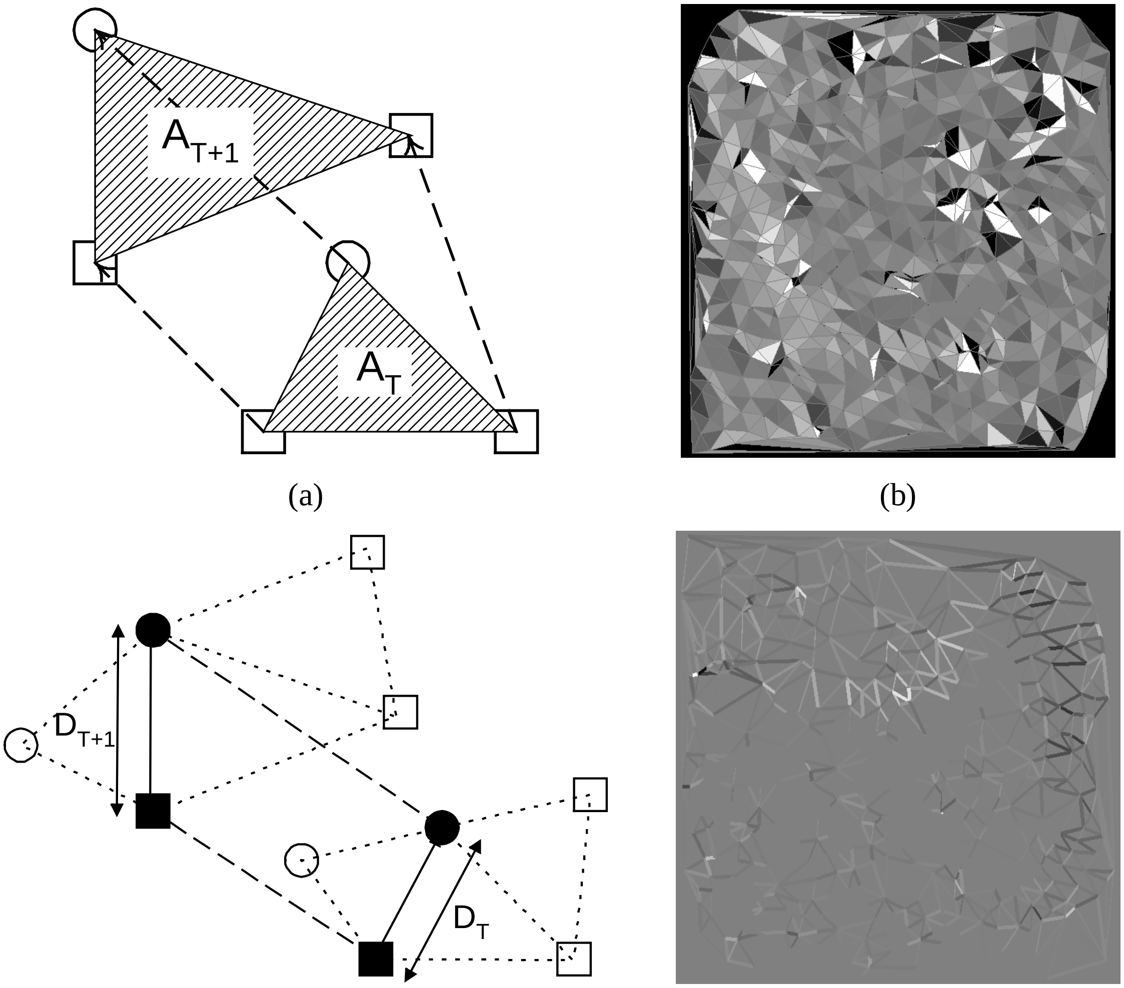


Figure 57. Delaunay triangulation of matched features.



(a) (b)

(c) (d)

Figure 58. Measurement of local skin stretch change by triangulation (white/black = +/-10%): (a) change in face area, (b) area change example, (c) change in edge length, and (d) length change example.

5.3.2 Local Movement

The change in skin stretch can also be estimated using the local movement of patches of skin. The displacement of a patch of skin is computed by taking the average of all feature movements relative to the fingertip (i.e. absolute movements minus fingertip movement) in a window of a given shape and size as illustrated in Figure 59. Figure 60 shows an example of average local movement computation.

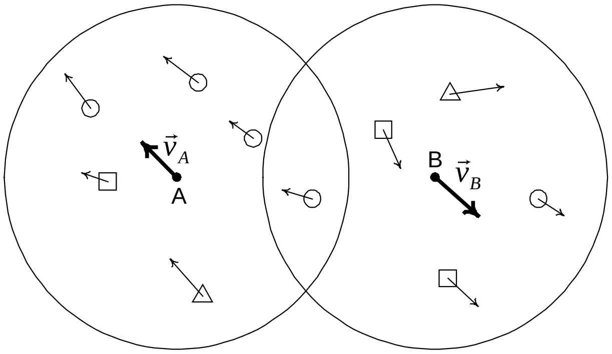
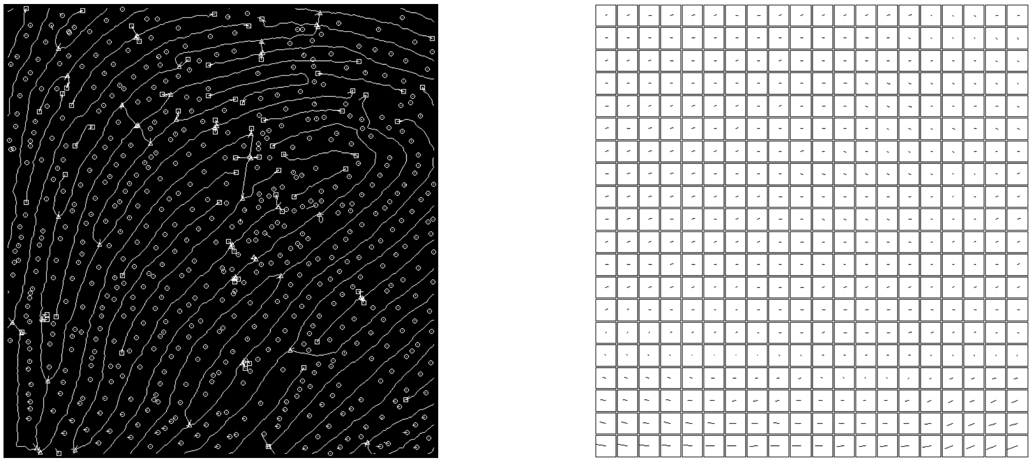


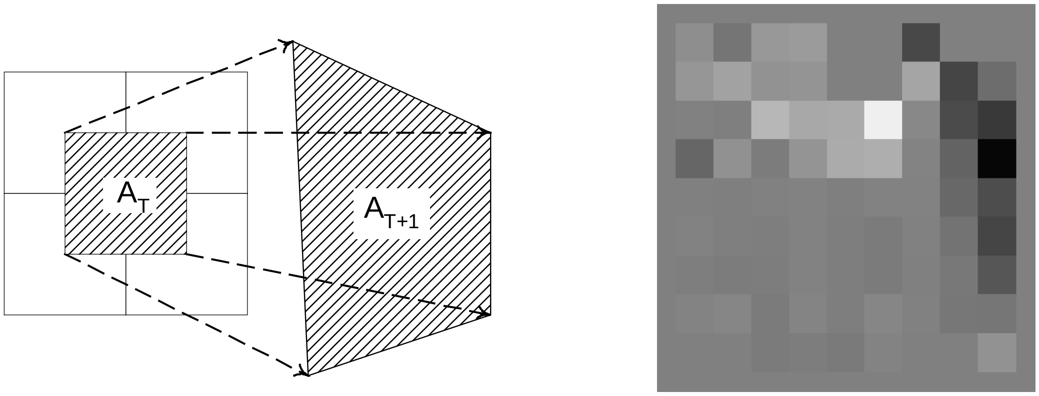
Figure 59. Average local movement illustration (disk shape).



(a) (b)

Figure 60. Average local movement example: (a) feature movement relative to the fingertip, and (b) average local movement map with circular averaging window of radius 1 mm and scaling to a maximum of 0.02 mm.

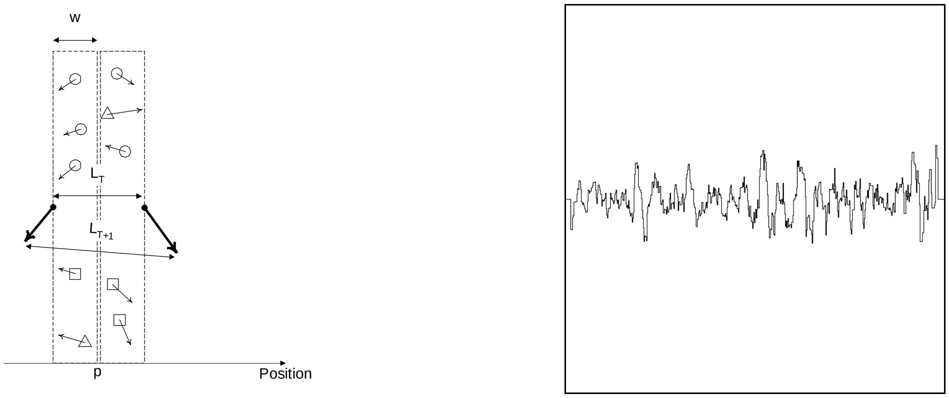
The relative movement of neighbouring patches of skin indicates the change in local skin stretch. Consider four adjacent patches of skin as illustrated in Figure 61(a). The current area A_T is given by the distance between the centres of patches of skin. The area of the tetragon formed by the centres of the patches after movement is the next area. An example of relative area change is illustrated with levels of grey in Figure 61(b).



(a) (b)

Figure 61. Change in skin stretch from local movement: (a) illustration, and (b) example for 10 x 10 local movement map (white/black = +/- 10%).

Assuming that the finger slides mostly horizontally, it is also possible to evaluate the variations in skin stretch along the x-axis. Consider vertical skin strips of width w on each side of point p on the x-axis. The average movement in the two strips is used to estimate the change in stretch at p as illustrated in Figure 62.



(a) (b)

Figure 62. Horizontal change in stretch: (a) illustration, and (b) example.

5.4 Tactile Display Mapping

The STReSS tactile display consists of an array of 10 x 10 skin contactors spaced evenly over a 10 mm by 10 mm square. The tactile display provides a programmable lateral strain field through the displacement of 100 skin contactors moving laterally in one axis. The local skin movement measured in section 5.3.2 is mapped to changes in skin contactor positions to reproduce a similar skin stretch on the skin of a user.

The 10 mm x 10 mm surface of the tactile display is associated with a patch of fingertip skin of the same size, centred on the visibility window. The patch of skin is divided in a grid of 10 x 10 squares. The local movement of each square segment of skin is computed as the average feature movement inside that square (see section 5.3.2). The skin contactor associated with a patch of skin should ideally move by the same amount and in the same direction, as shown in Figure 63(a). Assuming that the skin is initially at rest, the skin contactor's ideal position can be estimated by cumulating the movement vectors since the beginning of the image sequence as shown in Figure 63(b). Note however that the skin contactor is constrained to move horizontally. The horizontal position can be determined either by rotating the ideal position vector (Figure 63(c)) or by projecting it on the horizontal (Figure 63(d)).

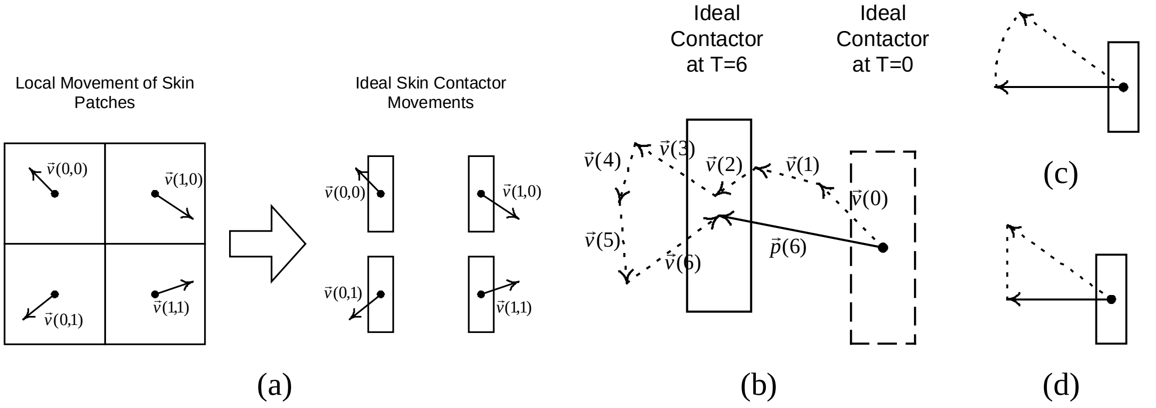


Figure 63. Skin contactor mapping: (a) illustration, (b) ideal actuator position, (c) actuator position by rotation, and (d) actuator position by projection.

As the tactile display tracks the visibility window, some of its contactors may fall outside the field of view of the camera. When this occurs, the position of the occluded contactors is reset. The same rule is applied when no feature movement is available in the square window of a skin patch. Figure 64 and Figure 65 give examples of tactile display mappings.

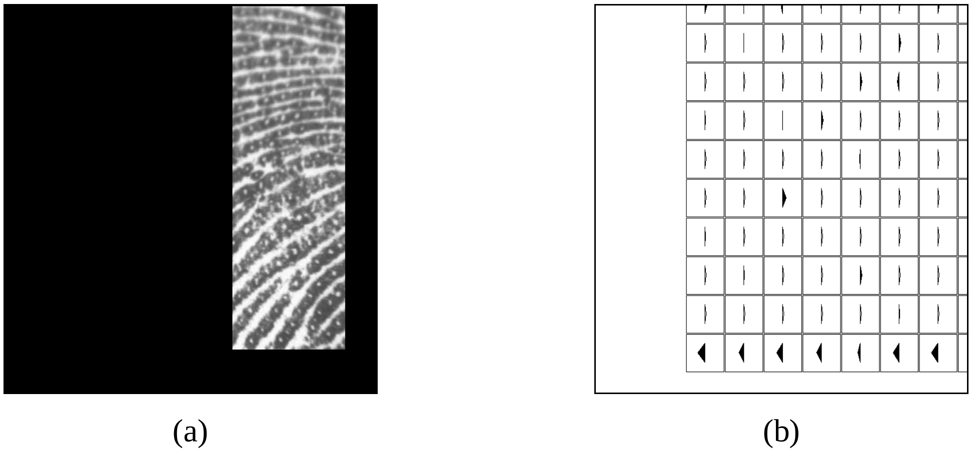


Figure 64. Position of tactile display with respect to imaging system: (a) visible window, and (b) the tactile display tracking the visible window.

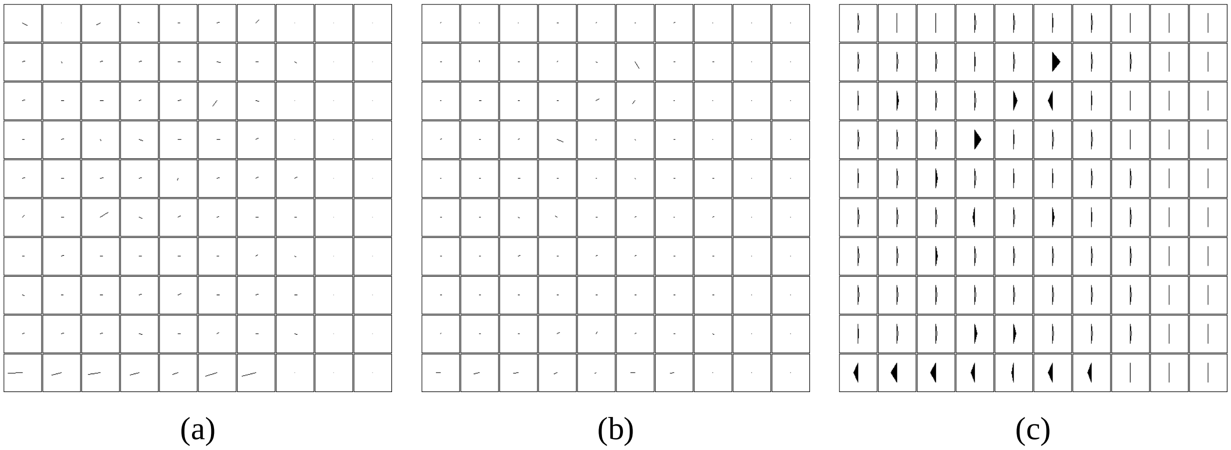
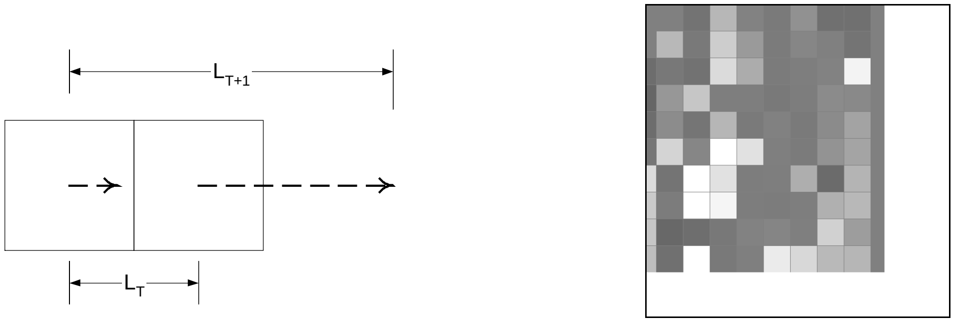


Figure 65. Skin contactor mapping (scaled by a factor of 10): (a) ideal movements, (b) ideal positions, and (c) positions by projection.

The change in stretch can be computed from the position or change in position of actuators as illustrated in Figure 66. The change in stretch is found from the change in distance between moving contactors. A representation similar to Figure 66(b) is obtained.



(a) (b)

Figure 66. Change in stretch from contactors: (a) illustration, and (b) example.

CHAPTER 6   EXPERIMENTAL RESULTS

This chapter presents the results of experiments testing the capabilities and limitations of the skin stretch measurement method developed in the previous chapters. Three test surfaces were used: a flat surface, a surface with a ridge, and a surface with a valley. Sections 6.1 and 6.2 present the results of tests aimed at the feature extraction and feature tracking subsystems respectively. Section 6.3 presents results of skin stretch measurement experiments. Section 6.4 concludes the chapter with the testing of the method as a signal generator for the STReSS tactile display.

6.1 Feature Extraction

A quantitative evaluation of the feature extraction system is beyond the scope of this thesis. This section provides results and observations based on the visual inspection of the output of the feature extraction system. Three aspects of the feature extraction performance are evaluated: the correctness of extracted features, the accuracy with which they are located, and the robustness of the system to fingertip variations.

The correctness of the extracted features was evaluated by visually inspecting the superposition of colour- and shape-coded results over the greyscale fingerprint. The feature extraction algorithm generally finds a large number of features correctly but also misses many others. Spurious features and misclassified features occur frequently but tend to be impossible to track and thus rejected during the tracking phase. The correctness of the features is often difficult to assess as can be seen in Figure 67.

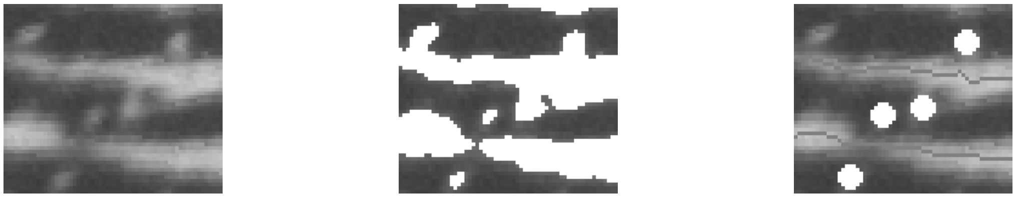

(a) (b) (c)

Figure 67. Inspection of extracted features: (a) greyscale segment, (b) binary overlay, and (c) feature overlay.

The accuracy of the position of extracted features is difficult to evaluate precisely by visual inspection. Although the current implementation of the algorithms artificially limits the precision of positions to pixel coordinates, it isn't clear that a better precision could be obtained. The precision depends in large part on the ability of the image acquisition and binarization system to capture the true shape of the fingerprint ridges, valleys and pores. Variations in pressure distribution are likely to bias the binarization of a valley or pore and shift their position slightly from frame to frame. Similarly the thinning algorithm may cause slight errors in the location of the fingerprint skeleton. The combined effect of these sources of errors is difficult to analyse. It is doubtful that the position of a feature could be known at a precision of more than ±1 pixel with the current system.

The robustness of the feature extraction system was evaluated by examining the features extracted from the fingertips of six subjects. Subjects were instructed to slowly slide a finger horizontally across a flat surface. Subjects adjusted the pressure applied with the fingertip by monitoring the quality of fingerprint images. Sequences of 180 frames (3 seconds at 60 frames per second) were acquired for each test subject. The average speed of the fingertip varied between 2.8 mm/s and 5.3 mm/s. The dimensions of the images, after calibration, were approximately 10.3 x 10.8 mm. Two of the six test subjects were rejected due to the poor quality of the acquired images. The best sequence was selected for each of the four remaining subjects. Apart from lowering the binarization thresholds in the fourth test, no further adjustment was necessary. Figure 68 shows typical results for the four selected sequences. The first fingertip, which was used in all other tests performed in this thesis, has a very good ridge definition and clear, large pores. Although the feature extraction process is not perfect, a large number of features are located correctly. In the last three tests, the ridge definition remains fairly good but the pores are much smaller and difficult to see. The feature extraction still detects many features but their density and reliability may be insufficient for skin stretch measurements. It appears from these tests that the current algorithms are not sufficiently robust to handle arbitrary fingertips. Results could be improved by exerting more control over the applied pressure and the condition of the subject's skin, for example by moistening the fingertip.

