

**The author(s) shown below used Federal funds provided by the U.S. Department of Justice and prepared the following final report:**

**Document Title:           Analysis of Footwear Impression Evidence**

**Author:                     Sargur N. Srihari**

**Document No.:           233981**

**Date Received:           March 2011**

**Award Number:           2007-DN-BX-K135**

**This report has not been published by the U.S. Department of Justice. To provide better customer service, NCJRS has made this Federally-funded grant final report available electronically in addition to traditional paper copies.**

**Opinions or points of view expressed are those of the author(s) and do not necessarily reflect the official position or policies of the U.S. Department of Justice.**

# *Analysis of Footwear Impression Evidence*

## FINAL TECHNICAL REPORT

**Award Number: 2007-DN-BX-K135**

**SUBMITTED TO:**

U.S. Department of Justice  
Office of Justice Programs  
National Institute of Justice  
810 Seventh Street N.W.  
Washington, DC 20531

**AWARDEE:**

Research Foundation of the State University of New York

**Author:**

Sargur N. Srihari  
520 Lee Entrance, Suite 202  
University at Buffalo, State University of New York  
Buffalo, New York 14260  
Tel: 716-645-6164 ext 113  
Email: [srihari@cedar.buffalo.edu](mailto:srihari@cedar.buffalo.edu)

September 28, 2010

## Abstract

Impressions of footwear are commonly found in crime scenes. The quality and wide variability of these impressions and the large number of footwear outsole designs makes their manual analysis time-consuming and difficult. The goal of this research was to develop new computational methods that will eventually assist the forensic footwear examiner in the U.S. Two scenarios encountered by the forensic examiner were addressed: (i) in the investigative phase, to determine the source of an impression given a known set of outsole prints; which is useful in homicides and assaults where there are no known prints to match, and (ii) in the prosecutorial phase, to determine whether a particular impression evidence is from a known suspect's shoe with a quantification of similarity and uncertainty. The research commenced with developing and acquiring representative footwear print images so that the algorithms developed would relate to the real problem encountered. Algorithms for several sub-problems were studied including image processing to improve the quality of the image for further automatic processing, extraction of features useful for discrimination, a measure of similarity between impressions and a content-based image retrieval system to reduce possible matches with knowns. The principal method pursued was one where the print is characterized as being composed of a pattern of geometric shapes, principally ellipses; with ellipses being able to represent straight line segments and circles as well. A distance measure based on comparing attribute relational graphs was developed. The retrieval system compares evidence features with pre-computed features of database entries and since comparison is time-consuming the database entries are clustered. Retrieval performance is better than that of other methods described in the literature, very few of which deal with real crime scene prints. Future research tasks are indicated including integration of the developed methods into a usable tool and a probabilistic measure of uncertainty in the verification task.

# Contents

<b>1</b>	<b>Executive Summary</b>	<b>2</b>
<b>2</b>	<b>Research narrative</b>	<b>8</b>
2.1	Introduction . . . . .	8
2.1.1	Current practice . . . . .	8
2.1.2	Statement of the problem . . . . .	10
2.1.3	Literature review . . . . .	10
2.1.4	Rationale for the research . . . . .	14
2.2	Methods . . . . .	14
2.2.1	Data sets . . . . .	14
2.2.2	Retrieval system design . . . . .	19
2.2.3	Image pre-processing . . . . .	19
2.2.4	Feature extraction . . . . .	30
2.2.5	Geometrical patterns . . . . .	33
2.2.6	Graph representation . . . . .	42
2.2.7	Graph distance . . . . .	51
2.2.8	Sensitivity analysis . . . . .	60
2.2.9	Clustering . . . . .	64
2.2.10	Retrieval performance evaluation . . . . .	71
2.2.11	Uncertainty computation . . . . .	75
2.2.12	User interface . . . . .	76
2.3	Conclusions . . . . .	76
2.3.1	Discussion of findings . . . . .	76
2.3.2	Implications for policy and practice . . . . .	78
2.3.3	Implications for further research . . . . .	78
2.4	Dissemination . . . . .	79
2.4.1	Publications . . . . .	79
2.4.2	Presentations . . . . .	80
<b>3</b>	<b>References</b>	<b>81</b>



# Chapter 1

## Executive Summary

Impressions of footwear patterns are the most commonly found type of evidence in crime scenes. Yet, the poor quality and wide variability of these impressions as well as the large number of manufactured outsole patterns makes their analysis and courtroom presentation difficult. The objective of this research was to develop new computational methods to assist the forensic footwear examiner in the U.S. both in the investigative phase as well as the prosecutorial phase.