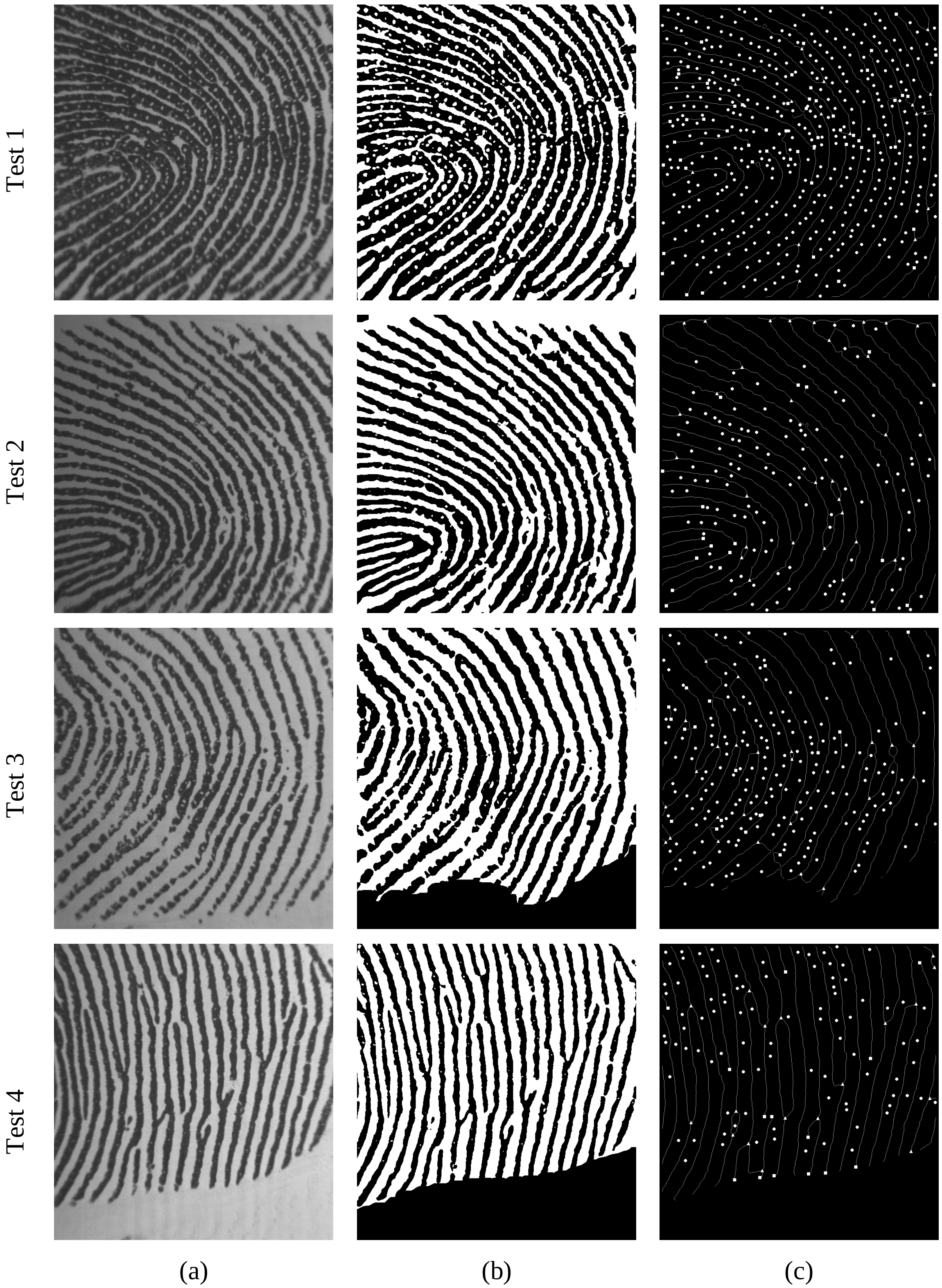


Figure 68. Typical feature extraction for four test subjects: (a) fingerprint image, (b) binary fingerprint, and (c) extracted features.

6.2 Feature tracking

The feature tracking performance was evaluated by visual inspection of the feature matches in successive frames, with the help of the interactive software introduced in section 5.1.2 (see Figure 51). Although a majority of correct matches is achieved, errors are impossible to avoid completely. Pore matching is especially problematic due to their high density and lack of distinguishing features.

Figure 69 illustrates a type of tracking failure that occasionally occurs. The three white-filled features are incorrectly matched by the tracking algorithm due to an accumulation of movement prediction errors. Matching errors may cause the predicted movement to veer away from the correct trajectory. In degenerate cases, an accumulation of errors causes a trajectory to jump from one pore to the next along a ridge. Incorrect tracking trajectories tend to be short and are thus generally eliminated during trajectory filtering.

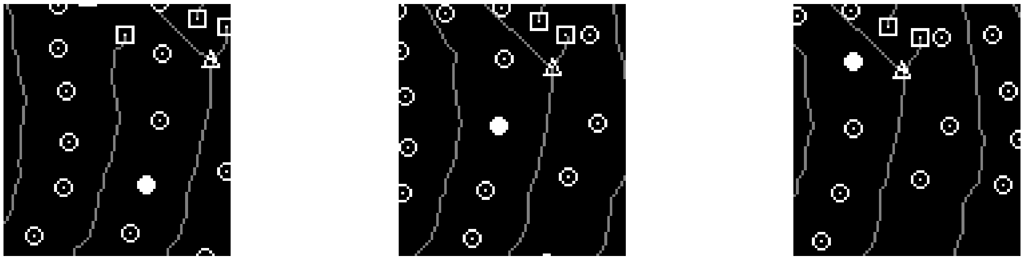

(a) (b) (c)

Figure 69. Example of tracking failure: (a) segment of frame 21, (b) segment of frame 22 and (c) segment of frame 23.

The performance of the tracking system was evaluated more quantitatively by observing the distribution of trajectory lengths in a frame. Figure 70 shows the distribution averaged over a sequence of images acquired as a finger sled over a flat surface. Although some trajectories are short, a significant proportion of the trajectories spans 50 or even 100 frames. This indicates that the system detects an acceptable number of long and reliable trajectories. The average length of distributions is likely to decrease, however, in more difficult cases such as when sliding over a ridge.

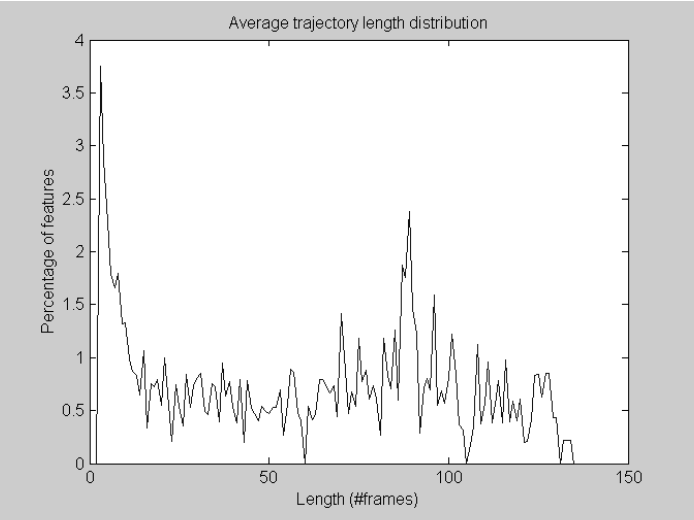


Figure 70. Trajectory length distribution averaged over a sequence of frames.

Lastly it is interesting to estimate how robust the tracking system is to fingertip velocity. While fast moving features can generally be tracked accurately (see Figure 71), the tracking system is likely to fail if the velocity of the fingertip changes rapidly, such as when changing direction.

(a) (b) (c)

Figure 71. Fast movement to the left: (a) frame 97, (b) frame 98, and (c) feature movement.

6.3 Skin Deformation Measurements

This section describes the results of experiments with skin deformation measurements. All experiments shown in this section were performed using the author's left-hand index fingertip. Section 6.3.1 presents some interesting results obtained with flat surfaces. Section 6.3.2 presents results obtained while sliding over a valley or ridge. 6.3.3 ends the section with a discussion of the results.

6.3.1 Flat Surface

Measurements cases obtained from movement over flat surfaces are generally difficult to interpret. This section provides interesting measurements obtained from select sequences of images for which a meaningful interpretation could be found. In the first example, a fingertip is firmly pressed against the surface and rotated. In the second example, a fingertip is pressed firmly and moved back and forth horizontally.

6.3.1.1 Rotation

This section describes the measurement of skin stretch variations in a sequence of images showing a fingertip pressed firmly against a flat surface and rotated (flat1.avi). While no obvious pattern emerges from measurements at most frames, a very clear and interesting phenomenon is visible at frame 70. At that point, the fingertip begins a counter-clockwise rotation. Most of the fingertip is sticking to the glass (Figure 72(a)-(b)). The top part of the finger is moving up, stretching the intermediate zone between the moving and non-moving segments. The right-hand part is moving toward the upper-left corner, resulting in compression at the junction of the moving and non-moving parts. This observation is very clear in the changes in triangulation (Figure 72(c)–(d)). The change in skin patch area found from local motion agrees (Figure 72(f)). These results are even more obvious in the variations from frame 70 to 75 (Figure 73). The change in triangulation edge length is particularly interesting. Notice that the fingerprint seems to be expanding itself vertically but compressing itself horizontally.

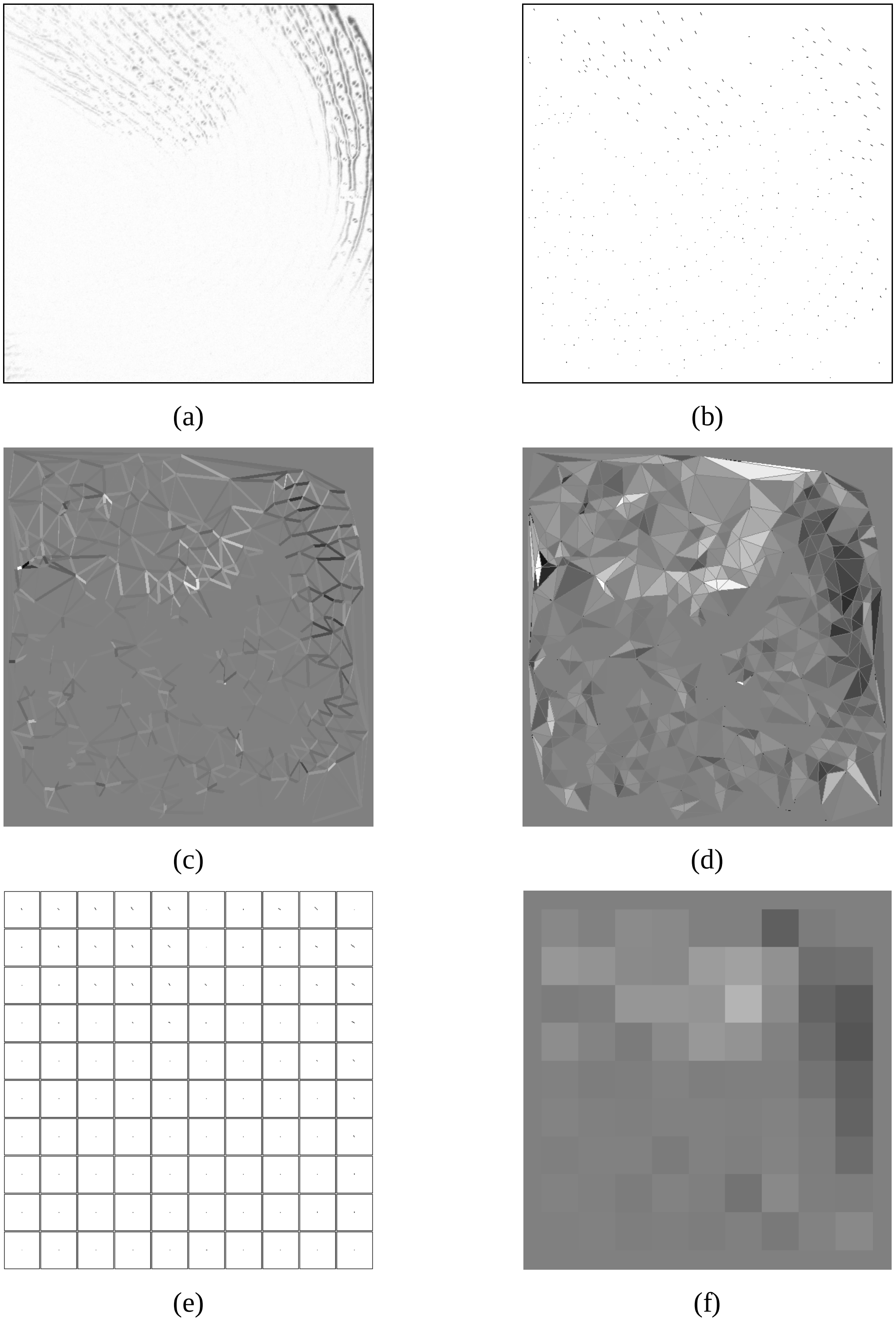

(a) (b) (c) (d) (e) (f)

Figure 72. Measurements from frame 70 to 71 of flat1.avi (white/black=+/-10%): (a) frame difference, (b) feature movement, (c) edge length change, (d) triangle area change, (e) movement of skin patches, and (f) skin patch area change.

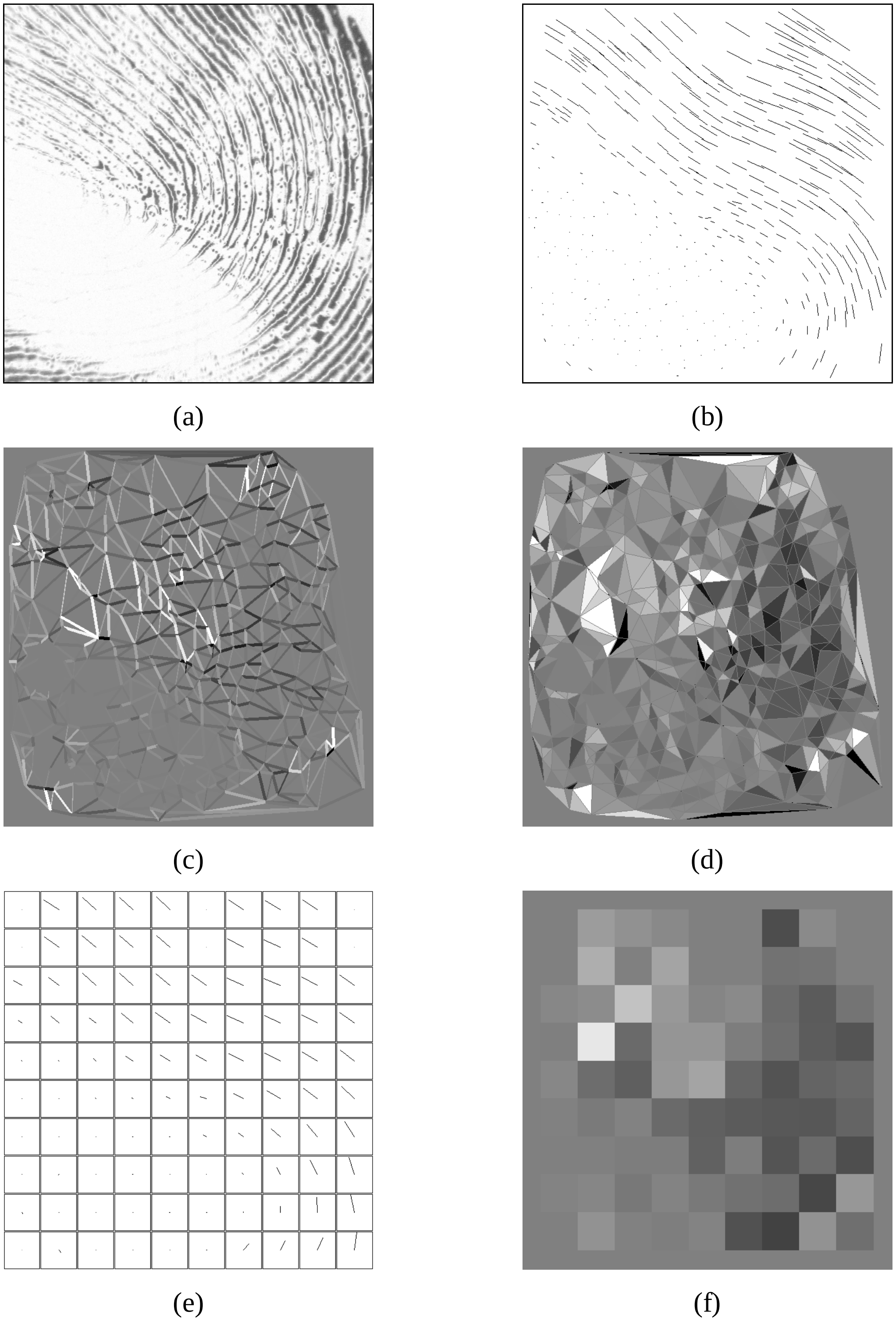
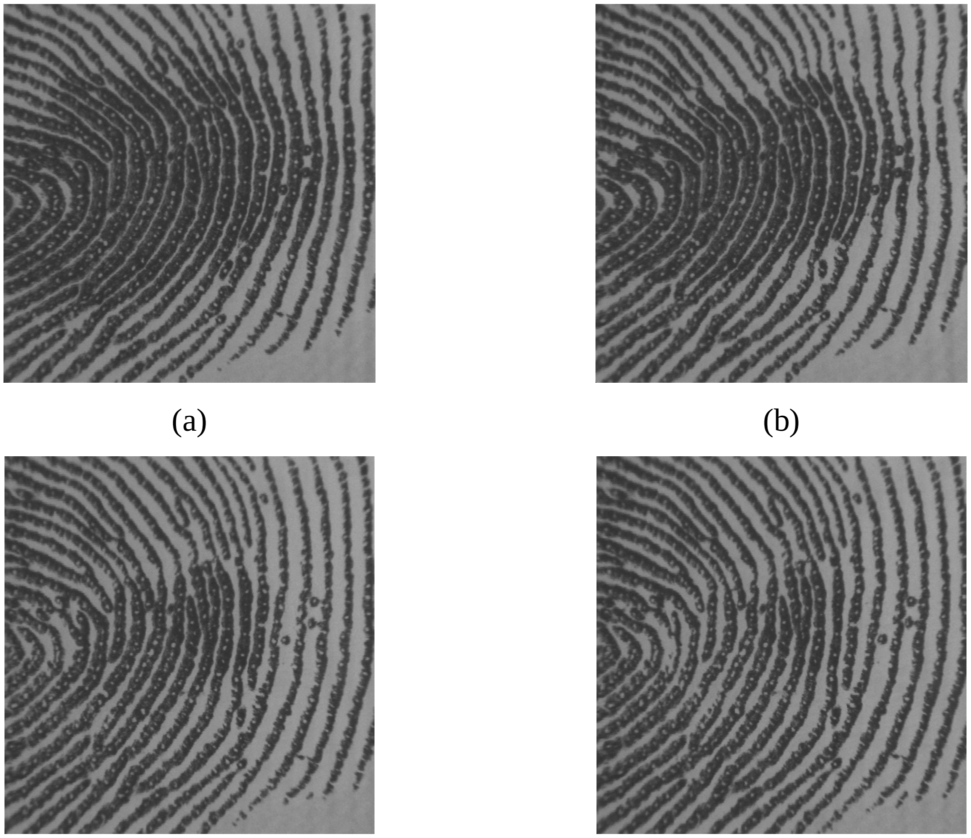


Figure 73. Measurements from frame 70 to 76 of flat1.avi (white/black=+/-20%): (a) frame difference, (b) feature movement, (c) edge length change, (d) triangle area change, (e) movement of skin patches, and (f) skin patch area change.

6.3.1.2 Back and forth

This section describes interesting results obtained from a sequence of images in which a finger pressed firmly against a flat surface moves back and forth horizontally (flat2.avi). A patch of skin is sticking to the glass while the surrounding skin moves with the finger. Again most measurements are difficult to interpret. Figure 74 shows a sequence of images in which the fingertip is returning from the left.