After a review of methods of footwear print examination as practiced in the US, as well as published literature on algorithms for footwear impression analysis, several subproblems were identified as needing solutions: image processing to improve the quality of the image for further automatic processing, extraction of features for class characterization, methods for measuring the similarity of prints for the purpose of ranking the database, identifying distinctive features for individualization, and characterizing uncertainty in individualization.

Two different approaches to separate foreground pixels from background pixels were evaluated. The first uses contextual information present in nearby pixels and is based on a machine learning approach known as conditional random fields. Since it uses contextual information, it performed better than simple image thresholding algorithms such as one based on the valley of the pixel histogram. However, an algorithm based on morphological operations and edge detection performed better both in terms of speed and performance.

Three types of features were compared for their suitability for the task. The following methods were implemented and evaluated : (i) a fixed length feature vector to represent the entire image: which incorporates *gradient, structural and concavity*, or GSC features, and one that has worked well in automatic handwriting recognition, (ii) a variable number of key points in the image, each point represented by a fixed length feature vector: known as the *scale invariant feature transform* or SIFT, used commonly in content-based image retrieval search engines such as Google Similar Images, (iii) a graph representing the structure of the components: an *attribute relational graph* (ARG), based on representing the image as a composite of sub-patterns together with relationships between them. The structural method was found to perform the best and was selected for image retrieval.

The structural method is based on first detecting the presence of geometrical patterns such as short straight line segments, circles and ellipses. The presence of the primitive elements is detected by using variations of a technique known as the Hough transform. The

method is robust even when many of the defining pixels are absent. The relationships between these elements in the print is then modeled as an ARG. Within the ARG, nodes represent primitive elements together with defining attributes relating to parameters such as radius as well as quality in the image. The edges are also attributed with a list of characteristics. The similarity between ARGs is determined by using a graph distance measure, one related to measuring histogram distance and the Wasserstein metric. It characterizes similarity by a number ranging from 0 to 1.

The retrieval task is to find the closest match to a crime scene print in a local/national database so as to determine footwear brand and model. This process is made faster if database prints are grouped into clusters of similar patterns. For this an ARG is constructed for each known print, where each node is a primitive feature and each edge represents a spatial relationship between nodes. The distance between ARGs is used as similarity measure. This distance is computed between each known print and a pre-determined set of canonical patterns to form clusters.

Several data sets were used in the research: (i) simulated prints (crime scene prints obtained by stepping on talcum powder and then on carpet, and known prints by stepping on chemically treated paper), (ii) photographs of outsoles retrieved by a web crawler from shoe-vendor websites, and (iii) 350 actual crime scene prints and over 5,000 known prints. Since results with simulated images tend to be over-optimistic most of the research reported here focused on real crime scene prints.

The results reported are among the first to automatically match crime scene prints to a data base of known prints. The performance appears to be significantly better than the results of another effort, the only one reported in the literature. The efficiency of the algorithms need to be improved before they can be useful for the practitioner. Some of the tasks remaining are converting parts of the code from MATLAB into C++, creating additional user interfaces where user input can be solicited and conversion of the results into a form suitable for courtroom expression.

Future research topics are: (i) the design of efficient algorithms to overcome the combinatorial explosion of relationships between primitive elements, (ii) detection and use of more complex primitive elements, and (iii) expressing the degree of certainty in foot-wear print comparison, e.g., distributions of similarities conditioned on same and different footwear prints, learnt from training samples, is used to determine the likelihood ratio for a print and a known.