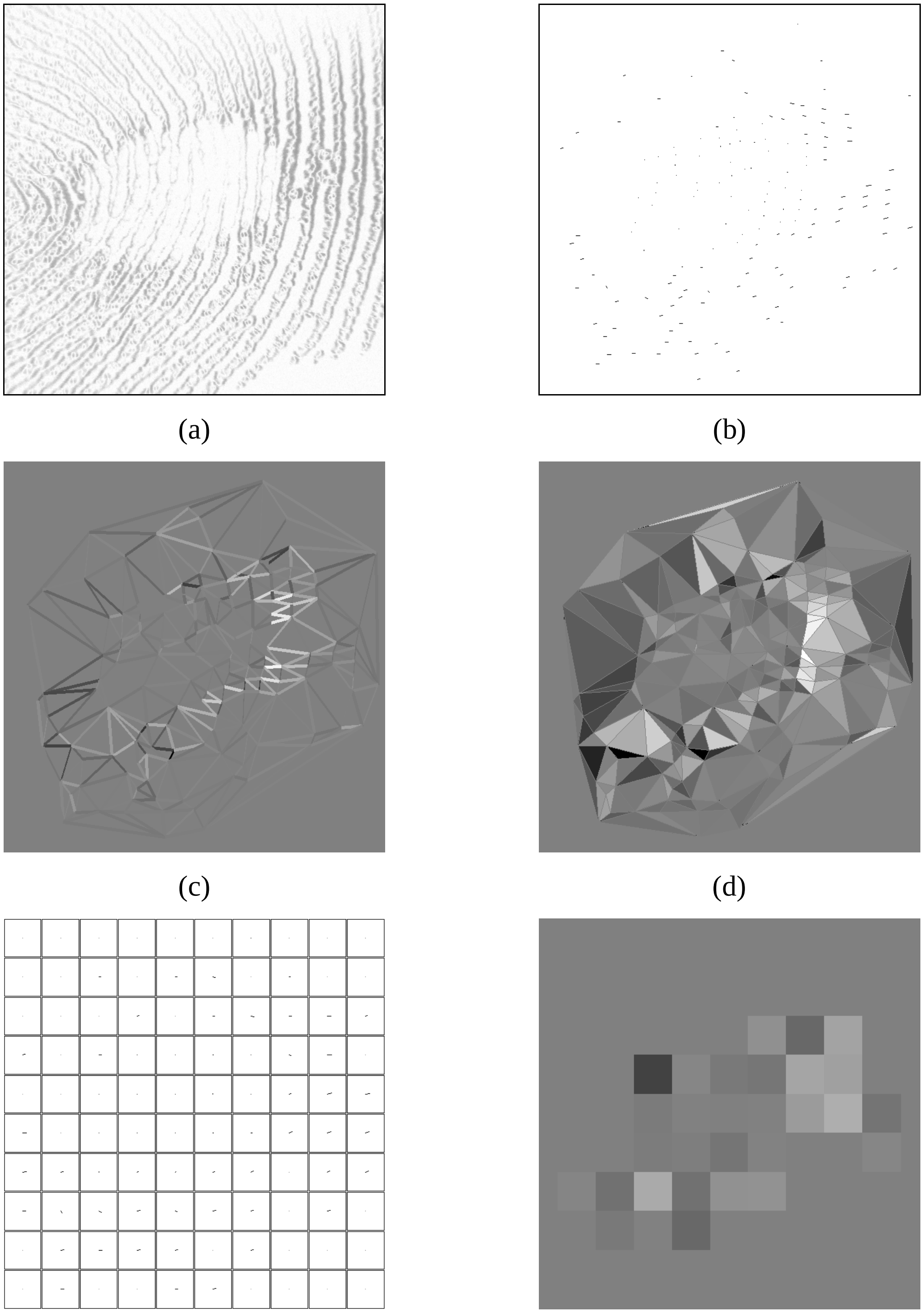


(a) (b)

(c) (d)

Figure 74. Fingertip returning from the left (flat2.avi): (a) frame 26, (b) frame 28, (c) frame 30, and (d) frame 32.

Figure 75 shows the variations in stretch from frame 30 to 31, near the end of this sequence. At that point, the fingertip is moving back from the left and is about to stop at its rest position. The centre is stationary while the sides are moving right. As a result there is compression to the left of the stationary patch and expansion to the right.



(a) (b)

(c) (d)

(e) (f)

Figure 75. Measurements from frame 30 to 31 of flat2.avi (white/black = +/-30%): (a) frame difference, (b) feature movement, (c) edge length change, (d) triangle area change, (e) movement of skin patches, and (f) skin patch area change.

6.3.2 Ridge/valley surfaces

This section presents the results of a series of tests performed to investigate the effect of a ridge or valley on the deformation of a sliding finger. Two sets of tests were performed. The first set consists of sequences in which the entire fingertip is visible (approximately 10.3 x 10.9 mm after calibration). The second set consists of a close-up on a smaller segment of the fingertip (approximately 6.1 x 6.2 mm after calibration). Each set consists of three sequences: a finger sliding over a flat surface, over a surface with a ridge and over a surface with a valley. The ridge and valley have a Gaussian profile with a width of 3 mm and a height of 0.5 mm (see section 3.3 for details). In all sequences, the fingertip is moving slowly from left to right at an average speed varying from 0.58 mm/s to 2.8 mm/s. Sequences are named flatL.avi, ridgeL.avi and valleyL.avi in the first set, and flatS.avi, ridgeS.avi and valleyS.avi in the second set.

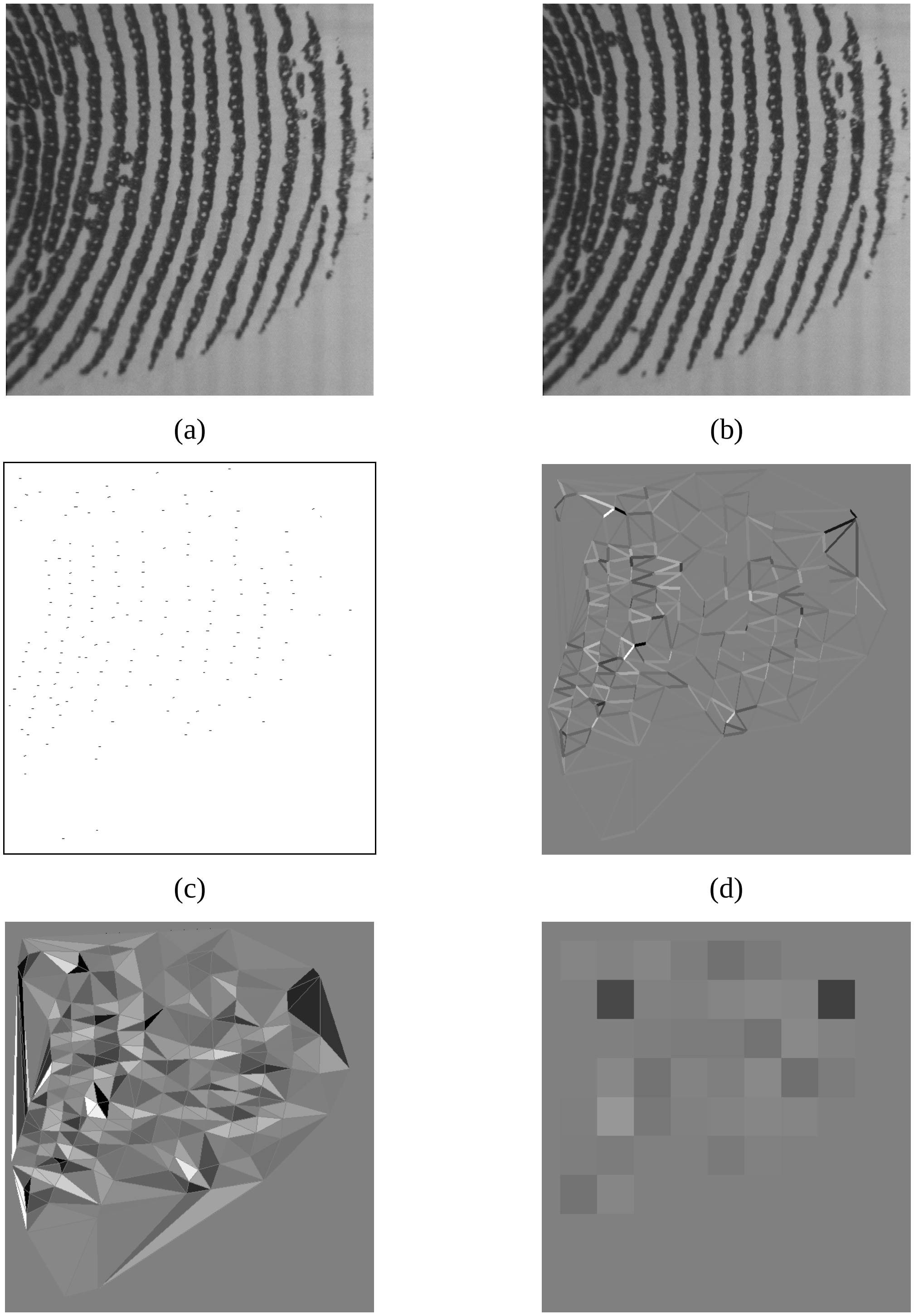
The same analysis is performed for each sequence of 180 frames. A first figure shows the variations in skin stretch over two consecutive images (Figures 76, 79, 82, 85, 88 and 91). This figure first shows the two greyscale fingerprints and the flow of tracked features. The width and approximate position of the valley or ridge is indicated by two dotted lines spaced by 3 mm on fingerprint images. Note however that the exact position of the shape is not known and can only be guessed by looking at variations in illumination. The next diagrams show the relative change in triangulation edge length and in face area. The figure finally shows the change in area of skin patches computed using the local motion of skin patches (see section 5.3.2). A second figure shows similar results computed over a span of 11 frames (Figures 77, 80, 83, 86, 89 and 92). The extended measurement span reduces the effect of noise and often provides cleaner results. A third figure shows the variations in triangulation edge length over a span of 11 frames at 6 positions in the sequence (Figures 78, 81, 84, 87, 90 and 93). This last figure shows that the observations made in the previous two figures apply equally well to other frames in the sequences. Figure 94 shows the relative change in stretch along the horizontal axis (see section 5.3.2) averaged over the entire sequence of images, for each sequence.

Figure 76, Figure 77 and Figure 78 show measurements obtained with a flat surface. Although no obvious pattern is discernible in the measurements, a significant amount of skin deformation is present. This observation is confirmed in Figure 94(a).

Figure 79, Figure 80 and Figure 81 show measurements obtained with a ridge. Although the exact position of the ridge is not known, the regions with darker illumination indicate that it is near the centre of the image. The measurements in Figure 79 show a slight tendency of compression on the left of the ridge and a tendency of expansion on the right. This phenomenon is much clearer in Figure 80, in which measurements are made over a span of 11 frames. Figure 81 confirms that this phenomenon occurs throughout the sequence. These observations are confirmed in Figure 94(c).

Figure 82, Figure 83 and Figure 84 show the measurements obtained with a valley. Again the position of the valley is indicated by the lighting intensity over the image. Measurements show a phenomenon opposite to that found in the case of the ridge. A tendency of expansion is clearly visible on the left while a tendency of compression is visible on the right. These observations are confirmed in Figure 94(e).

The second set of sequences (Figure 85-93) confirms the previous observations. Although this second set should in theory provide more accurate results due to the increase in feature size, there is no obvious improvement in the measurements.

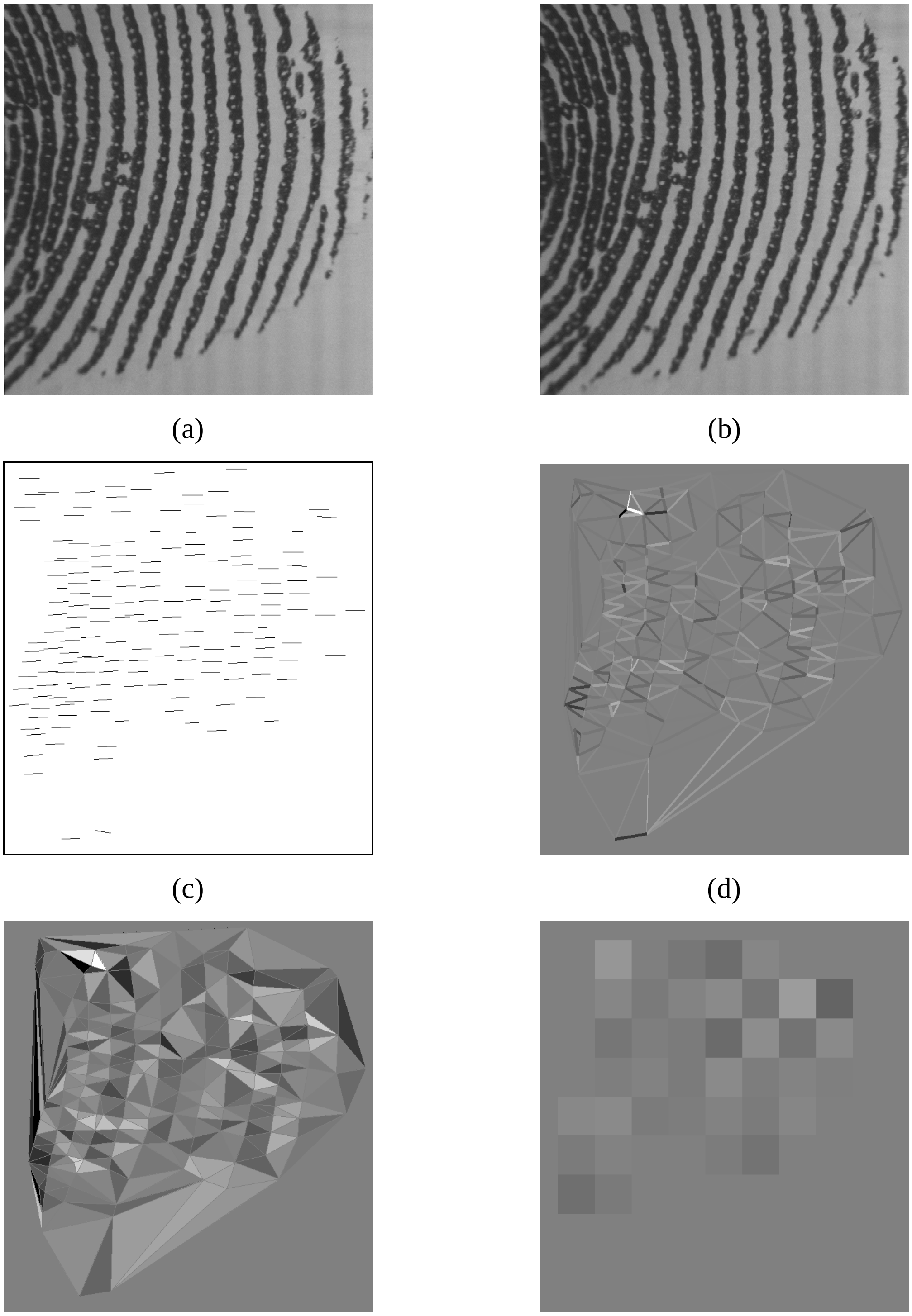


(a) (b)

(c) (d)

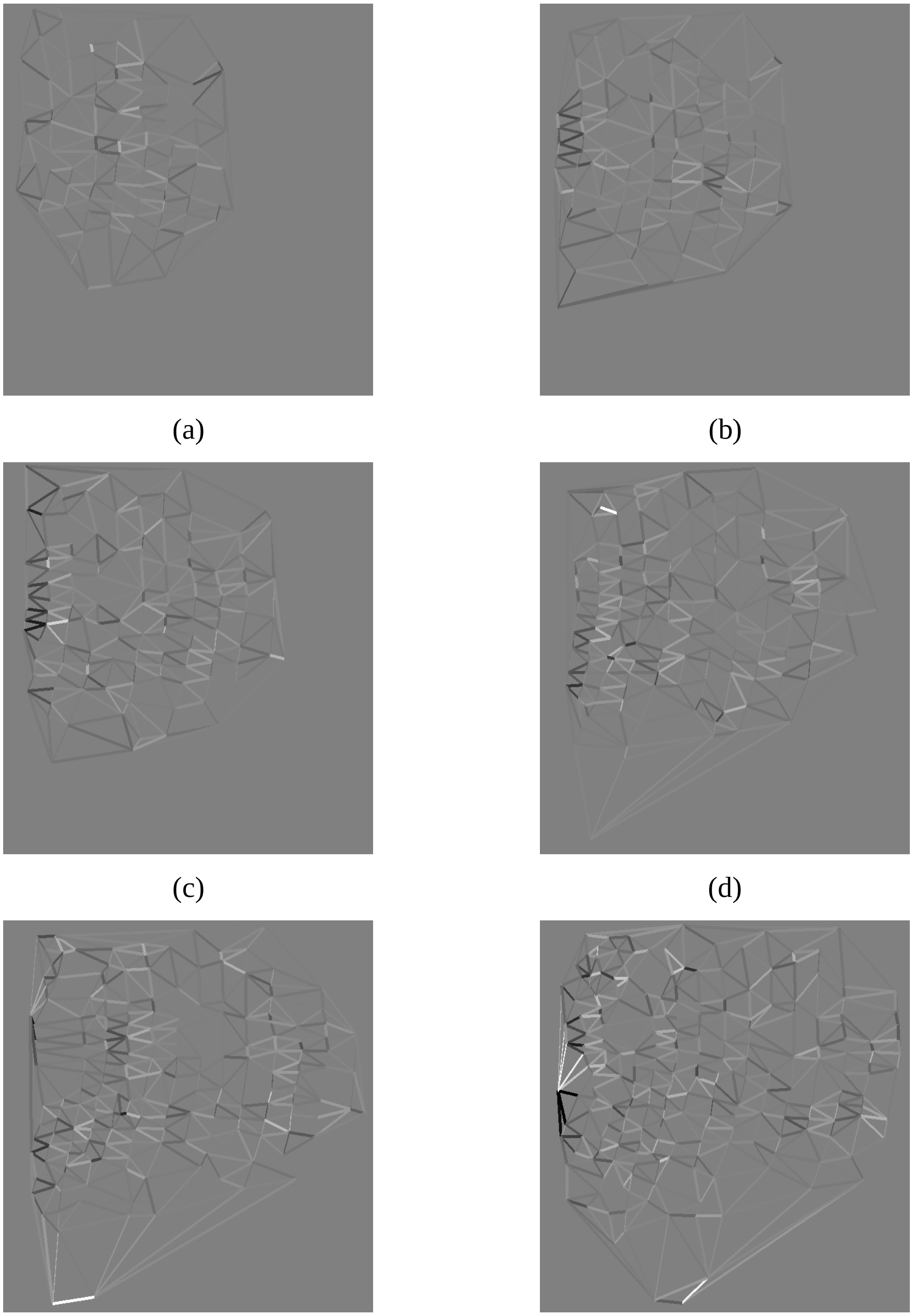
(e) (f)

Figure 76. Measurements from frame 115 to 116 of flatL.avi (white/black = +/-5%): (a) frame 115, (b) frame 116, (c) feature movement, (d) edge length change, (e) triangle area change, and (f) skin patch area change.



(a) (b) (c) (d) (e) (f)

Figure 77. Measurements from frame 115 to 126 of flatL.avi (white/black = +/- 15%): (a) frame 115, (b) frame 126, (c) feature movement, (d) edge length change, (e) triangle area change, and (f) skin patch area change.

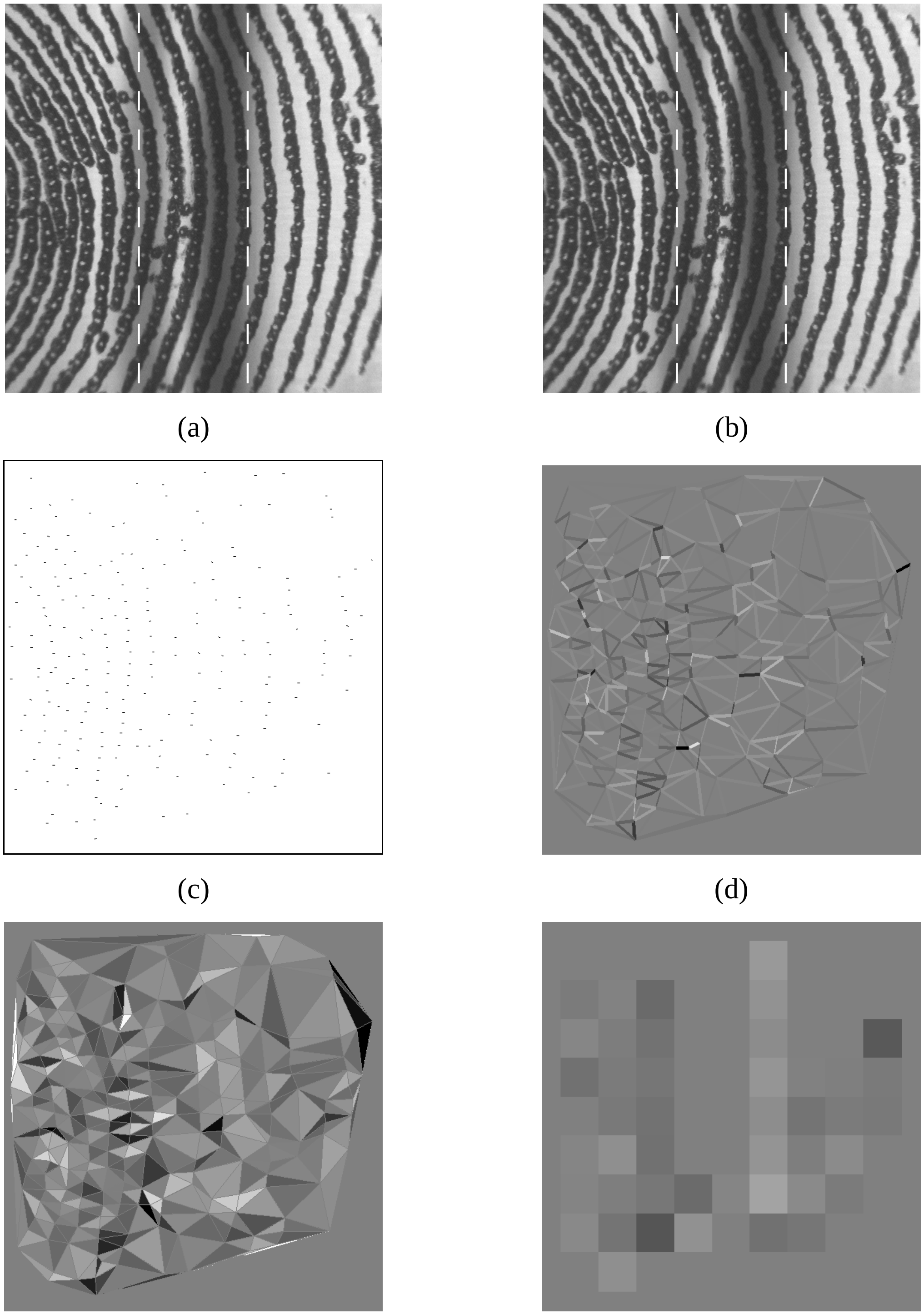


(a) (b)

(c) (d)

(e) (f)

Figure 78. Measurements of edge length change in flatL.avi (white/black = +/-15%): (a) frames 25-36, (b) frames 50-61, (c) frames 75-86, (d) frames 100-111, (e) frames 125-136, and (f) frames 150-161.