# List of Figures

2.1	Simulated crime scene and known prints: (a) print on carpet with powder, and (b) chemical print. . . . .	15
2.2	Shoe photographs of outsoles and uppers on commercial website. Model shown is Nike Air Force 1 which is most often encountered in U. S. crime scenes. . . . .	16
2.3	Some crime scene images in database. . . . .	17
2.4	Some known prints in database. . . . .	18
2.5	Ground Truth associated with database images. . . . .	20
2.6	System flow in retrieval from sole-print image database. . . . .	21
2.7	Results of image pixel labeling: (a) latent image, (b) Otsu thresholding, (c) neural network thresholding and (d) CRF segmentation. . . . .	24
2.8	Adaptive thresholding results: (a) original impression (b) enhanced image using adaptive thresholding. . . . .	25
2.9	Results of edge detection on crime scene images: (a), (c) and (e) are originals, (b),(d) and (f) are corresponding edge images. . . . .	26
2.10	Results of edge detection on data base images. . . . .	27
2.11	Morphological Operations for Shoe-print Enhancement. . . . .	28
2.12	Results of edge detection showing intermediate morphological operations on data base images. . . . .	29
2.13	GSC representation: (a) shoe-print characterized by (b) a 1024 dimensional binary feature vector. . . . .	31
2.14	SIFT Representation: (a) key points where each blue arrow shows key point orientation, and (b) descriptor for one key point. . . . .	32
2.15	Footwear outsole patterns containing line segments only. . . . .	34
2.16	Footwear outsole patterns containing circles only. . . . .	34
2.17	Footwear outsole patterns containing ellipses only. . . . .	35
2.18	Footwear outsole patterns containing lines and circles. . . . .	35
2.19	Footwear outsole patterns containing lines and ellipses. . . . .	36
2.20	Footwear outsole patterns containing circles and ellipses. . . . .	36
2.21	Footwear outsole patterns containing lines, circles and ellipses. . . . .	37
2.22	Footwear outsole patterns containing texture only. . . . .	37
2.23	Elimination of spurious ellipses using gradient orientation. (a) before elimination, and (b) after elimination. . . . .	40
2.24	Shapes automatically detected in crime scenes: (a) print where circles (red) are prominent, and (b) print where line segments (green) are prominent. . . . .	40

LIST OF FIGURES

LIST OF FIGURES

2.25 Shapes detected in database prints: lines, circles and ellipses are shown in green, red and blue respectively. . . . . 41

2.26 Normalization of some attributes is done using the squashing function  $f(x) = x/\sqrt{1+x^2}$ . When only positive vales of  $x$  are input, the attribute is normalized to have values in the interval  $[0,1]$ . . . . . 44

2.27 Relative position definitions for line-line, line-circle and circle-ellipse. . . . . 45

2.28 Attribute relational graph of a crime scene print: (a) print image, (b) detected circles and straight lines with magnified sub-image showing three straight lines to left of circle, (c) centers of 61 straight line segments (green points) and 5 circles (red), and (d) sub-graph for the three straight lines and circle in (b) whose attributes are given in Table 2.5. . . . . 49

2.29 Illustration of distance computation between two simple prints: (a) print  $P_1$  with five primitive elements, (b) attributed relational graph of  $P_1$  with vertices  $V_{11}..V_{15}$ , (c) attributed tree rooted at  $V_{11}$ , (d) print  $P_2$  with six elements, (e) attributed relational graph of  $P_2$  with vertices  $V_{21}..V_{26}$  and (f) attributed tree rooted at  $V_{21}$ . Using the attribute values shown in Tables 2.6 and 2.7 the distance evaluates to 0.5674. . . . . 56

2.30 Similarity between a crime scene and a known print:(a) inputs, (b) detected lines and circles, and (c) graphs, where only nodes are shown for clarity, and similarity value. . . . . 59

2.31 Sensitivity of distance measure to variations in each attribute, showing that D is *insensitive* to small errors in feature extraction. There are fifteen graphs– from top to bottom, left to right: labelled (a)-(e), (f)-(j), (k)-(o). A linear correlation is seen between D and most attributes indicating that D *consistently* measures human perceptual distance. The exceptions are (b, j, n, o) which are explained as follows: (b) D drops after reaching peak, e.g., say two lines  $l_1$  and  $l_2$ , where  $l_1.len > l_2.len$ , as  $l_1$  becomes shorter  $l_1.len < l_2.len$  is reached and in that case the algorithm will switch their roles to minimize D, (j) Similar reason as (b), (n) D initially increases fast and then saturates because when the major axes of two ellipses is far, the rate of change in  $rp$  becomes increasingly small, (o) when the radius  $r$  of one of the two circles vary randomly within 15%, the change of D is always below 0.025. . . . . 61