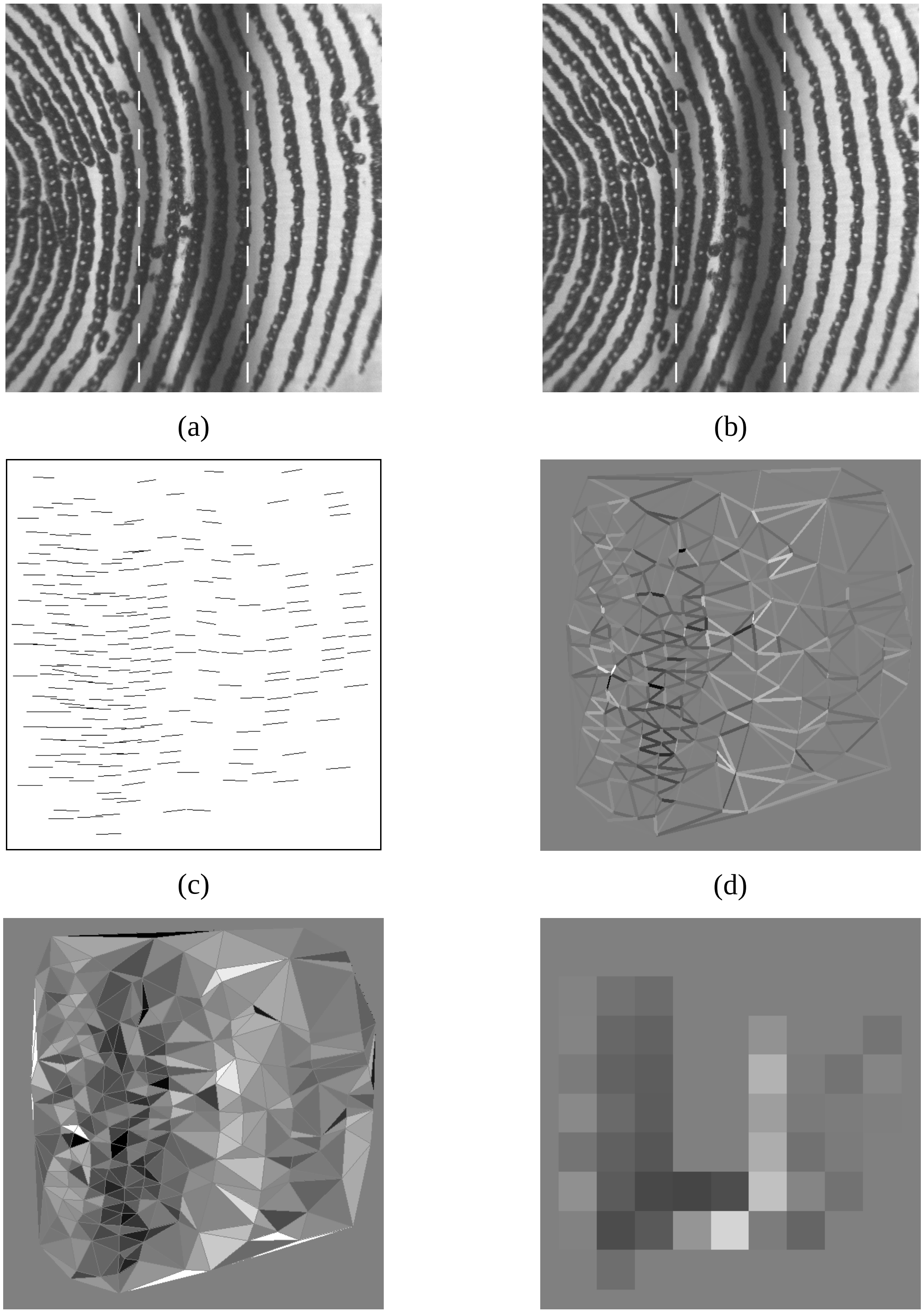


(a) (b)

(c) (d)

(e) (f)

Figure 79. Measurements from frame 112 to 113 of ridgeL.avi (white/black = +/-5%): (a) frame 112, (b) frame 113, (c) feature movement, (d) edge length change, (e) triangle area change, and (f) skin patch area change.

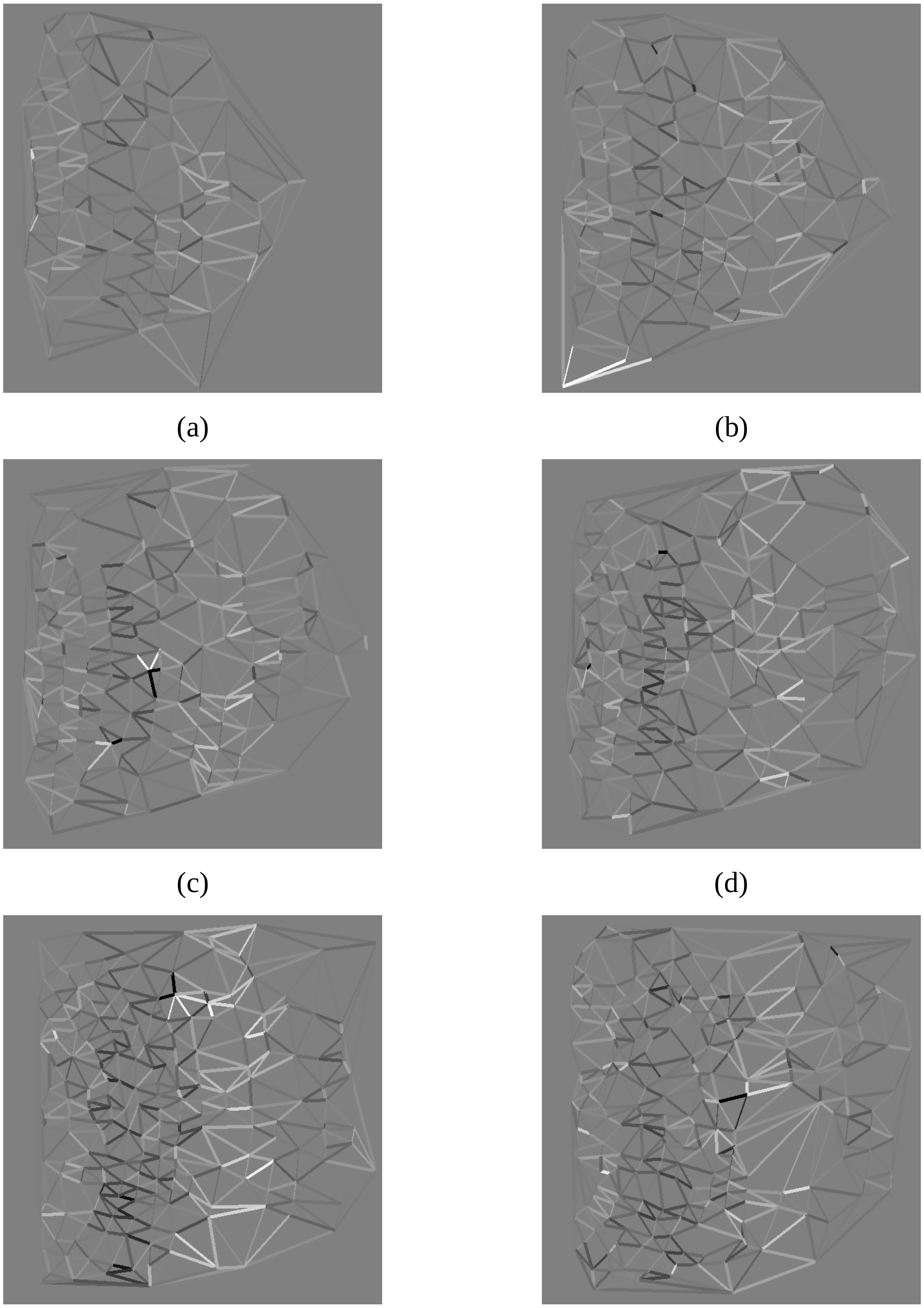


(a) (b)

(c) (d)

(e) (f)

Figure 80. Measurements from frame 112 to 123 of ridgeL.avi (white/black = +/- 15%): (a) frame 112, (b) frame 123, (c) feature movement, (d) edge length change, (e) triangle area change, and (f) skin patch area change.

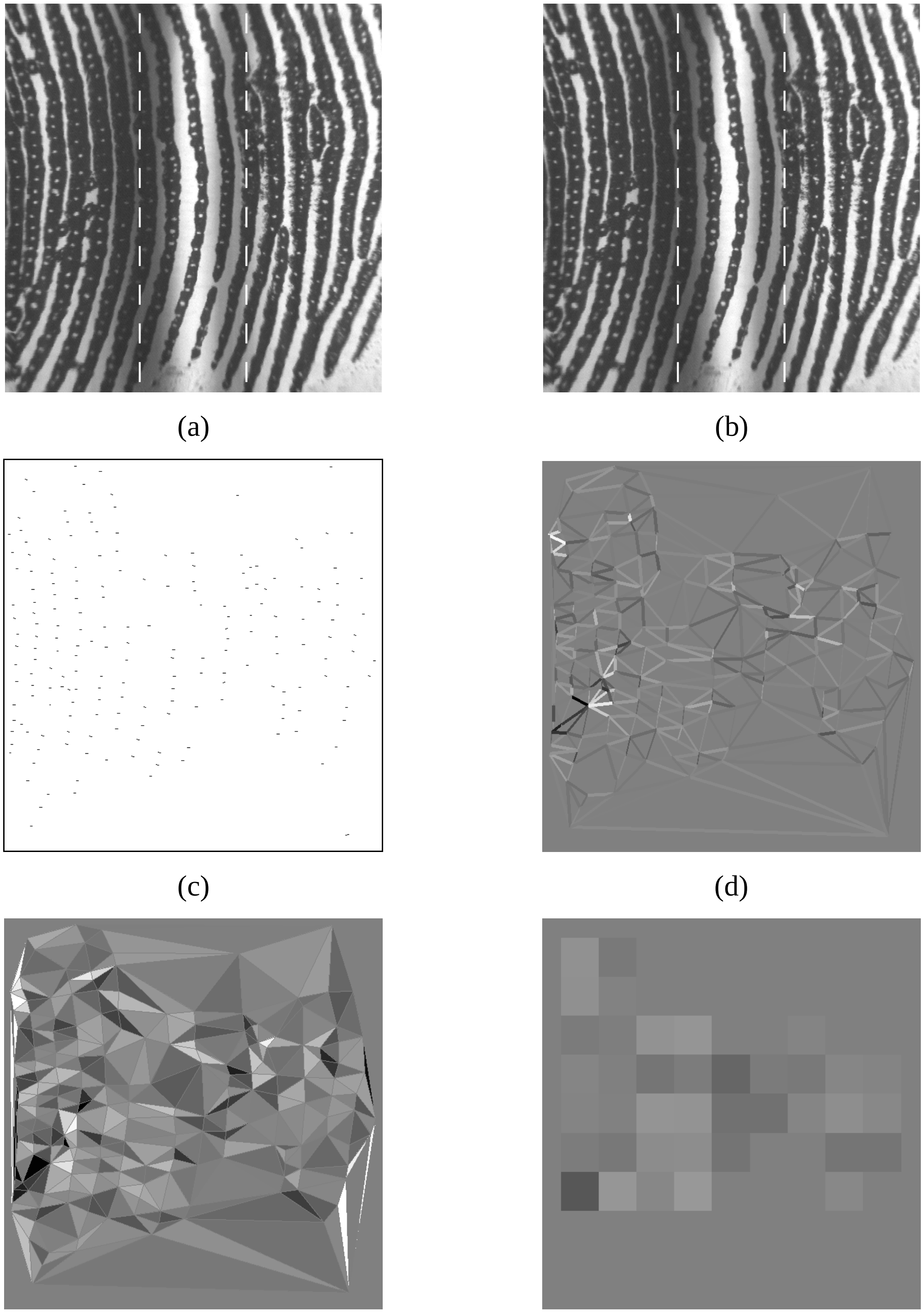


(a) (b)

(c) (d)

(e) (f)

Figure 81. Measurements of edge length change in ridgeL.avi (white/black = +/-15%): (a) frames 25-36, (b) frames 50-61, (c) frames 75-86, (d) frames 100-111, (e) frames 125-136, and (f) frames 150-161.



(a)

(b)

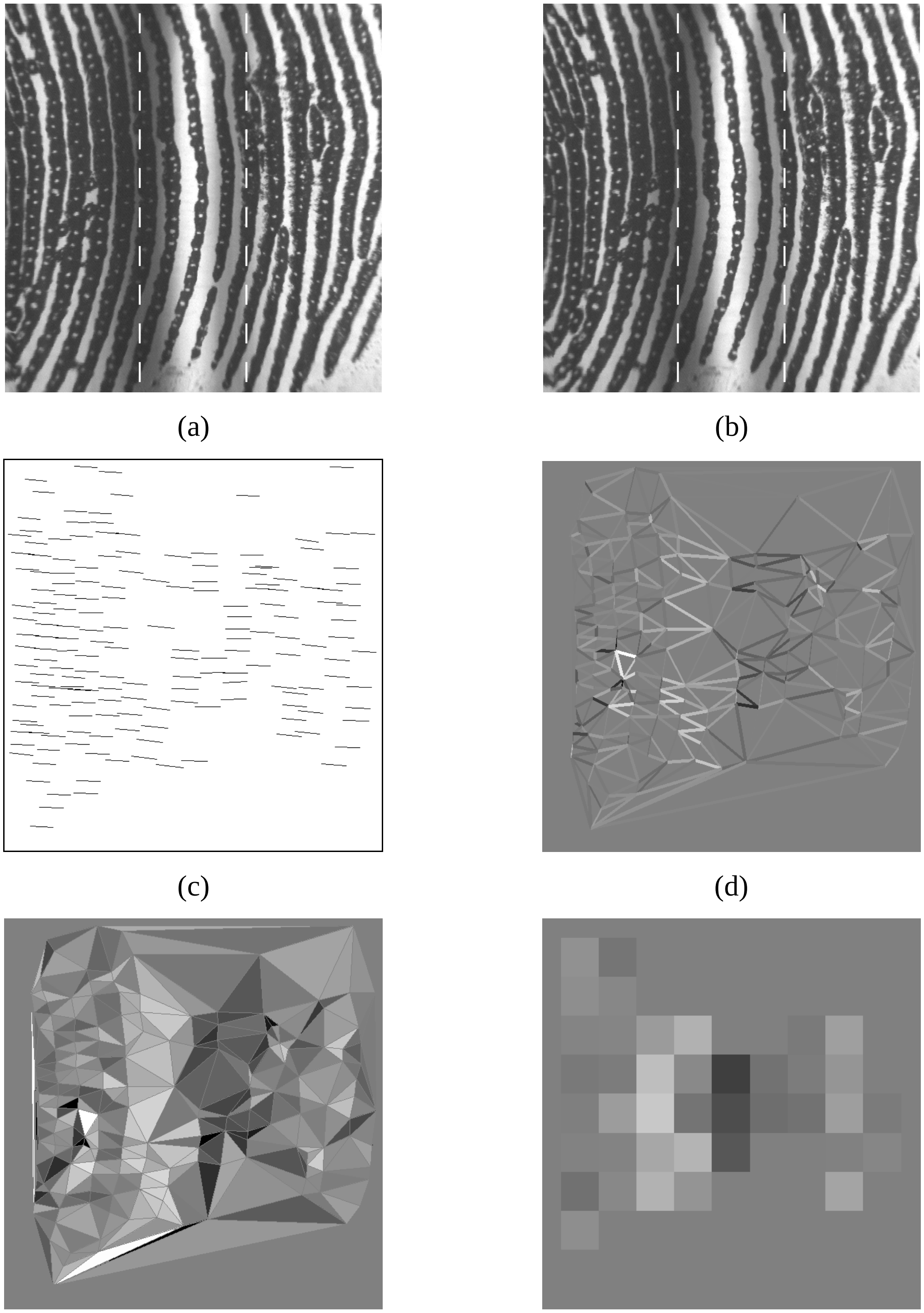
(c)

(d)

(e)

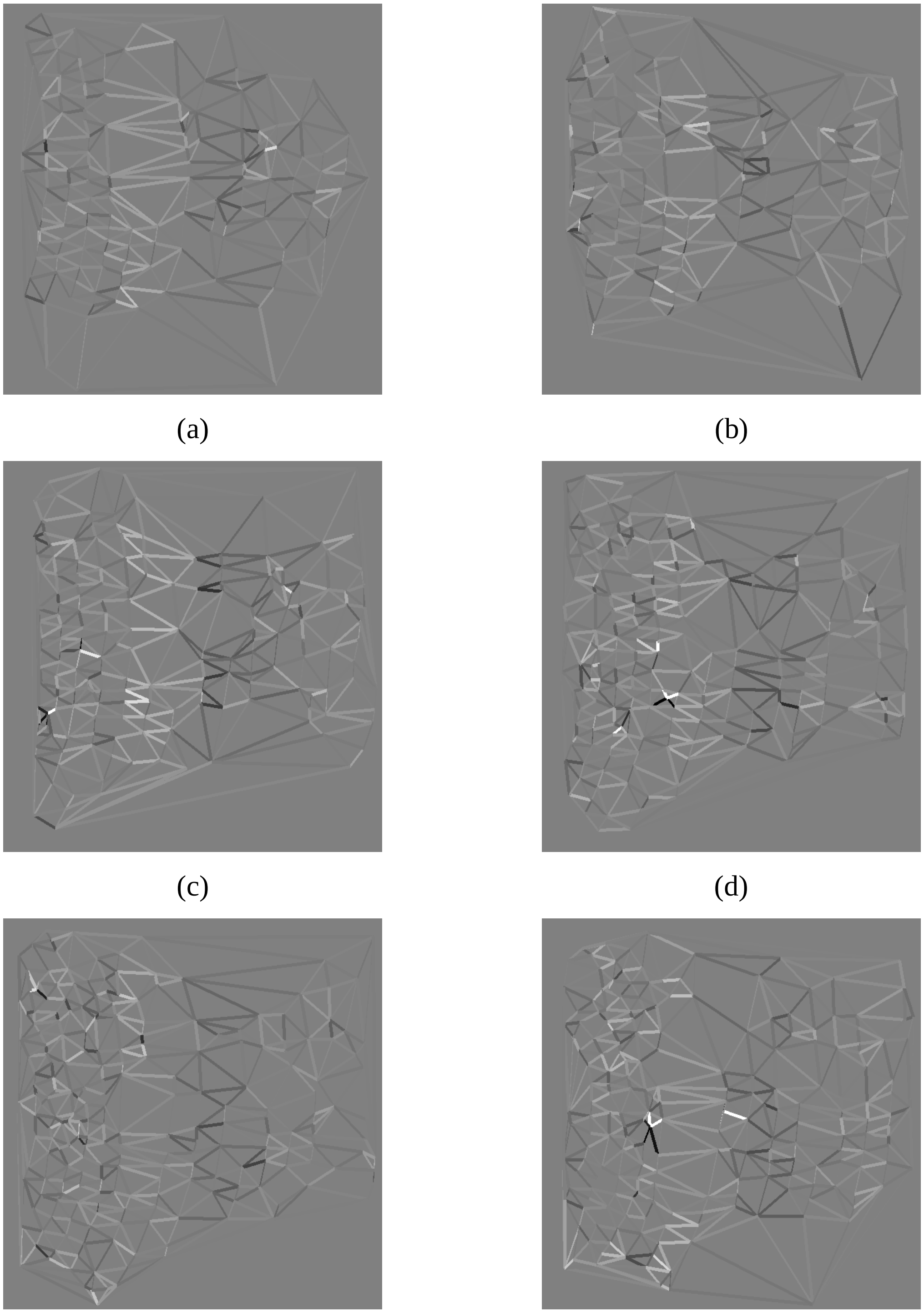
(f)

Figure 82. Measurements from frame 73 to 74 of valleyL.avi (white/black = +/-5%): (a) frame 73, (b) frame 74, (c) feature movement, (d) edge length change, (e) triangle area change, and (f) skin patch area change.