2.32 Example of two 2-node prints used in sensitivity analysis. . . . . 62

2.33 Plot of distance  $D$  against scale factor  $Q_k$  for attribute  $E2E$   $N-rd$ . Prints with only two nodes were used in the experiments.  $N-rd(P_1) = 0$ ,  $N-rd(P_2) = 1$ . . . . . 63

2.34 Process flow for clustering footwear outsole prints in database. . . . . 65

2.35 Canonical Patterns for Clustering. . . . . 66

2.36 Clustering Step 1(Morphology): (a) Original Gray-scale Image, (b) Edge Image of (a), (c) Result of Morphological Operation on (a), (d)Edge Image of (c). . . . . 66

2.37 Clustering Step 2 (Hough Transform). Extracting features in the sequence circle→ellipse→line: (a) Circles. (b) Ellipses. (c) Line Segments. (d) All features. Red box indicates a small region in the footwear print. . . . . 67

*LIST OF FIGURES*

*LIST OF FIGURES*

---

2.38 Clustering Step 3 (ARG): (a) nodes in graph corresponding to image in Figure 2.37 with edges omitted due to complete connectivity, (b) subgraph for region enclosed in the red box of Figure 2.37 (d). Red and green dots represent circles and lines respectively. . . . .	68
2.39 Sample clusters based on using the canonical patterns in Fig. 2.35. . . . .	69
2.40 Retrieval Performance: F-measure corresponds to its maximum value. . . . .	70
2.41 Results of automatic retrieval with two queries shown as the left-most images followed on the right by the top database entries retrieved. It can be seen that the top choices are similar to human perception. . . . .	72
2.42 Cumulative Match Characteristic of ARG-EMD and SIFT. . . . .	74
2.43 Probabilistic Model: intra- and inter-class distributions of distance allow computing likelihood ratios. . . . .	75
2.44 User Interface: (a) opening screen with tool bar, and (b) a query and results. . . . .	77

# List of Tables

2.1	Distribution of geometric patterns in footwear outsole prints. . . . .	33
2.2	Node Attribute Definitions . . . . .	43
2.3	Edge Attribute Definitions . . . . .	47
2.4	Edge Attribute Definitions (Continued) . . . . .	48
2.5	Node and Edge Attributes for subgraph in Figure 2.28(d). . . . .	50
2.6	Node and edge attribute values of print $P_1$ shown in Fig. 2.29 (a) . . . . .	57
2.7	Node and edge attributes of print $P_2$ shown in Fig. 2.29 (d) . . . . .	58
2.8	Cost Matrices in comparing $P_1$ and $P_2$ . . . . .	59
2.9	Comparison of ARG-FPD approach with the state-of-the-art . . . . .	73
2.10	Retrieval speed before and after clustering. . . . .	74

# Chapter 2

## Research narrative

### 2.1 Introduction

Footwear impression marks – the mark made by the outside surface of the sole of a shoe (the outsole) – are distinctive patterns often found at crime scenes. They are among the most commonly found evidence at crime scenes and present more frequently than fingerprints. Footwear marks provide valuable forensic evidence. In many instances, shoe marks can be positively identified as having been made by a specific shoe to the exclusion of all other shoes. Identification is based on the physical match of random individual characteristics the shoe has acquired during its life. Evidence provided by a positively identified shoe mark is as strong as the evidence from fingerprints, tool marks, and typewritten impressions [1].

Footwear impressions are created when footwear is pressed or stamped against a surface such as a floor or furniture, in which process, the characteristics of the shoe is transferred to the surface. Footwear marks can be broadly broken into two classes: 1) impressions which contain 3-dimensional information (e.g., on snow, wet dirt or at the beach) and 2) impressions which contain 2-dimensional information (e.g., on a floor or carpet).