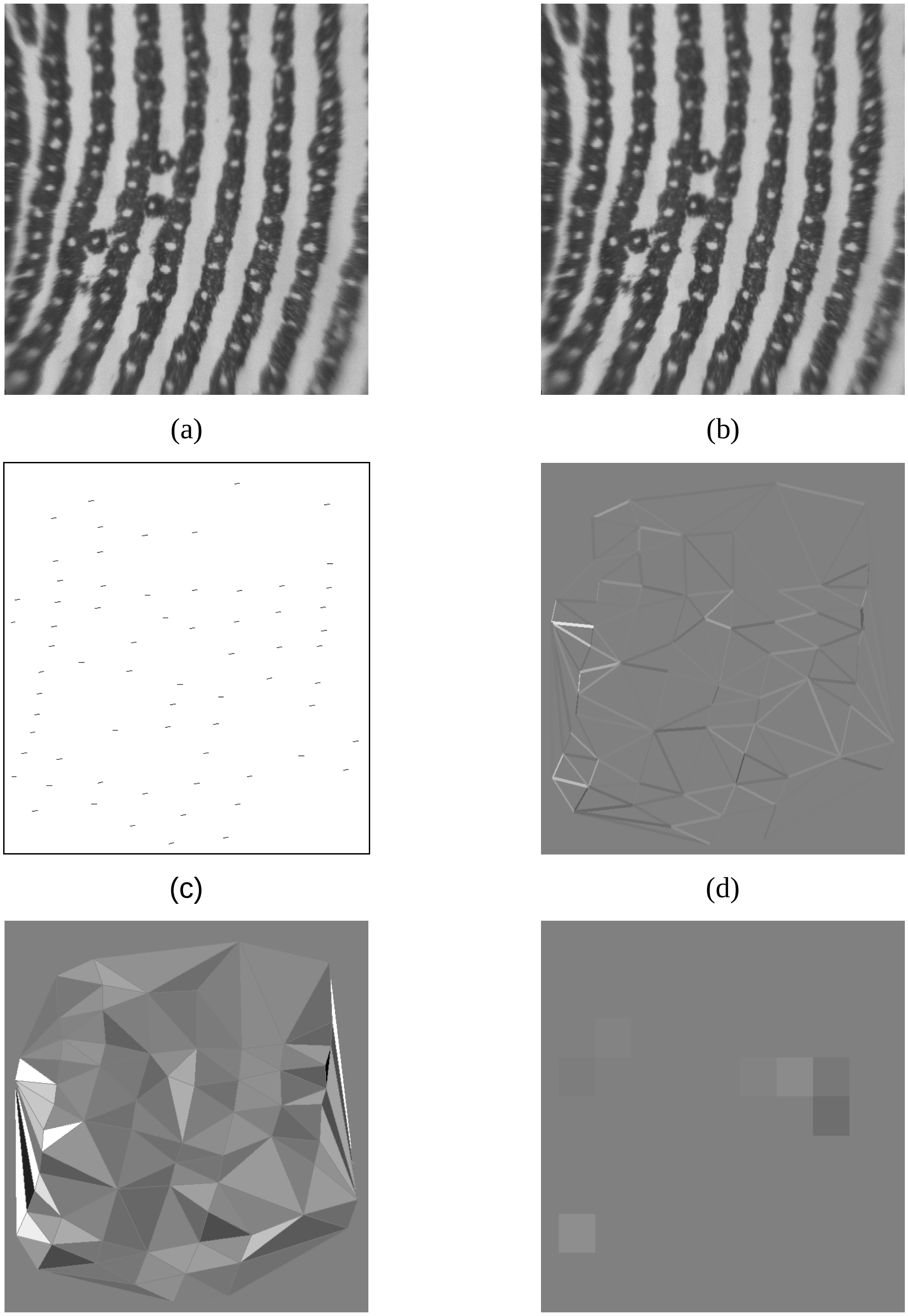
(a) (b) (c) (d) (e) (f)

Figure 83. Measurements from frame 73 to 84 of valleyL.avi (white/black = +/-15%): (a) frame 73, (b) frame 84, (c) feature movement, (d) edge length change, (e) triangle area change, and (f) skin patch area change.



(a) (b) (c) (d) (e) (f)

Figure 84. Measurements of edge length change in valleyL.avi (white/black = +/- 15%): (a) frames 25-36, (b) frames 50-61, (c) frames 75-86, (d) frames 100-111, (e) frames 125-136, and (f) frames 150-161.

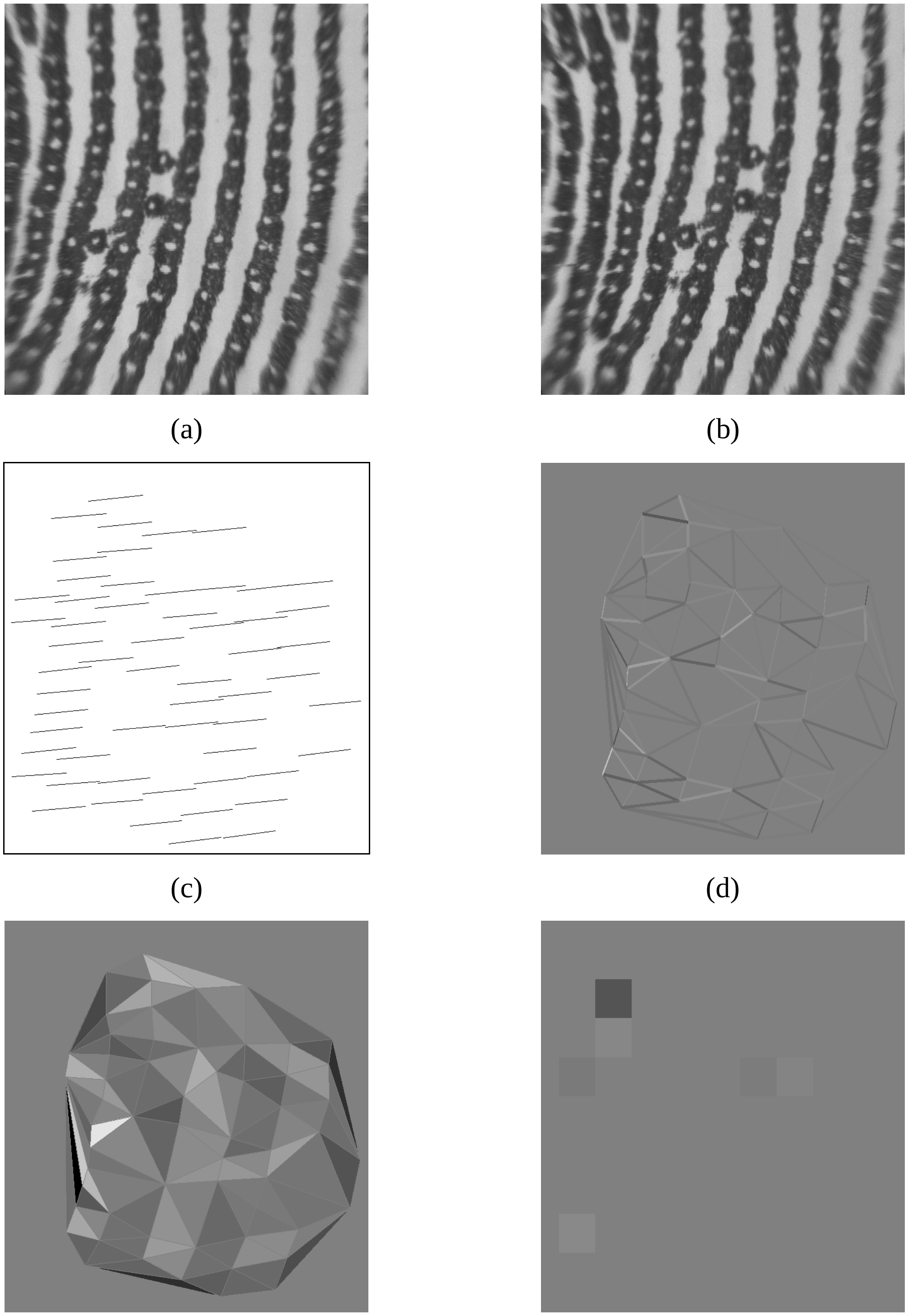


(a) (b)

(c) (d)

(e) (f)

Figure 85. Measurements from frame 47 to 48 of flatS.avi (white/black = +/-5%): (a) frame 47, (b) frame 48, (c) feature movement, (d) edge length change, (e) triangle area change, and (f) skin patch area change.

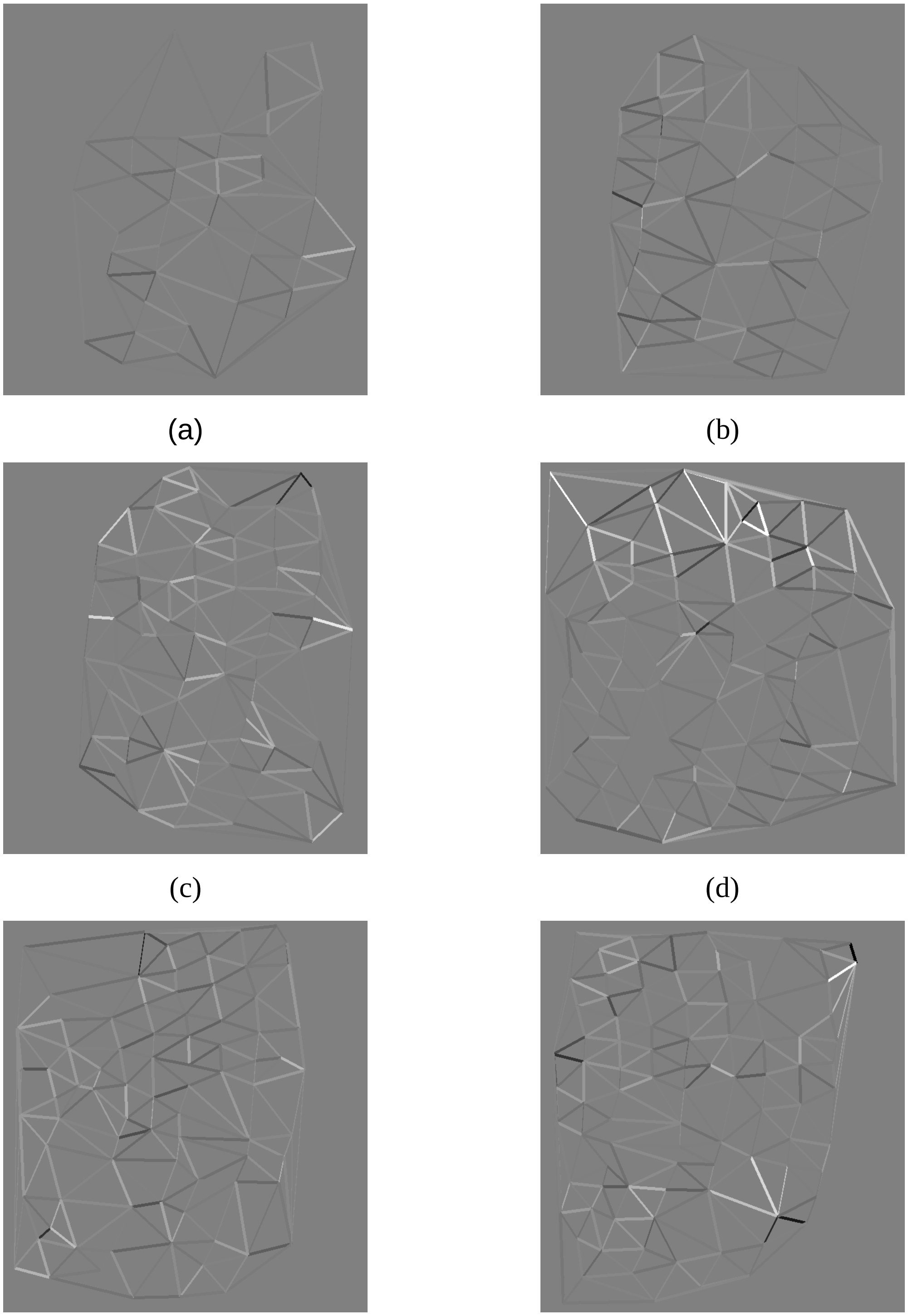


(a) (b)

(c) (d)

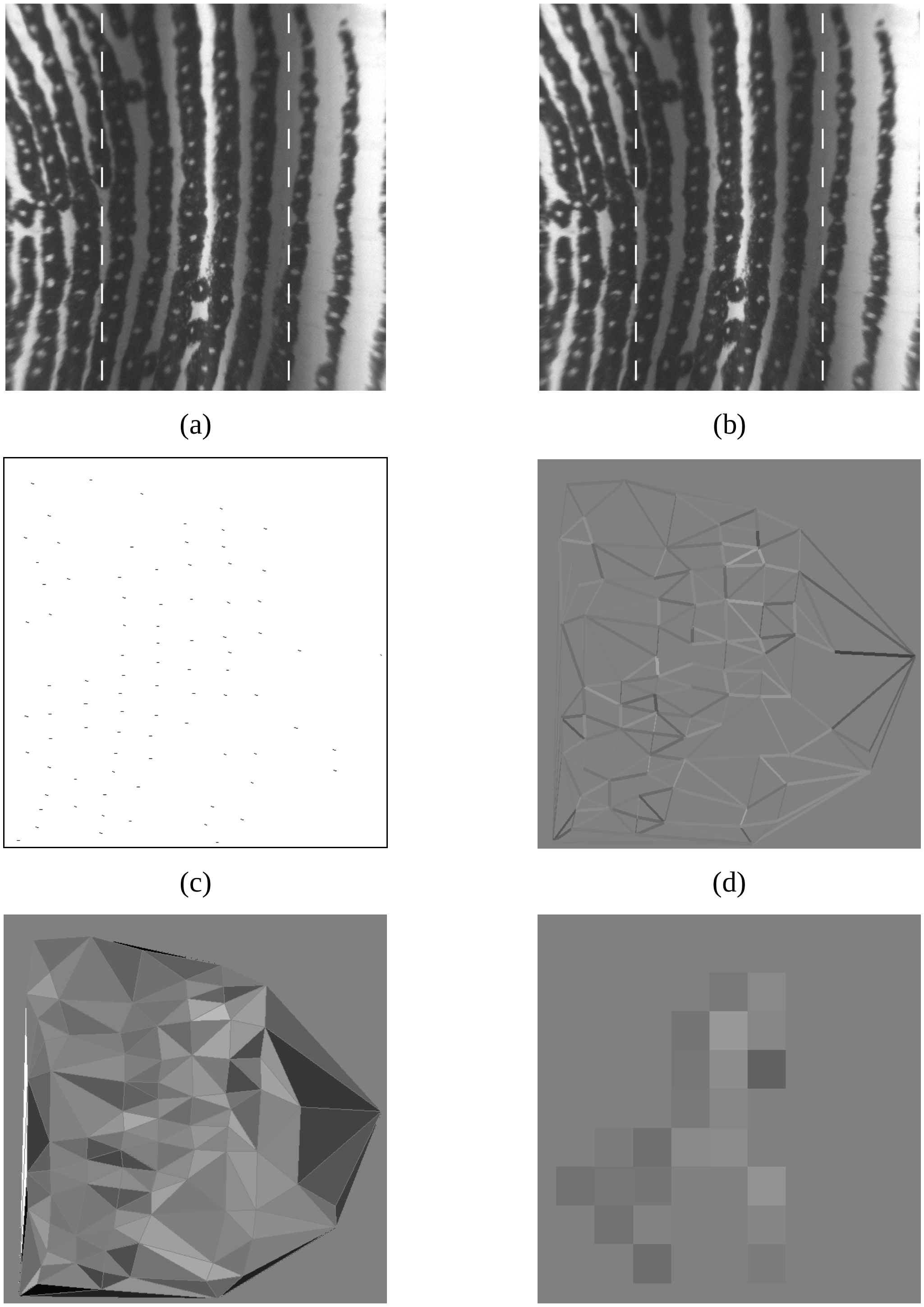
(e) (f)

Figure 86. Measurements from frame 47 to 58 of flatS.avi (white/black = +/-15%): (a) frame 47, (b) frame 58, (c) feature movement, (d) edge length change, (e) triangle area change, and (f) skin patch area change.



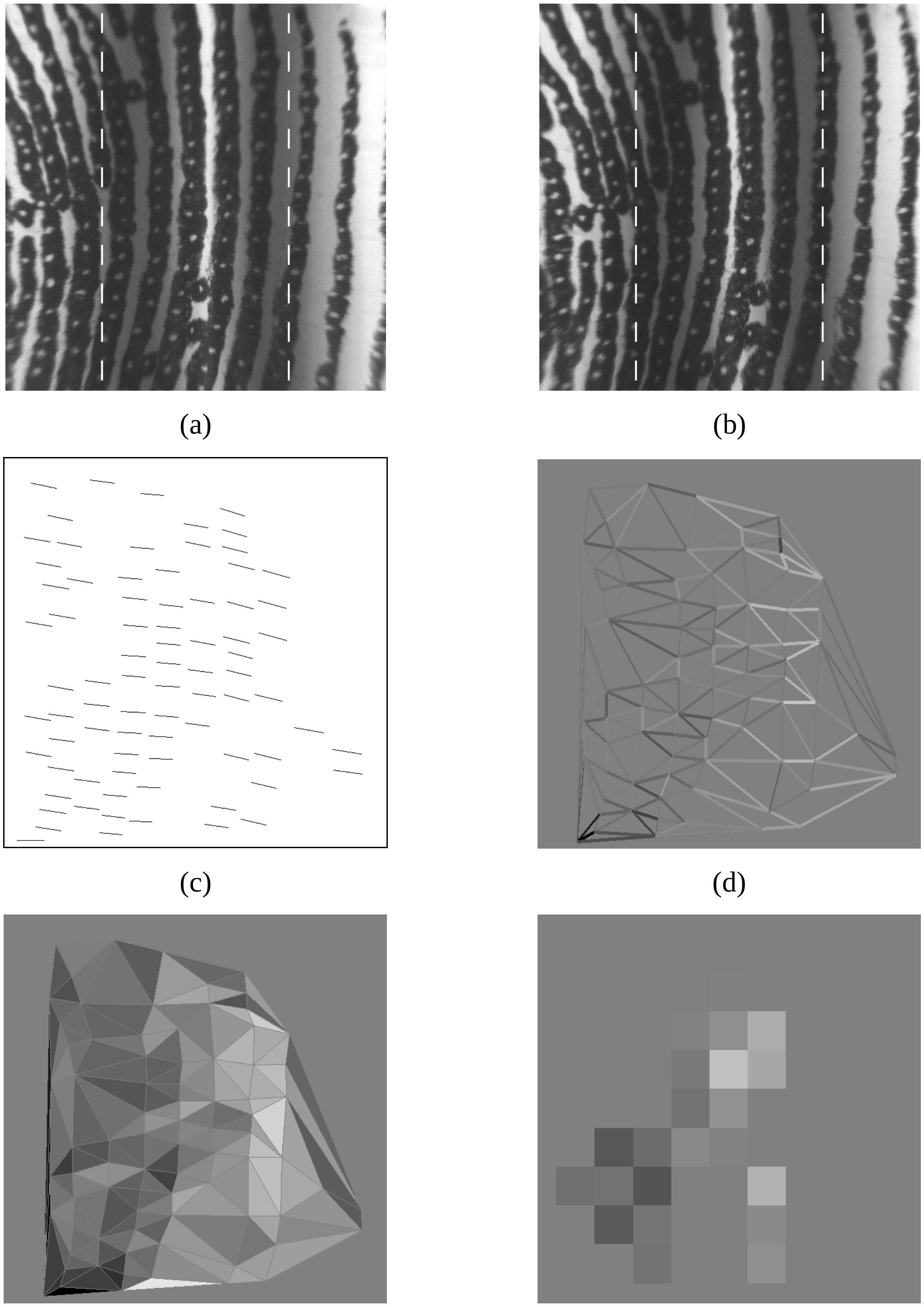
(a) (b) (c) (d) (e) (f)

Figure 87. Measurements of edge length change in flatS.avi (white/black = +/-15%): (a) frames 25-36, (b) frames 50-61, (c) frames 75-86, (d) frames 100-111, (e) frames 125-136, and (f) frames 150-161.



(a) (b) (c) (d) (e) (f)

Figure 88. Measurements from frame 95 to 96 of ridgeS.avi (white/black = +/-5%): (a) frame 95, (b) frame 96, (c) feature movement, (d) edge length change, (e) triangle area change, and (f) skin patch area change.

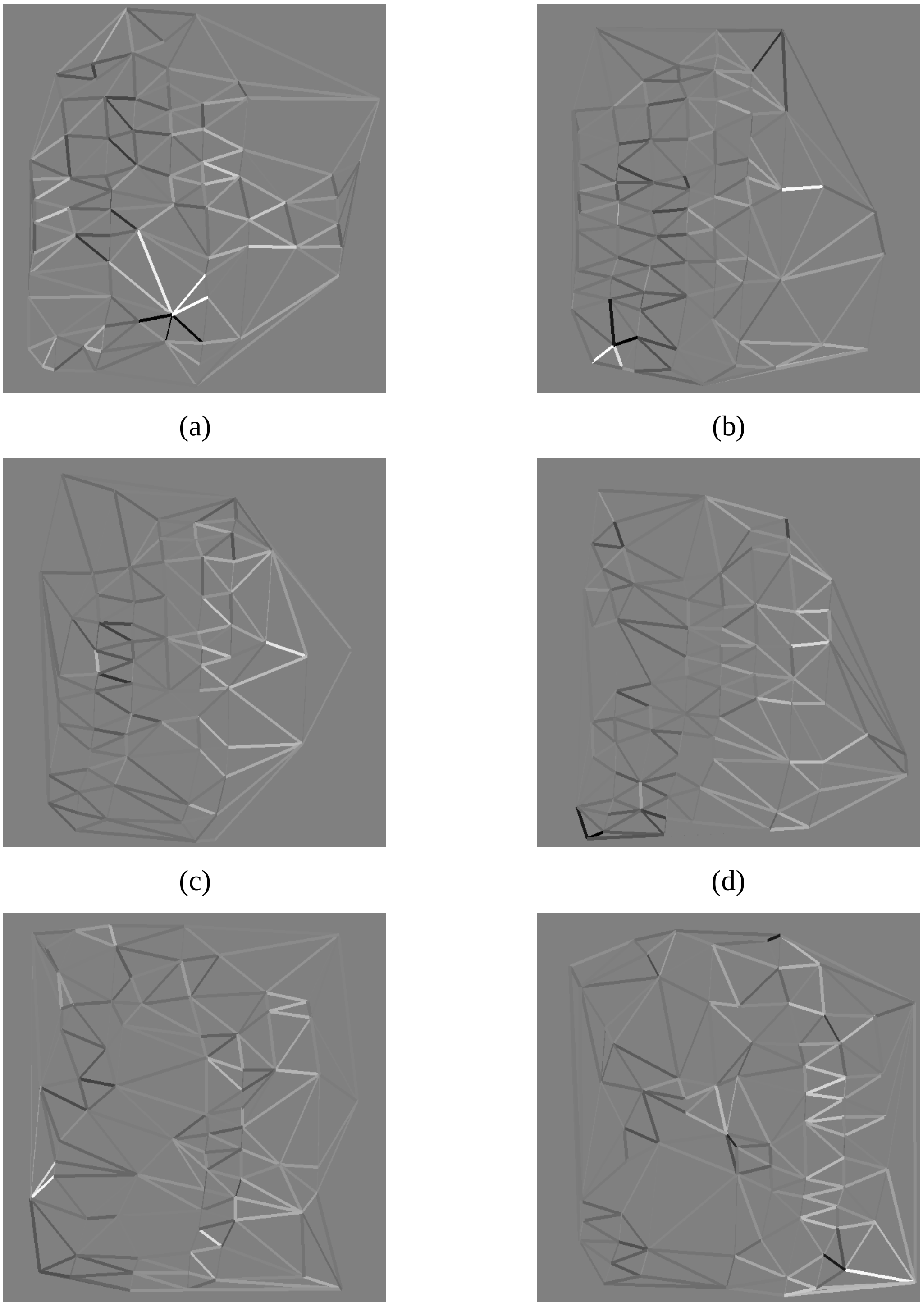


(a) (b)

(c) (d)

(e) (f)

Figure 89. Measurements from frame 95 to 106 of ridgeS.avi (white/black = +/- 15%): (a) frame 95, (b) frame 106, (c) feature movement, (d) edge length change, (e) triangle area change, and (f) skin patch area change.