There is variability in the quality of footwear impressions because of the variety of surfaces on which the impressions are made. Detail retained in a shoe mark may be insufficient to uniquely identify an individual shoe but is still very valuable. Due to the wide variety of shoes available on the market, with most having distinctive outsole patterns, this implies that any specific model of shoe will be owned by a very small fraction of the general population. Furthermore the same outsole pattern can be found on several different footwear brands and models. If the outsole pattern of a shoe can be determined from its mark, then this can significantly narrow the search for a particular suspect.

#### 2.1.1 Current practice

The forensic examiner collects and preserves footwear and tire tread impression evidence, makes examinations, comparisons, and analyses in order to: (i) include, identify, or eliminate a shoe, or type of outsole, as the source of an impression, (ii) determine the brand or manufacturer of the outsole or footwear, (iii) link scenes of crime, and (iii) write reports and provide testimony as needed. The photograph of the impression or of the lifted impression or

cast can be subsequently scanned and a digital image produced. Forensic analysis requires comparison of this image against specific databases. These databases include: (i) marks made by shoes currently and previously available on the market and (ii) marks found at other crime scenes.

An image of a shoe mark can be obtained using photography, gel, or electrostatic lifting or by making a cast when the impression is in soil. Subsequently, in the forensic laboratory, the image of the shoe mark is compared with the shoe-prints and shoe impressions of known shoe samples. Interactive image enhancement operations are available in Photoshop and other image processing software that are available to the footwear examiner.

Footwear images collected directly from crime scenes are of poor quality. The environment under which the questioned shoe print is lifted at the crime scene is different from those available in the known prints. One approach is to design digital image enhancement techniques, such as contextual thresholding, to enhance the quality of questioned shoe-prints to achieve feasibility of matching shoe-prints in the database. Debris and shadows and other artifacts in the crime scene impressions are difficult to filter out from footwear impressions. They have interfered with attempts to store and search in the database. Therefore, after digital image enhancement, some algorithms are desired to be able to classify different regions of footwear impression to be one of two types: useful regions (impressed by footwear) and discardable regions (impressed by other artifacts such as debris).

In a computerized tool to assist in identification, firstly, known shoe-prints are scanned, processed and indexed into a database. The collection of test prints involves careful human expertise in order to ensure the capture of all possible information from the shoe-print. All such information is indexed into a database so as to be matched against shoe-print evidence. An automatic footwear identification system accepts as input shoe-print evidence and retrieves the most likely matching prints.

There has been significant research conducted in footwear-print analysis in Europe focusing on the needs of the European forensic community. There are important differences for the task in the US<sup>1</sup>. Homicides and assaults are paid more attention to than burglaries in the U.S., where shoe prints have a very low likelihood of appearing in other cases. Due to this reason the classification task, i.e., determining brand, style, size, gender etc., is of importance. Through such classification, even if the person could not be identified, the search could be narrowed down to a smaller set of suspects.

Forensic examiners of footwear and tire impression evidence are a community of about

---

<sup>1</sup>Europe has a few locations that collect sufficient footwear impressions from scenes to assemble into a data base, which will be searched with detected impressions from future burglaries. Approximately 30% of crime scenes have usable shoe-prints[2]. A study of several jurisdictions in Switzerland revealed that 35% of crime scenes had usable footwear impressions in forensic investigation, and 30% of all burglaries provide usable impressions[3]. It is known that the majority of crimes are committed by repeat offenders and it is common for burglars to commit a number of offenses in the same day. As it would be unusual for an offender to discard footwear between crimes[4], timely identification and matching of shoe-prints allows different crime scenes to be linked. Since manual identification is laborious there exists a real need for automated methods.

However, this is not the practice in the US. Most crimes that time is spent on in the US are not burglaries, but homicides and assaults. In those cases, particularly homicides, there is far less likelihood that those impressions will appear in another case.



200 professionals in the United States<sup>2</sup>. Footwear prints constitute about 80-90% of the case-work of the tread examiner who deals with both footwear and tire-marks.

Due to its time consuming nature, footwear impression evidence is not used as frequently as it could be<sup>3</sup>. This is because footwear impressions are usually highly degraded, prints are inherently complex and databases are too large for manual comparison.