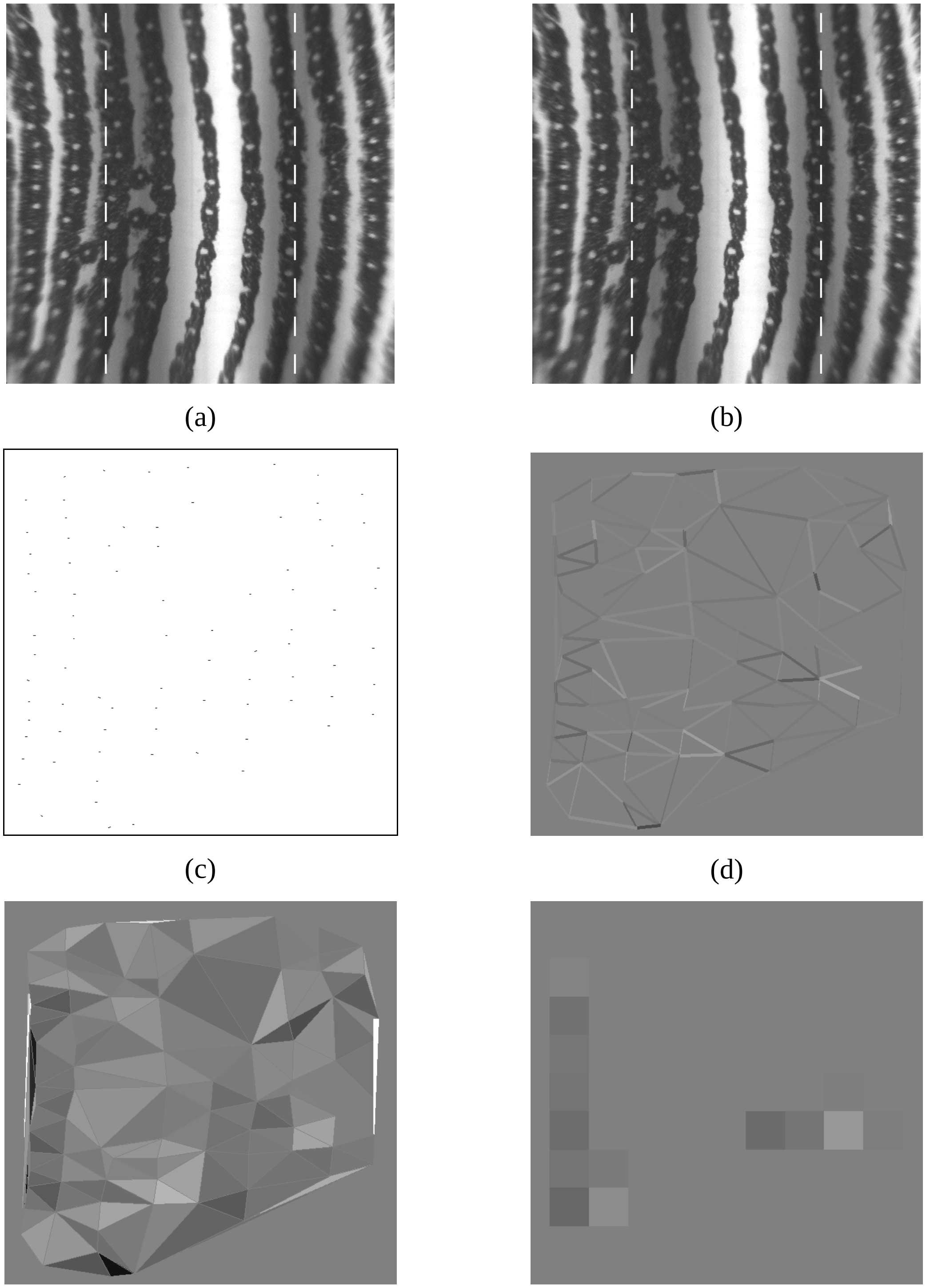


(a) (b)

(c) (d)

(e) (f)

Figure 90. Measurements of edge length change in ridgeS.avi (white/black = +/-15%): (a) frames 25-36, (b) frames 50-61, (c) frames 75-86, (d) frames 100-111, (e) frames 125-136, and (f) frames 150-161.



(a)

(b)

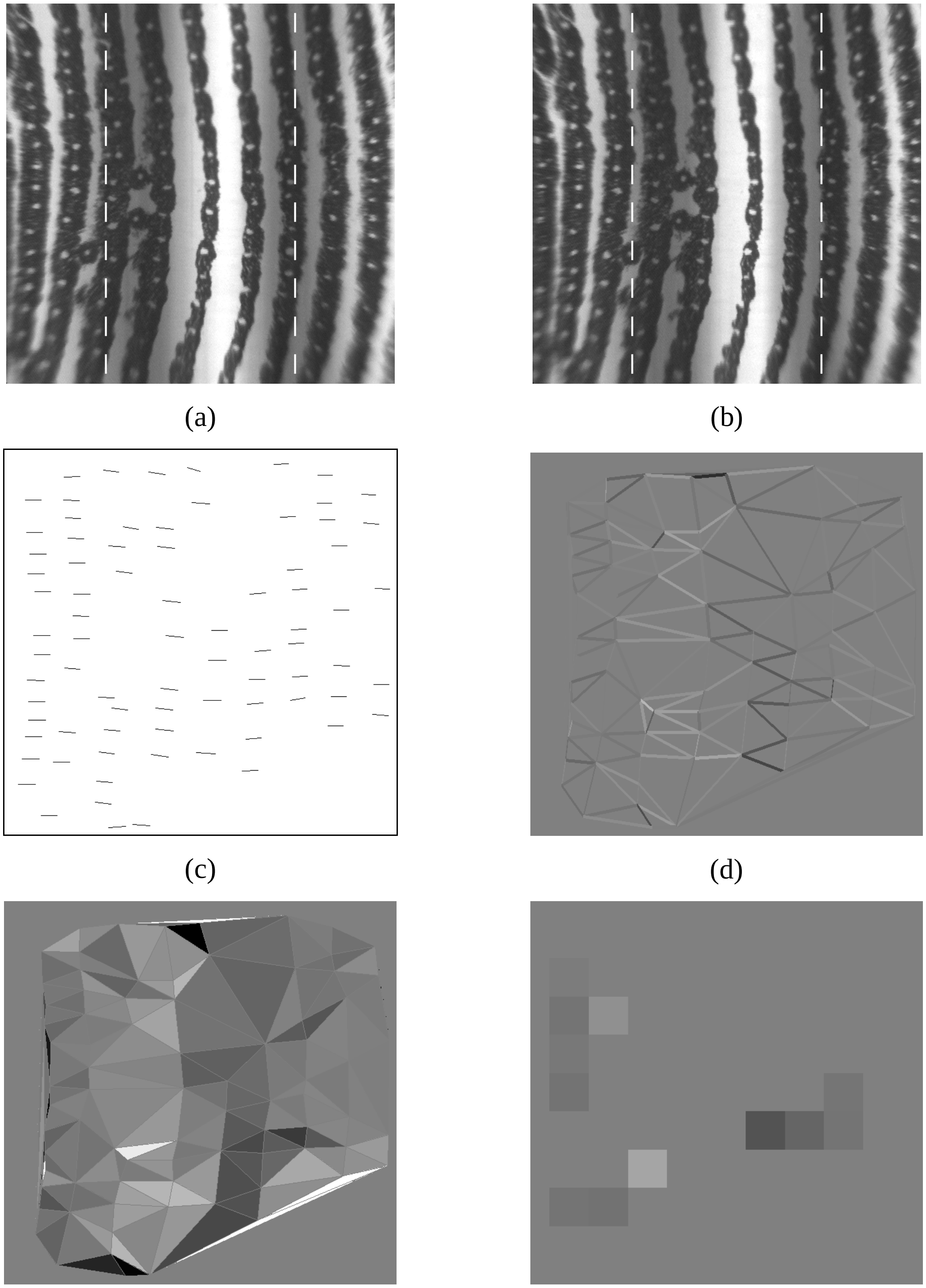
(c)

(d)

(e)

(f)

Figure 91. Measurements from frame 124 to 125 of valleyS.avi (white/black = +/-5%): (a) frame 124, (b) frame 125, (c) feature movement, (d) edge length change, (e) triangle area change, and (f) skin patch area change.



(a) (b)

(c) (d)

(e) (f)

Figure 92. Measurements from frame 124 to 135 of valleyS.avi (white/black = +/-15%): (a) frame 124, (b) frame 135, (c) feature movement, (d) edge length change, (e) triangle area change, and (f) skin patch area change.

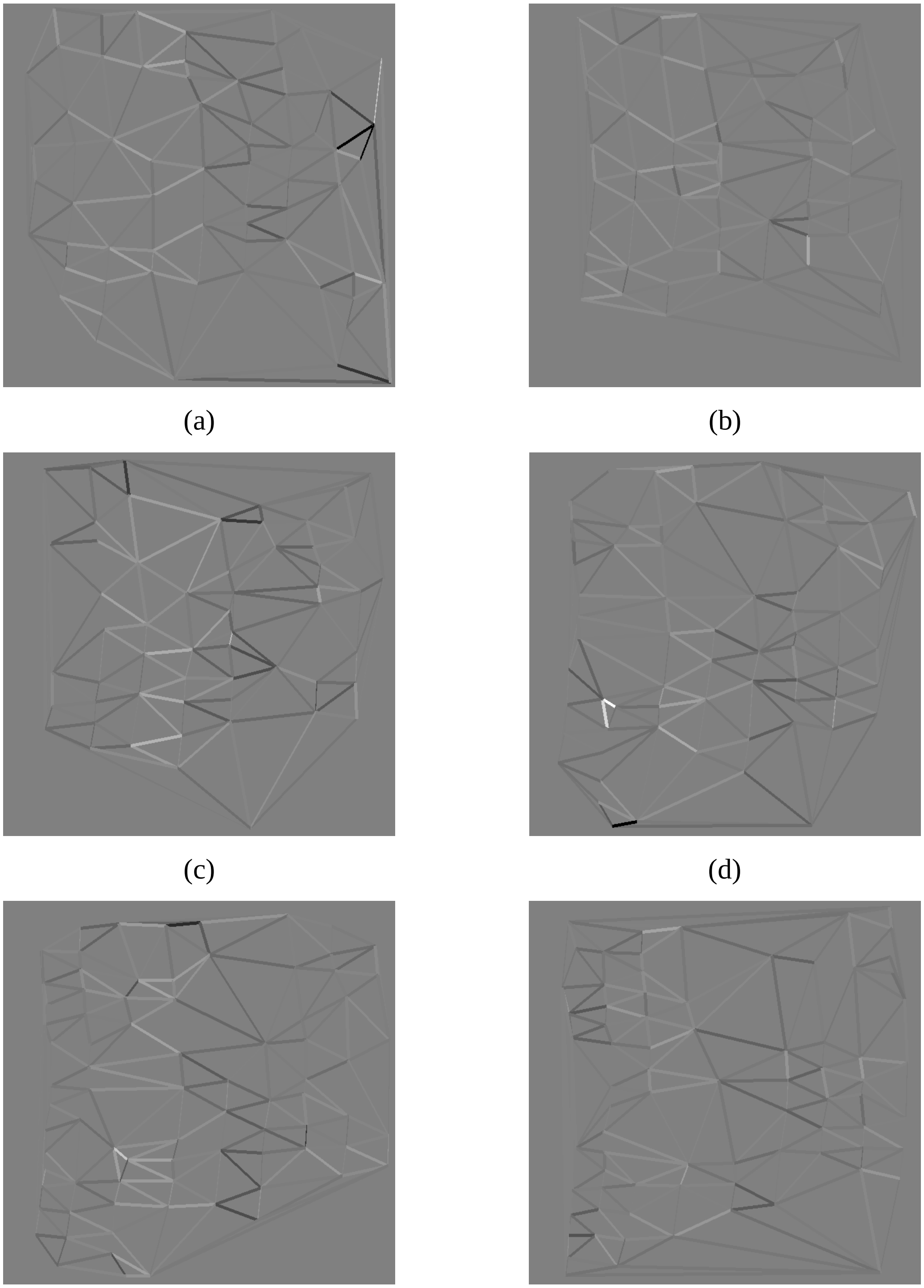
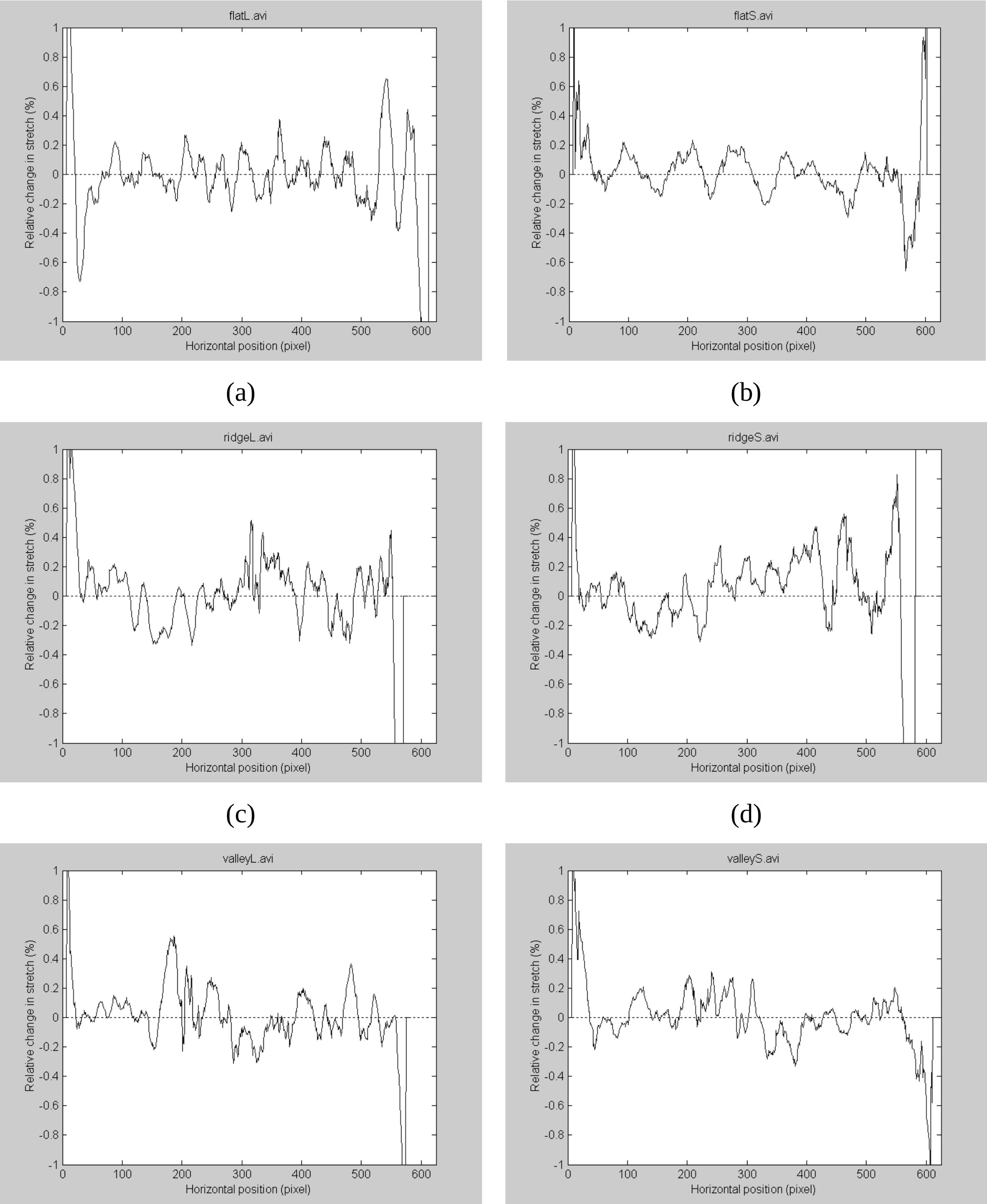

(a) (b) (c) (d) (e) (f)

Figure 93. Measurements of edge length change in valleyS.avi (white/black = +/- 15%): (a) frames 25-36, (b) frames 50-61, (c) frames 75-86, (d) frames 100-111, (e) frames 125-136, and (f) frames 150-161.



(a)

(b)

(c)

(d)

(e)

(f)

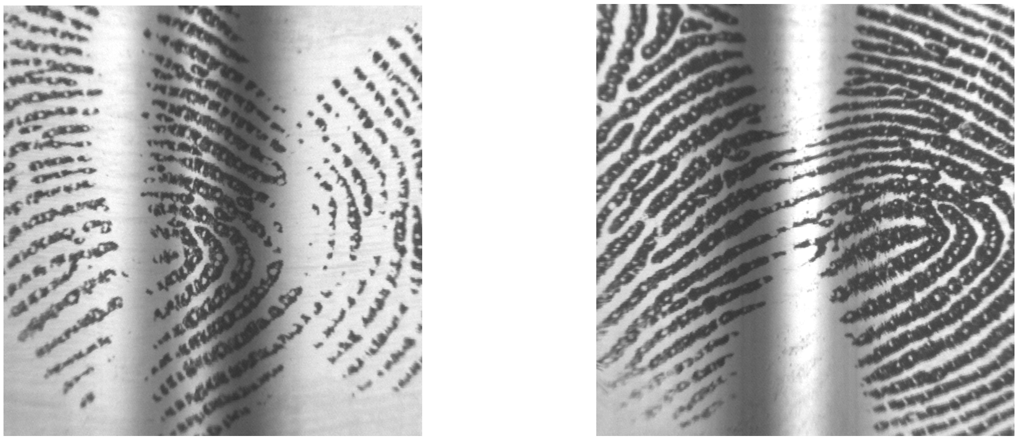
Figure 94. Relative change in stretch with respect to position averaged over sequences of 180 frames: (a) flatL.avi, (b) flatS.avi, (c) ridgeL.avi, (d) ridgeS.avi, (e) valleyL.avi, and (f) valleyS.avi.

6.3.3 Discussion

The results obtained in the previous section are very encouraging. The patterns of compression and expansion resulting from the presence of a ridge or valley make sense intuitively. These patterns, however, are barely visible through the rest of the deformations occurring at the fingertip. Measurements made with a flat surface show that significant deformations are present even in the absence of a shape. It is unclear at this point whether these measurements are representative of the actual deformations of the fingertip or due to measurement errors and noise. These deformations could be explained, for example, by the stick-slip of the fingertip ridges.

As mentioned in section 6.1, the accuracy of a feature's position is not known precisely. It is still interesting to consider the sensitivity of our measurements to errors in feature positions. Consider for example the triangulation shown in Figure 76(d)-(e). The average edge length of 48 pixels implies that an error of one pixel causes an error of approximately 2.1% in the relative length change. Similarly, consider an equilateral triangle with sides of length 48 pixels. As the length of one side increases by a pixel, the area of the triangle increases by approximately 1.4%. Considering the range of relative changes observed (approximately ±5%), it is clear that the signal-to-noise ratio may well be low. The improvement in pattern clarity when measuring through extended periods is also consistent with the presence of noise. Note however that the nature of this noise is not clear at this time. An increase in camera resolution could possibly reduce the noise significantly. A close-up on the fingertip, however, could not be shown to result in an obvious improvement. The noise may also be inherent to the contact-based imaging technique used.

It is also important to mention that the exploration conditions are not necessarily natural. Excessive pressure must generally be applied by the fingertip in order to maintain contact with the surface while sliding over a ridge or valley. The contact is particularly difficult to maintain while sliding over a valley. Figure 95 shows examples of images acquired with more natural contact pressure.