### 2.1.2 Statement of the problem

The tasks for the forensic footwear examiner are: (i) identification of class characteristics by comparing the evidence against a possibly large set of knowns to determine generic footwear brand, gender and size, and (ii) individualization of a known print as having been source of the evidence. Comparing crime scene shoe mark images to databases, a task encountered in the *investigative phase*, is currently a difficult and laborious task and it is commonly performed manually by searching through catalogs or computer databases. An individualization statement, useful for the *prosecutorial phase*, is realistically accompanied by a statement of uncertainty involved. The goal of this research was to explore computational methods for both phases, keeping in mind the needs of the U. S. forensic community. Computer-based methods that reduce operator effort offers great benefit to forensic scientists and the criminal justice system.

Tasks to be addressed were:

1. Develop or identify suitable image processing algorithms to enhance the quality of the images for further processing.
2. Evaluate or develop feature extraction methods that are; (i) suitable for describing geometrical patterns in outsoles and (ii) are robust in processing poor quality crime scene images.
3. Develop similarity measures for the comparison of footwear prints based on the features extracted.
4. Determine metrics to evaluate the algorithms and measures so developed on a realistic and significant sized data set.
5. Develop measures for characterizing uncertainty of match between evidence and known.

### 2.1.3 Literature review

Most research on automated techniques have originated from European research groups where the needs have important differences than in the U. S. The literature is described below under the overlapping headings of: semiautomated methods, classification, feature extraction and interpretation.

---

<sup>2</sup>Guidelines for the profession are given on the IAI website dealing with the Scientific Working Group on Footwear and Tire Tread Evidence (SWGTTREAD).

<sup>3</sup>For example, in 1993, only 500 of 14,000 recovered prints in the Netherlands were identified [5].

### Semi-automated methods

A number of semi-automatic shoe-print classification methods have been reported [3, 6]. Early work [7, 8] involves semi-automatic methods of manually annotated footwear print descriptions using a codebook of shape primitives, e.g., wavy patterns, geometric shapes and logos. The query print needs encoding in a similar manner. The most popular semi-automated systems today are SOLEMATE and SICAR [9, 10]. These systems rely on manually encoding shoe-prints using a codebook of shapes and geometric primitives, such as wavy patterns, zig-zags, circles, triangles. Then searching for similar patterns for a given query shoe-print requires it to be encoded in a similar manner. This process is laborious, time-consuming and can be the source of poor performance as the same pattern can be encoded differently by different users. The study of automatic shoe-print pattern classification is still new, immature and has not been adopted.

### Features

Features for class characterization are those image measurements that are useful for discriminating between different sole types. They capture the geometry of the pattern so as to be able to distinguish it from every other sole type. Since there a large number of sole types, the task of determining sole type is a problem of image retrieval where the query is the print of unknown type and the database consists of all known prints.

Features for individualization are characteristics that are unique to the particular shoe that made the crime scene print. Characteristics for individualization based on shoe sole defects are described by Stone [11]. Defects consist of nicks, scratches, cuts, punctures, tears, embedded air bubbles caused by manufacturing imperfections, and ragged holes. A combination of position, configuration, and orientation of each defect, which are the result of events that occurred in its life, are unique to each shoe. A defect position is characterized relative to: shoe print perimeter, particular tread elements or portions of patterns, or other defects. A defect shape is characterized by its length, width, and other shape measures. The rotational orientation of the defect helps differentiate from other similarly shaped defects. These individual characteristics, along with the class characteristics, enable determining whether a crime scene print matches a known.

Feature-point based methods, such as SIFT (Scale invariant feature transform) [12], have demonstrated good performance in image retrieval due to invariance with respect to scale, rotation and translation. However, they may be inappropriate for shoe-prints. This is partly because, as local extrema in the scale space, SIFT key points may not be preserved both among different shoes of the same class and through the life-time of a shoe. This problem is further complicated by the extremely poor quality and incompleteness of crime scene footwear impressions. Pavlou and Allinson (2006) [13] presented footwear classification results where maximally stable extremal region (MSER) feature detectors are encoded with SIFT descriptors as features after which a Gaussian feature similarity matrix and Gaussian proximity matrix are used as the similarity measure. In some crime scenes, only partial shoe-prints (termed as “half prints” and “quarter prints”) are available. Partial shoe-print matching has to focus on how to fully make use of regions available, with the accuracy of

matching algorithms decreasing with print size.

Prints found at crime scenes can be used to narrow-down the search space. This is done by elimination of the type of shoe, by matching it against a set of known shoe-prints (captured impressions of many different types of shoes on a chemical surface). Most existing footwear print retrieval systems are semi-automatic. De Chazal et al. [14] proposed a fully automated shoe print classification system which uses power spectral density of the print as a pattern descriptor; crucial information of the print is preserved by removing low and high frequency components. Zhang et al. [15] proposed an automated shoe-print retrieval system in which edge direction histogram is used to find the closest matching print. There is no published literature on mining footwear print databases to aid in retrieval. As an exercise in data mining, Sun et. al. [16] clustered shoe outsoles using color (RGB) information as features where the number of clusters  $k$  was varied from 2 to 7 and the clustering results of  $k$ -means and expectation maximization were compared; the results are of limited use since RGB information of outsole photographs are absent in impression evidence.

Algarni and Hamiane (2009) [17] proposed an automatic shoe-print retrieval system in which Hu's moment invariants are used as features. Then results from standard similarity measures like Euclidean, city block, Canberra and correlation distances are compared. Xiao and Shi (2008) [18] presented a computerized shoe-print matching using PSD and Zernike moments. Jing et al. (2009) [19] presented a new feature, directionality to match shoe-prints. Here, features extracted from co-occurrence matrix, Fourier transform and directional mask are matched using sum-of-absolute-difference. Nibouche et al. (2009) [20] proposed a solution for matching rotated partial shoe-prints. Harris points encoded with SIFT descriptors are used as features and they are matched using random sample consensus (RANSAC).

Dardi et al. (2009) [21] described a texture based retrieval system for shoe-prints. A Mahalanobis map is used to capture texture and then matched using a correlation co-efficient measure. In subsequent work [22, 23] they offer a cumulative match score comparison between Mahalanobis, [14] and [24].

Wang et al. (2009) [25] presented a wavelet and fuzzy neural network to recognize footprints. Patil et al. (2009) [26] proposed using the Gabor transform to extract multi-resolution features and then the Euclidean distance for matching.

### **Automatic classification**

Mikkonen and Astikainen (1994) [27] proposed a classification system for shoe-prints in which classification codes based on basic shapes are used as a pattern descriptor to identify and classify the partial footwear impressions. Geradts and Keijzer (1996) [5] described an automatic classification for shoe outsole designs. Here, different shapes in shoes are recognized using Fourier features and then these features are used in a neural network to classify the footwear. Alexander et al. (1999) [2] presented a fractal pattern matching technique with mean square noise error as a matching criteria to match the collected impression against database prints. de Chazal et al. (2005) [14] proposed a fully automated shoe print classification system which uses power spectral density (PSD) of the print as a pattern descriptor. Here, PSD is invariant to translation and rotation of an image, crucial information of the print is preserved by removing the low and high frequency components and 2D correlation coefficient

is used as similarity measure. Zhang and Allinson (2005) [15] proposed an automated shoe print retrieval system in which edge direction histogram is used to represent the shapes in shoes. The features consist of 1-D discrete Fourier Transform (FT) on the normalized edge direction histogram and the Euclidean distance is used as similarity measure.

The approach of [5] employs shapes generated from footwear prints using image morphology operators. Spatial positioning and frequencies of shapes are used for classification with a neural network. No performance measures are reported. [2, 4] uses fractals to represent prints and mean square noise error classification. FT features, which are invariant to translation and rotation, have also been used for classification of full and partial prints [28, 14]. First and fifth rank classification are 65% and 87% on full-prints, and 55% and 78% for partials. The approach shows that although footwear prints are processed globally they are encoded in terms of the local information evident in the print. In [15] pattern edge information is employed for classification. After image de-noising and smoothing operations, extracted edge directions are grouped into a quantized set of 72 bins at five degree intervals. This generates an edge direction histogram for each pattern which after applying a Discrete FT provides a description with scale, translational and rotational invariance. The approach deals well with variations, however query examples originate from the learning set and no performance is given for partial prints.