(a) (b)

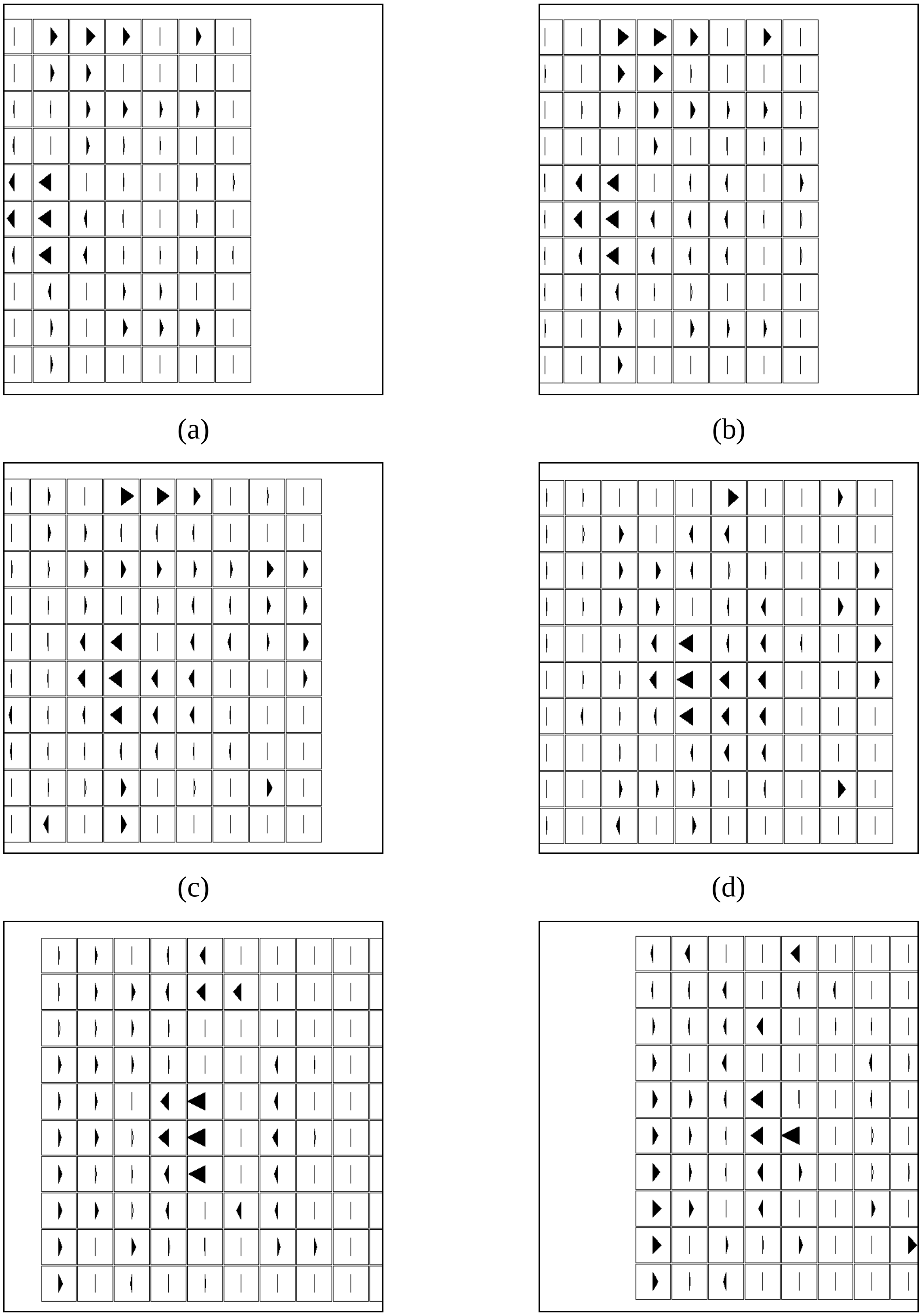
Figure 95. Loss of contact while sliding over a shape: (a) ridge and (b) valley.

Lastly, a comparison of the two stretch measurement methods reveals that the triangulation-based method is often easier to interpret. Edge length changes are generally also easier to interpret than area changes. They also permit the distinction between horizontal and vertical stretch. Changes in triangulation are also direct measurements that take into account only pairs or triplets of feature movements. Changes in skin patch area, on the other hand, are the result of many movement averaging operations. While the former method is likely to be more precise, the latter should be more resistant to noise. There is no clear winner at this point.

6.4 Tactile Display Mapping

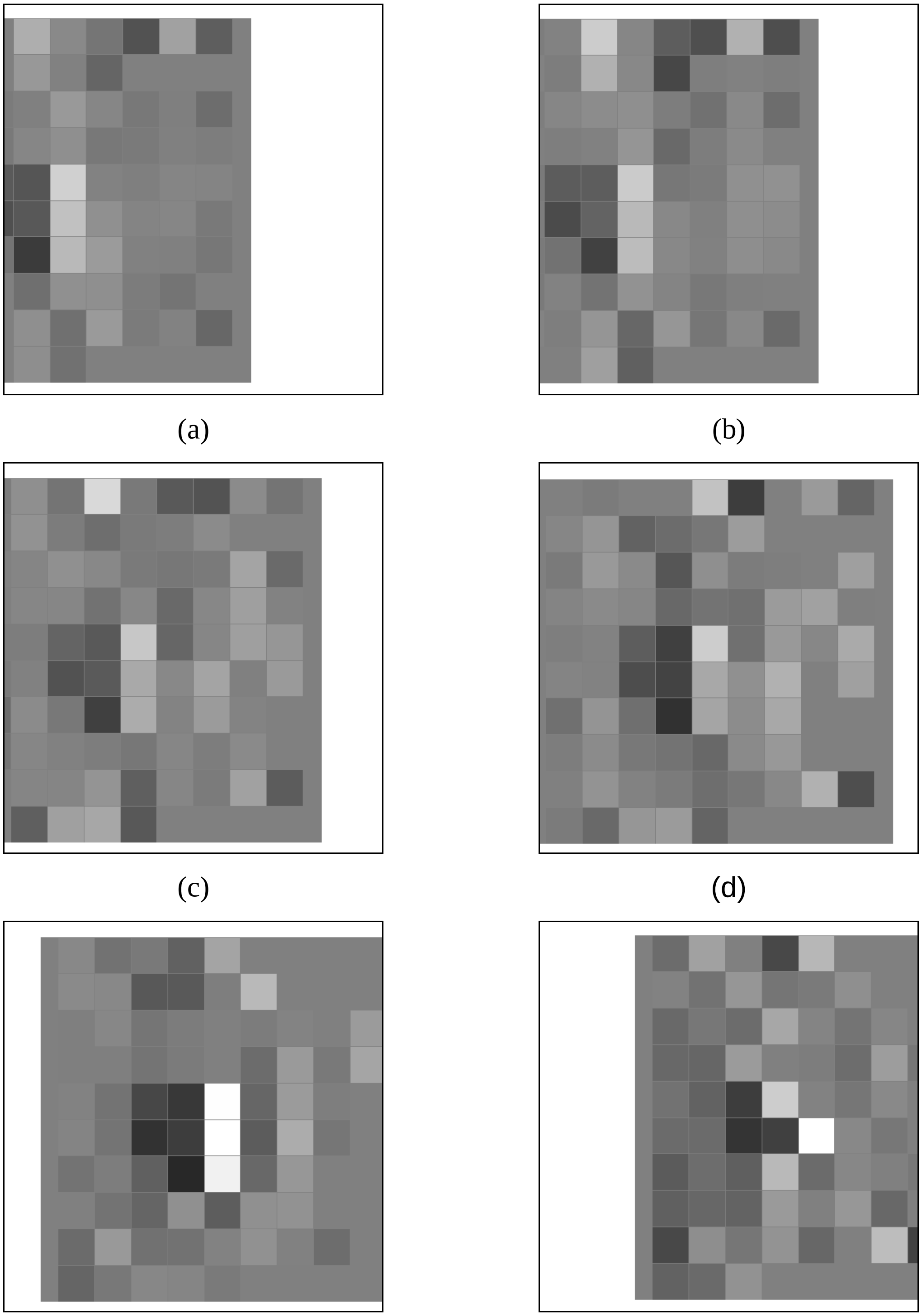
This section discusses the results of tests related to the generation of control signals for the STReSS tactile display. The mapping from skin stretch measurements to tactile display contactor positions is currently limited to the local motion-based method described in section 5.4. Figure 96 shows five fields of contactor positions generated from ridgeL.avi. Figure 97 shows corresponding fields of applied skin stretch (see section 5.4). The contactor position fields are difficult to interpret intuitively. On the other hand, the field of skin stretch should be expected to compress itself on the left of the ridge and to expand itself on the right. There is no obvious tendency towards this result in Figure 97. The tactile display mapping is likely to suffer from the accumulation of measurement noise. It also suffers from the low density of tracked features in some regions. Remember that if no feature is found in the vicinity of a contactor, its movement is unknown and its position is reset.

The latest prototype of the STReSS is not sufficiently accurate to be used with this signal generation method. A new prototype is currently being manufactured using more advanced techniques and should provide greater control over the movement of the skin contactors. Testing of this signal generation method is postponed until the new prototype is in service.



(a) (b) (c) (d) (e) (f)

Figure 96. Actuator positions for STReSS tactile display based on ridgeL.avi: (a) frame 25, (b) frame 50, (c) frame 75, (d) frame 100, (e) frame 125, and (f) frame 150.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 97. Stretch fields applied by the STReSS tactile display based on ridgeL.avi (white/black=+/- 50%): (a) frame 25, (b) frame 50, (c) frame 75, (d) frame 100, (e) frame 125, and (f) frame 150.

CHAPTER 7 CONCLUSION

This section concludes the thesis. Section 7.1 gives a brief summary of the skin stretch measurement method and of results presented in this thesis. Section 7.2 provides a list of potential improvements and ideas for future work. Section 7.3 describes the potential applications of this work. Section 7.4 closes the thesis with concluding remarks.

7.1 Summary

This thesis proposed a method for the measurement of skin stretch on the fingertip during tactile exploration tasks. The method consists of filming the fingertip as it moves across a transparent surface with simple geometric features. A prism-based image acquisition system improves the fingerprint contrast. The resulting sequence of images is analysed to extract and track fingerprint features. The tracked features are used to obtain dynamic information about skin stretch variations. The method is used to produce driving signals for a tactile display using distributed lateral skin stretch.

The method was tested with three surfaces: a flat surface, a surface with a ridge and a surface with a valley. Interesting measurements obtained from movement on a flat surface were presented. Significant deformation was shown to occur even in the absence of a shape. It was also shown that a fingerpad sliding over a ridge tends to compress itself before the shape and stretch itself after the shape. The reverse was observed in the case of a valley. The use of the method as a signal generator for the STReSS tactile display could not be tested satisfactorily.

7.2 Improvements and Future Work

The work presented in this thesis is a first attempt at measuring skin stretch from the tracking of anatomical features of the fingertip. Much work remains to be done and many improvements are possible. It is clear, for example, that the implementation of the various algorithms could be greatly improved. The analysis of a sequence of 180 frames currently takes a few minutes on a modern processor (approximately 2 minutes on an AMD Athlon XP 2000+ processor). More drastic changes in research orientation could also be considered. While this measurement method relies on the extraction of natural landmarks from the fingertip, other methods, such as an analysis of the optical flow, could provide better results. The remainder of this section discusses specific changes to the various subsystems of the current method that could improve its reliability and performance.

7.2.1 Fingertip Image Acquisition

The fingertip image acquisition system currently suffers from significant limitations due mostly to the contact-based image acquisition method. The contact surfaces are limited to parallel ruled surfaces and must have a low gradient in order to maintain frustrated total internal reflections. This prevents experimentation with more general shapes such as braille dots. The surface must also be made of transparent material. This complicates manufacturing and prevents the use of many perceptually interesting materials. The FTIR principle also fails when contact is lost between the fingertip and the surface, for example, when the finger moves across a high ridge. The deformation of the skin in the regions not seen is likely to be of significance perceptually. Moreover this limitation currently forces the use of excessive pressure when exploring a ridge or valley. Alternative imaging technologies were introduced in section 2.3.1 and evaluated in the current context in section 3.1. Although the feasibility of many of these technologies is doubtful, further investigation could uncover a more powerful imaging system.

A more powerful and regular illumination system would improve the depth of field and the contrast of images. It would also reduce the appearance of darker regions on the sides of ridges and valleys. An increase in spatial or temporal resolution of the camera would in turn improve the precision of the measurements without significant changes to the software.

The geometric calibration methodology employed in the current work could easily be improved. The means at our immediate disposal prevent the printout of a grid with a dot spacing lower than 0.5 mm. This spacing results in a rather sparse coverage of the ridge and valley on contact surfaces. Moreover, the precision of the grid is limited by the quality of the laser printer used. Applying the grid on a ruled surface is also a significant challenge. The rigidity of the grid prevents it from naturally following the contour of the ruled surface. The grid was forced into position using either a negative of the shape (i.e. pressing a ridge against a valley) or using pliable material such as play dough. Applying the grid correctly is an art, as bubbles tend to form in the oil film between the grid and the contact surface. The film of oil can also be sufficiently thick to cause errors, by filling valleys for example. A possible improvement consists of taking multiple pictures of a moving grid. A model of the surface and of the image acquisition system could then be used in a minimization strategy to determine the orientation of the surface plane and the position of the shape. This calibration method would also have the added benefit of precisely locating the shape in the image and thus allow correlation of measurements to its position.

The lighting compensation could also be improved by acquiring an image of the surface before finger contact. The current illumination map, corresponding to the maximum intensity for each pixel over a movie, occasionally shows irregular lighting due to the fact that some parts of the surface are always occluded by the fingerprint. Reflections at valleys tend to be slightly darker than reflections on free surfaces.

7.2.2 Feature Extraction

The feature extraction algorithms used in this work derive mostly from the work of Stosz and colleagues with a few enhancements and modifications [30][33]. The need to extract pore information precludes the use of numerous powerful fingerprint enhancement algorithms, which typically consider pores as noise to be eliminated. The extraction of minutiae could be improved by using these powerful algorithms after detecting the pores with the blob-based method. This approach would probably be too computationally expensive for an online fingerprinting system but is acceptable for our application. Maio and Maltoni provide an overview of interesting image enhancement and adaptive binarization techniques in [21].

Numerous improvements of the existing algorithms are also possible. Most of the processing is performed in terms of pixel position. Sub-pixel precision could easily be obtained by improving the implementation of the various algorithms. It may also be possible to improve the feature extraction performance by using more computationally expensive algorithms. Most of the recent work on fingerprint matching has focused on providing acceptable reliability at the lowest cost for practicality. Straying from the accepted best practices in the field could lead to improvements for the current application. It is clear, for example, that a fingerprint matching system must recover the true fingerprint of a person in order to match it to another one acquired in different conditions. It isn't clear however that fingerprint imperfections are bad for our purposes. A false valley ending that is stable through a sequence of images is a valid tracking marker. A careful study of the stability of false minutiae and other artefacts would be necessary in order to decide if they should be discarded or maintained.

7.2.3 Feature Tracking

Feature tracking currently assumes that the inter-frame displacement of features is much shorter than the inter-feature distance. It also assumes that feature movement is regular. These assumptions are often false in practice, especially at locations where interesting, abrupt deformations occur. While valley endings and bifurcations tend to be lost in case of error, pores can easily be matched incorrectly due to their lack of distinguishing features. A degenerate case was presented in section 6.2. Many of these incorrect matches are eliminated when filtering short trajectories but the feature density is still reduced.

Matching could be greatly improved by using more advanced point-pattern or graph matching techniques. The fingerprint matching literature contains numerous examples of the use of the local structure to match fingerprint features. Although these technologies are meant for the matching of a small number of features, they could be adapted for this application. The number of valleys that a line drawn between two features crosses should, for example, remain the same from one image to the next. Such improvements could potentially reduce the noise in skin stretch measurements and allow the tracking to function properly when the finger changes speed or direction abruptly.

As explained previously, the tracking algorithm also makes no attempt to register and track points through the entire sequence of images. Some simple predictive algorithms could possibly keep track of features even if they disappear for a few frames.

7.2.4 Skin Stretch Measurement

As discussed in chapter 6, the accuracy of measurements is not well understood and should be studied more carefully. The differences between the various skin stretch measurement techniques proposed should also be investigated. There is also no mapping, at the moment, from triangulation changes to tactile display contactor positions. The use of local motion for contactor positioning was tested with inconclusive results and deserves more attention. The problem of gaps in the local motion of skin patches and their effect on contactor positioning also has yet to be solved satisfactorily. The method should be tested directly on the physical tactile display once it is in operation. It may also be possible to use different measurement methods such as the perimeter of triangulation faces. Averaging measurements over a large number of image sequences could also reduce the noise and expose the effect of shapes on the deformation of the fingertip more clearly.

7.3 Applications

The target application of this work is the STReSS tactile display. As explained previously, determining the dynamic lateral stress field to apply to the fingertip for specific tactile sensations is not trivial. This work proposes a method for the observation and measurement of skin stretch during the exploration of perceptually meaningful surfaces. The end result is a sequence of skin contactor positions, or tactile movie, that can be used to control the STReSS.

More generally, this work aims at providing the instrumentation necessary to study deformations of the fingertip. A better understanding of the fingertip mechanics is an essential step in the understanding of the sense of touch. This instrumentation could potentially be used to derive a model of the fingertip or to verify existing models.

Skin deformation is also a significant issue in online fingerprinting applications. Cappelli, Maio and Maltoni recently published a model of fingertip deformations occurring during fingerprint acquisition in the hopes of improving the accuracy of matching algorithms and artificially extending fingerprint databases for algorithm testing [8]. The current work could potentially be used to verify and/or improve their model.

7.4 Concluding remarks

The past century has seen great leaps in our understanding of vision, leading to the development of technologies with a profound impact on our everyday life. Ubiquitous visual equipment, such as television sets and computer monitors, have been key at improving human communication and human-computer interaction. Similarly, sound recording and reproduction technology has been available for decades. Despite research efforts to improve our understanding of human touch, much about this sense remains to be discovered and exploited properly. It is hoped that the work presented in this thesis will play a small role in increasing our knowledge of this underestimated sense and in bringing tactile display technology to a practical reality.

REFERENCES

[1] R. Adhami and P. Meenen, “Fingerprinting for Security”, *IEEE Potentials*, vol. 20, no. 3, pp. 33-38, 2001.

[2] Atmel, “FCD4B14 FingerChip”, Datasheet, http://www.atmel.com/atmel/acrobat/doc1962.pdf

[3] R. D. Bahuguna and T. Corboline, “Prism fingerprint sensor that uses a holographic optical element”, *Applied Optics*, vol. 35, no. 26, pp 5242-5245, 1996.

[4] G. Bebis, T. Deaconu and M. Georgiopoulos, “Fingerprint Identification Using Delaunay Triangulation”, IEEE International Conference on Intelligence, Information and Systems (ICIIS'99), pp. 452- 459, Maryland, 1999.

[5] J.-A. Beraldin, M. Rioux, F.R. Livingstone, L. King, “Development of a real-time laser scanning system for object recognition, inspection, and robot control,” *SPIE Proceedings*, Telemanipulator Technology and Space Telerobotics. vol. 2057, pp. 451-461, 1993.

[6] W. Bicz, D. Banasiak, P. Bruciak, Z. Gumienny, S. Gumuliński, D. Kosz, A. Krysiak, W. Kuczyński, M. Pluta and G. Rabiej, “Fingerprint structure imaging based on an ultrasound camera”, NDTnet, vol. 3, no. 5, http://www.ndt.net/article/0598/optel/optel.htm, 1998.

[7] Biometric Partners Inc., http://www.biometricpartners.com/Touchless/touchless.html

[8] R. Cappelli, D. Maio and D. Maltoni, "Modelling Plastic Distortion in Fingerprint Images" in proceedings Second International Conference on Advances in Pattern Recognition (ICAPR2001), Rio de Janeiro, pp.369-376, March 2001.

[9] K. Dandekar and M. A. Srinivasan, Role of Mechanics in Tactile Sensing of Shape, Touch Lab Report 2, RLE TR-604, MIT, Cambridge, 1997.

[10] C. Desmarais, Skin Stretch Tactile Display, Honours Thesis, Dept. of Mechanical Eng., McGill University, 1997.

[11] C. Dorai, N. Ratha and R. Bolle, “Reliable Distortion Detection in Compressed Fingerprint Videos” in Proc. CVPR-2000, Hilton Head SC, 2000.

[12] Ethentica, “Ethentica and TactileSense™: A Breaktrhough in Fingerprint Authentication”, White Paper, http://www.ethentica.com/tactwhtpr.pdf.

[13] I. Fujieda and H. Haga, “Fingerprint input based on scattered-light detection”, *Applied Optics*, vol. 36, no. 35, 1997.

[14] E.B. Goldstein, “The Cutaneous Senses” in Sensation & Perception, Brooks/Cole Publishing Company, Pacific Grove, CA, 1999.

[15] U. Halici, L.C. Jain and A. Erol “Introduction to Fingerprint Recognition” in Intelligent Biometric Techniques in Fingerprint and Face Recognition, L.C. Jain, U. Halici, I. Hayashi, S.B. Lee, Editors, CRC Press, Boca Raton, 1999.

[16] V. Hayward and M. Cruz-Hernandez, “Tactile Display Device Using Distributed Lateral Skin Stretch,” in Proc. ASME, vol. DSC-69-2, 2000, pp. 1309-1314.

[17] A. Jain, L. Hong, R. Bolle, On-Line Fingerprint Verification, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 4, April 1997, pp 302-314.

[18] A. Jain and S. Pankanti, “Automated Fingerprint Identification and Imaging Systems,” in *Advances in Fingerprint Technology,* 2nd Ed., H. C. Lee and R. E. Gaensslen, Eds., Elsevier Science, New York, 2001.

[19] D.A. Kontarinis, J.S. Son, W. Peine, R.D. Howe, “A Tactile Shape Sensing and Display System for Teleoperated Manipulation”, IEEE International Conference on Robotics and Automation, 1995.

[20] T. Maeno, K. Kobayashi, and N. Yamakazi, “Relationship between the Structure of Human Finger Tissue and the Location of Tactile Receptors”, Bulletin of JSME International Journal, Vol. 41, No. 1, C, 1998, pp. 94-100.

[21] D. Maio and D. Maltoni “Minutiae Extraction and Filtering from Gray-Scale Images” in Intelligent Biometric Techniques in Fingerprint and Face Recognition, L.C. Jain, U. Halici, I. Hayashi, S.B. Lee, Editors, CRC Press, Boca Raton, pp. 155-192, 1999.

[22] T. Maucher, K. Meier, J. Schemmel, “An Interactive Tactile Graphics Display”, Sixth International Symposium on Signal Processing and its Applications, ISSPA 2001.

[23] B.M. Mehtre and B. Chatterjee, “Segmentation of fingerprint images – a composite method”, Pattern Recognition, vol. 22, no. 4, pp. 381-385, 1989.

[24] G. Moy, C. Wagner, R.S. Fearing, “A Compliant Tactile Display for Teletaction”, IEEE Int. Conf. on Robotics and Automation, April 2000.

[25] J. Pasquero, McGill University, Dept. of Electrical and Computer Engineering, M. Eng. Thesis, in preparation.

[26] D.T.V. Pawluk and R.D. Howe, Dynamic contact of the human fingerpad against a flat surface, ASME Journal of BiomechanicalEngineering 121(6):605-611, December 1999.

[27] N.K. Ratha, R.M. Bolle, V.D. Pandit and V. Vaish, Robust Fingerprint Authentication Using Local Structural Similarity, Fifth IEEE Workshop on Applications of Computer Vision, pp. 29-34, 2000.

[28] G. X. Ritter, J. N. Wilson, Handbook of Computer Vision Algorithms in Image Algebra, Second Edition, CRC Press, 2000, pp. 163-168.

[29] A.R. Roddy and J.D. Stosz “Fingerprint Features – Statistical Analysis and System Performance Estimates,” in Proceedings of the IEEE, vol. 85, no. 9, pp. 1390-1421, 1997.

[30] A.R. Roddy and J.D. Stosz “Fingerprint Feature Processing Techniques and Poroscopy” in Intelligent Biometric Techniques in Fingerprint and Face Recognition, L.C. Jain, U. Halici, I. Hayashi, S.B. Lee, Editors, CRC Press, Boca Raton, 1999.

[31] M.A. Srinivasan, Surface deflection of primate fingertip under line load, J. Biomechanics, vol. 22, no. 4, pp. 343-349, 1989.

[32] M.A. Srinivasan and K. Dandekar, An investigation of the mechanics of tactile sense using two dimensional models of the primate fingertip, Journal of Biomechanical Engineering, vol. 118, pp. 48-55, 1996.

[33] J.D. Stosz and L.A. Alyea, Automated system for fingerprint authentication using pores and ridge structure, Proceedings of SPIE, Automatic Systems for the Identification and Inspection of Humans, San Diego, vol. 2277, pp. 210-223, 1994.

[34] Ultra-Scan Corporation, http://www.ultra-scan.com.

[35] C. R. Wagner, S.J. Lederman, R.D. Howe, “A Tactile Shape Display Using RC Servomotors”, Tenth symposium on haptic interfaces for virtual environment teleoperator systems, March 24-25, 2002, Orlando.

[36] E.W. Weisstein, “Frustrated Total Internal Reflection” in Eric Weisstein’s World of Physics, http://scienceworld.wolfram.com/physics/FrustratedTotalInternalReflection.html.

[37] E.W. Weisstein, “Variance” in Eric Weisstein’s World of Mathematics, http://mathworld.wolfram.com/Variance.html.

[38] X. Xia and L. O’Gorman, “Innovations in fingerprint capture devices”, *Pattern Recognition*, 2002, in press.

[39] ___, Biomechanics of Human Fingerpad-Object Contact, Touch Lab, MIT, http://touchlab.mit.edu/oldresearch/areas/biomechtouch/fingerpadobject.html.

[40] ___, “CGAL Manual – Basic Library”, Release 2.4, May 2002, p. 574, http://www.cgal.org.

APPENDIX A – CONTACT SURFACE CONSTRAINTS

Placing a diffuser on the entry face of the right-angle prism makes it possible to apply certain surfaces to its hypotenuse face without breaking the FTIR principles. The gradient of the surface must be constrained so that for all positions of interest there is a ray of light coming from the diffuser that is totally reflected and leaves the prism perpendicularly to the exit face. This appendix derives the constraints on the surface $S(x,y)$ using the notation shown in Figure 98. It is assumed that the entry and exit faces of the prism are square with sides of length D. This assumption holds for most right-angle prisms available off-the-shelf. It is also assumed that the prism has an index of refraction of 1.5 (approximately that of BK7 glass) and thus a critical angle of 41.8°.

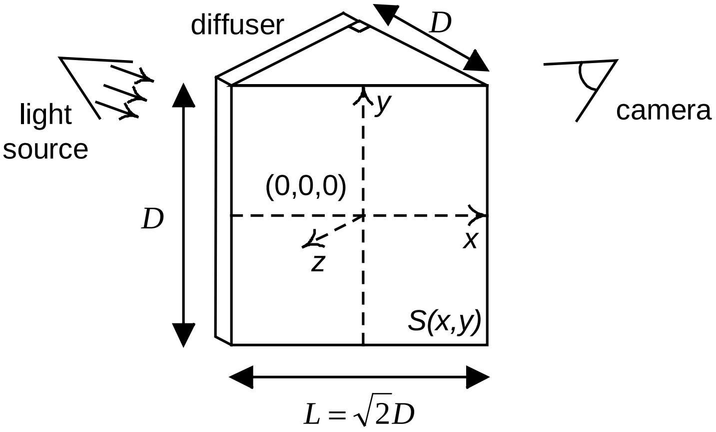


Figure 98. Notation for contact surface constraints.

The horizontal ($\nabla_x S$) and vertical ($\nabla_y S$) components of the surface gradient will be studied separately. In order to maintain the FTIR principle, either $\nabla_x S$ or $\nabla_y S$ must insure that it is possible to reflect a ray towards the camera. This condition is checked by verifying that rays normal to the exit face of the prism strike the contact surface at an angle (with respect to the surface normal) greater than the critical angle. As illustrated in Figure 99, a null vertical gradient implies that rays strike the surface perpendicularly and are not reflected unless the striking angle in the horizontal plane is greater than the critical angle. Since a $\nabla_y S$ near zero is hardly avoidable, the vertical gradient cannot insure reflection satisfactorily. The next section will show that $\nabla_x S$ is severely constrained by the need to insure reflection and should thus be set to zero. Constraints on $\nabla_y S$ will then be derived and analysed.

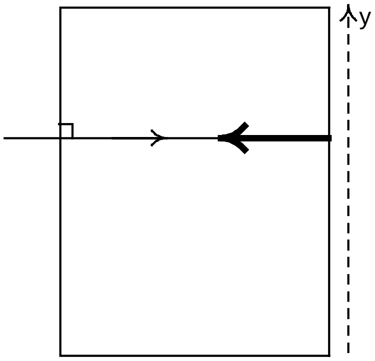


Figure 99. Relfection in Y-axis.

Horizontal Component of Gradient

The horizontal component of the surface gradient ($\nabla_x S$) must be constrained so that at all points at least one ray coming from the diffuser strikes the surface at an angle sufficient to cause reflection and leaves the exit face perpendicularly. The lower bound on $\nabla_x S$ is limited by the ray parallel to the hypotenuse face as shown in Figure 100. Equations (A.1)-(A.3) show that $\nabla_x S$ must be greater than –0.41.

(A.1) $\Theta_R = \dfrac{180° - 45°}{2} = 67.5°$

(A.2) $\Theta_S = 90° - \Theta_R = 22.5°$

(A.3) $\nabla_x S(x, y) > -\tan\Theta_S = -0.41$

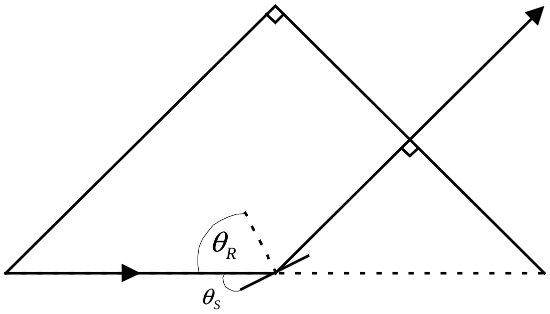


Figure 100. Minimum horizontal gradient.

The upper bound on $\nabla_x S$ is strongly dependent on the critical angle. Figure 101 shows the limit at which a light ray coming from the apex of the prism can no longer be reflected. Equation (A.4)-(A.7) show that limit to be $x = 0.4L$.

(A.4) $\theta = 180° - 2\theta_C - 45° = 51.4°$

(A.5) $a = L/2$

(A.6) $b = x$

(A.7) $\tan\theta = a/b \Rightarrow x = \dfrac{L}{2\tan\theta} = 0.4L$

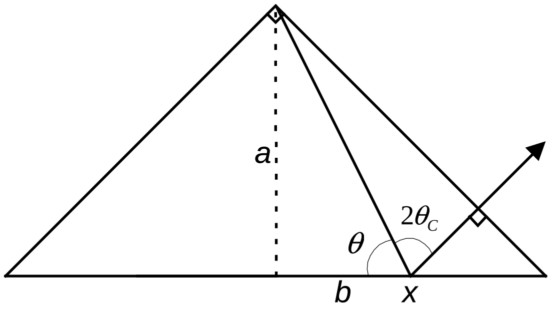


Figure 101. Limit on horizontal reflection.

Below the 0.4L limit, $\nabla_x S$ is limited by the critical angle as shown in Figure 102. Equations (A.8)-(A.9) show that the maximum $\nabla_x S$ is then 0.056.

(A.8) $\theta_S = 90° - 45° - \theta_C = 3.2°$

(A.9) $\nabla_x S(x,y)_{x<0.4L} < \tan\theta_S = 0.056$

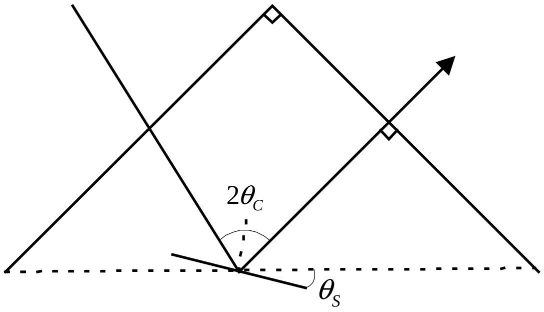


Figure 102. Maximum horizontal gradient (x<0.4L).

Beyond 0.4L, the maximum $\nabla_x S$ depends on the ray sent from the apex of the prism as shown in Figure 103. Equation (A.16) shows that the maximum $\nabla_x S$ decreases steadily until it reaches zero.

(A.10) $a = L/2$

(A.11) $b = x$

(A.12) $\theta = 135° - 2\theta_R$

(A.13) $\tan\theta = \tan(135° - 2\theta_R) = \frac{a}{b} = \frac{L}{2x}$

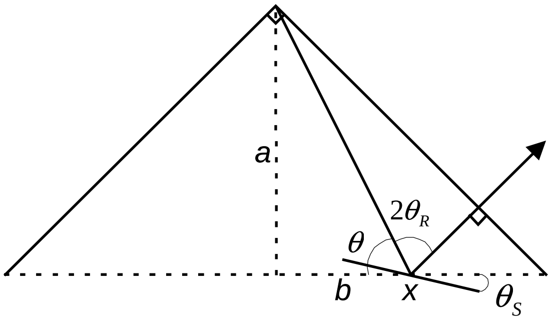


Figure 103. Maximum horizontal gradient (x>0.4L).

(A.14) $\theta_R = 67.5° - 0.5\arctan\frac{L}{2x}$

(A.15) $\theta_S = 45° - \theta_R = 0.5\arctan\frac{L}{2x} - 22.5°$

(A.16) $\nabla_x S(x,y)_{x>0.4L} < \tan\left(0.5\arctan\frac{L}{2x} - 22.5°\right)$

In summary, the need to insure reflection on the horizontal axis constrains the surface to a horizontal gradient varying between -0.41 and 0.056 over 90% of the surface. This constraint severely limits the use of a gradient in the horizontal axis. To insure proper reflection and to simplify vertical-axis constraint computations, it is thus recommended to maintain the surface at a zero horizontal gradient.

Vertical Component of Gradient

It has been established in the previous section that the horizontal component of the surface gradient should be set to zero in order to insure reflection. This results in a y-varying parallel ruled surface *S*(*y*). It follows that light strikes the hypotenuse face of the prism at a 45° angle as shown in Figure 104. The constraints on $\nabla_y S$ will now be estimated by studying the cross-section of width $w(x)$ given by Equation (A.17).

$$w(x)=\frac{x+0.5L}{\sqrt{2}}=\frac{2\sqrt{2}x+2D}{4} \tag{A.17}$$

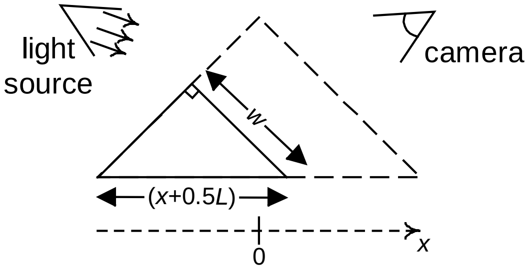


Figure 104. Cross-section of prism.

The upper bound on $\nabla_y S$ is determined by the ray cast from the upper-left corner of the prism cross-section as shown in Figure 105. The maximum $\nabla_y S$ is given by Inequality (A.20).

$$\tan 2\theta_R=\frac{0.5D-y}{w(x)}=\frac{D-2y}{D+\sqrt{2}x} \tag{A.18}$$

$$\theta_S(x,y)=\theta_R=0.5\arctan\frac{D-2y}{D+\sqrt{2}x} \tag{A.19}$$

$$\nabla_y S(y)<\tan\left(0.5\arctan\frac{D-2y}{D+\sqrt{2}x}\right) \tag{A.20}$$

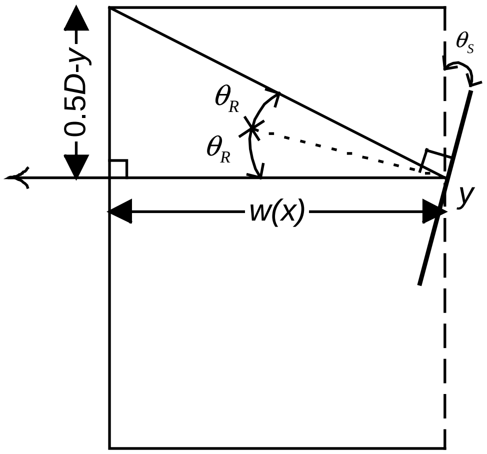


Figure 105. Maximum vertical gradient.

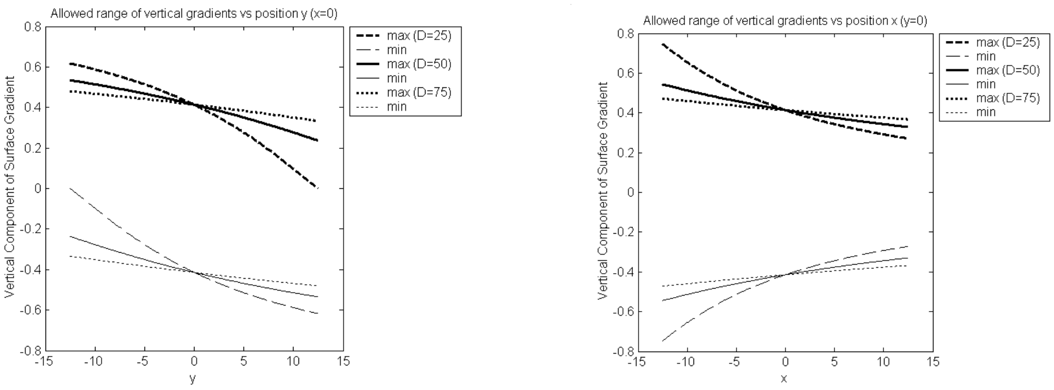
Similarly, the lower bound on $\nabla_y S$ can be computed by replacing *y* by -*y* and changing the sign of the slope. Inequality (A.21) summarizes the range of allowed $\nabla_y S$. Note that while the gradient varies only in *y*, its allowed range varies in both *x* and *y*.

$$-\tan\left(0.5\arctan\frac{D+2y}{D+\sqrt{2}x}\right)<\nabla_y S(y)<\tan\left(0.5\arctan\frac{D-2y}{D+\sqrt{2}x}\right) \tag{A.21}$$

Analysis

Previous sections established that $\nabla_x S = 0$ and that $\nabla_y S$ is constrained by Inequality (A.21). This section analyses the variations in the range of $\nabla_y S$ with respect to position (*x*,*y*) and prism size *D* to gain some insight in the limitations imposed on the surface.

Figure 106(a) shows the variations in $\nabla_y S$ range with respect to the vertical position at *x*=0 and for different values of *D*. It is clear from this diagram that the best flexibility is obtained for greater prism dimensions. It is also clear that the best range is obtained when *y*=0. Figure 106(b) shows the variations in $\nabla_y S$ range with respect to the horizontal position. It is clear from this diagram that lower values of *x* provide a greater range. The prism size, on the other hand, is ambiguous. Greater values of *D* cause the range to widen rapidly with decreasing *x*. However, the range also narrows more rapidly with increasing *x*.



(a) (b)

Figure 106. Variations in range of vertical surface gradient: (a) *x*=0, *y* varying, and (b) *y*=0, *x* varying.

Figure 107 shows a 3D plot of the upper and lower bounds of $\nabla_y S$ with respect to position. We can conclude that the region of interest should be centred on *y*=0 and moved as far towards low values of *x* as possible. Assuming that the region of interest is square with side *l*, it is clear from the above discussion that the most constraining region will be at $x = l - \frac{\sqrt{2}D}{2}$ with *y* varying between –*l*/2 and *l*/2. Since the upper limit decreases with increasing *y*, the worst constraint will occur at *y*=*l*/2 as given by Inequality (A.22).

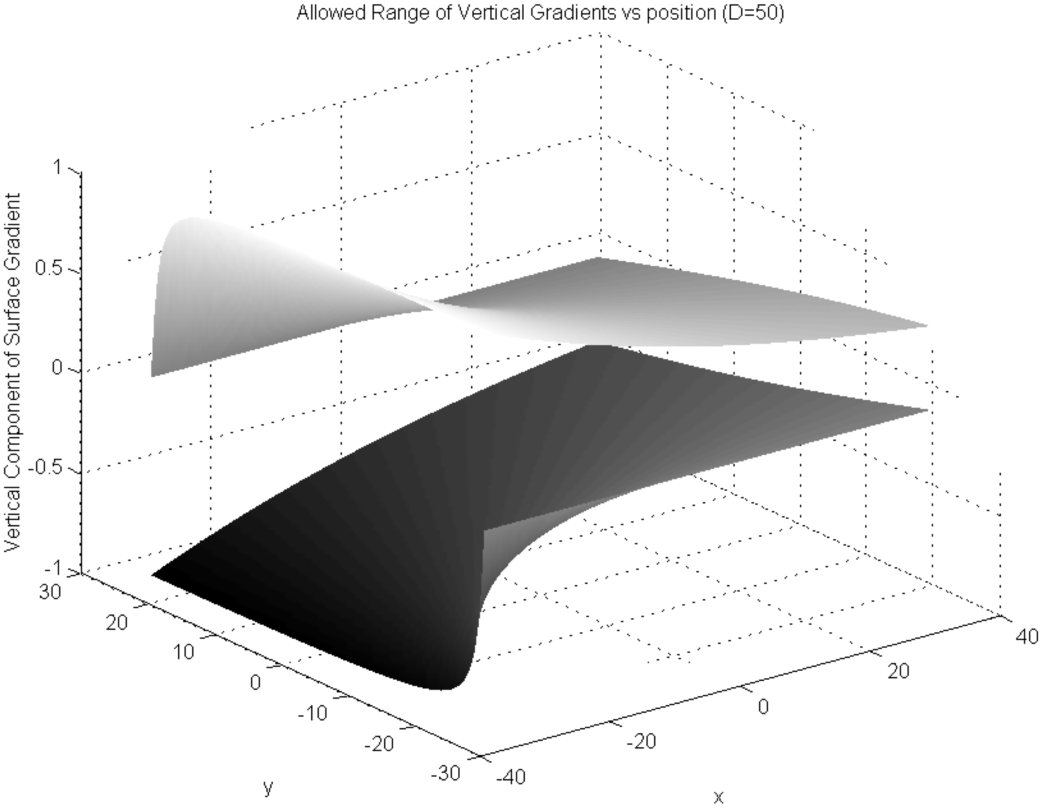


Figure 107. Allowed range of vertical gradients versus position.

$$\text{(A.22)}\quad \nabla_y S(y)_{worst-case} < \tan\left(0.5\arctan\frac{D-l}{\sqrt{2}l}\right)$$

It is interesting to see that the worst-case upper bound increases with *D* and decreases with *l*. The prism should thus be as large as possible and the region of interest should be as small as possible. Assuming a region of interest of *l*=20 mm and a prism of size *D*=50 mm, the limits on the vertical gradient are at worst –0.67 to 0.43 mm/mm (at *y*=10 mm) and –0.43 to 0.67 mm/mm (at *y*=-10 mm), and at best -0.58 to 0.58 mm/mm (at *y*=0 mm).

APPENDIX B - LOCAL AVERAGE ALGORITHM

This appendix describes an efficient algorithm for the computation of a local average in a greyscale image as mentioned briefly in section 4.2. Consider the local average $A(x,y)$ computed within a window of width n as shown in Equation (B.1).

$$A(x,y) = \sum_{j=y-n/2}^{y+n/2} \left(\sum_{i=x-n/2}^{x+n/2} S(i,j) \right) \Bigg/ n^2 \qquad \text{(B.1)}$$

The local averaging operation can be completed in two independent steps. First the horizontal average is computed for all relevant rows. The local average is then the vertical average of the horizontal averages. This process can be further optimized by pre-computing all horizontal averages and storing these values in an image. The local average image is then the result of similarly computing the vertical average over the horizontal average image. It is also possible to avoid redundant computations while computing averages over rows of columns using a sliding window. As the window moves, the local sum is updated by adding the newly covered pixel and subtracting the pixel that is no longer covered. The local averaging algorithm is given in pseudo-code in Figure 108. This algorithm ignores parts of the averaging window that fall outside the image boundary.

```
for each $row of image $img
      $sum=0
      $nbPixelsInSum=0
      for ($left=-2*$avgRadius; $left<$size-$avgRadius; ++$left)
            $right = $left + 2*$avgRadius
            $center = $left + $avgRadius
            if ValidArrayIndex($left)
                  $sum -= $img[$left-1][$row]
                  $nbPixelsInSum--
            if ValidArrayIndex($right)
                  $sum += $img[$right][$row]
                  $nbPixelsInSum++
            if ValidArrayIndex($center)
                  $horAvgImg[$center][$row] = $sum/$nbPixelsInSum
for each $col of $horAvgImg
      [… repeat operations vertically …]
```

Figure 108. Local averaging algorithm.