In [2], fractals are used to represent shoe-prints and a mean squared noise error method is adopted for the final matching. Fourier transform is used in [14] to process the full and partial prints and a 2D correlation coefficient similarity measure is used for matching. Most recently, Gabor transform [26] has been used to extract multi-resolution features of a shoe-print. Rotation of a shoe-print image is estimated by Radon Transform and compensated by rotating the image in opposite direction.

Ghouthi et al. (2006) [29] describe a so-called ShoeHash approach for classification where directional filter banks (DFB) are used to capture local/global details of shoe-prints with energy dominant blocks used as feature vector and normalized Euclidean-distance similarity. Su et al. (2007) [30] proposed a shoe-print retrieval system based on topological and pattern spectra, where a pattern spectrum is constructed using the area measure of granulometry, the topological spectrum constructed using the Euler number and a normalized hybrid measure of both used for matching. Crookes et al. (2007) [31] described two ways to classify shoe-prints: (i) in the spatial domain, modification of existing techniques: Harris-Laplace detectors and SIFT descriptors is proposed; the Harris corner detector is used to find local features; Laplace based automatic scale selection is used to decide the final local features and a nearest neighbor similarity measure, and (ii) in the transform domain, phase-only correlation (POC) is used to match shoe-prints. Gueham et al. (2008) [24] evaluated the performance of Optimum Trade-off Synthetic Discriminant Function (OTSDF) filter and unconstrained OTSDF filter in classifying partial shoe-prints.

## Interpretation

For an automated shoe-print retrieval and examination system, the role of a forensic scientist is not only to provide a technical solution, but also to interpret the results in terms of the strength of the evidence it can support. There are many statistical methods for computing

the strength of evidence, e.g., [32], for presenting forensic evidence in the courtroom.

For evidence interpretation, three different approaches have been stated: “Classical”, “Likelihood Ratio” and “Full Bayes’ Rule”. The likelihood ratio approach [33] is widely accepted among various forensic investigations as it provides a transparent, consistent and logical framework to discriminate among competing hypotheses. In the Full Bayes’ Rule approach, the posterior probability of a set of hypotheses given the existing evidence is determined. Although this method has been a very common practice of forensic document examiners in central European countries, it has been said that there is no creditable justifications for its validity and appropriateness[34].

In order to establish a uniform ground for interpreting shoeprint evidence, ENFSI shoeprint and toolmark Working Group have proposed a 5-point conclusion scale<sup>4</sup>, ranging from identification, very strong (strong) support, moderately strong support, limited support. A rule for converting likelihood ratios into scales has also been suggested[35].

### 2.1.4 Rationale for the research

While there have been several academic papers on the design of algorithms for footwear image analysis none have dealt with the challenging problem of real crime scene impressions. The large number of outsoles manufactured makes the database matching problem a difficult one. There are many image processing and feature extraction algorithms in the literature and it is not clear as to which ones are most suitable for the problem, The need for matching of prints in a time-efficient manner poses another requirement on the algorithms designed.

## 2.2 Methods

The research methods are described under the following headings: (i) Data Sets, (ii) System Design, (iii) Image Pre-processing, (iv) Feature Extraction, (v) Graph Representation, (vi) Similarity Measurement, (vii) Clustering, (viii) System Evaluation and (viii) Uncertainty Computation.

### 2.2.1 Data sets

Three different data sets were created/acquired for this research:

1. *Simulated Crime Scene*: Volunteers were asked to step on talcum powder and then onto a carpet to create a simulated crime scene print. Their prints were also captured on chemical paper to create the known. Both were converted into digital camera images, examples of which are shown in Fig. 2.1. Since the simulated crime scene prints were of relatively high quality this led to over-optimistic results in both verification and identification.

---

<sup>4</sup>[http://www.intermin.fi/intermin/hankkeet/wgm/home.nsf/files/Harmonized\\_Conclusion\\_Scale\\_of\\_EWGM/\\\$file/Harmonized\\_Conclusion\\_Scale\\_of\\_EWGM.pdf](http://www.intermin.fi/intermin/hankkeet/wgm/home.nsf/files/Harmonized_Conclusion_Scale_of_EWGM/\$file/Harmonized_Conclusion_Scale_of_EWGM.pdf)

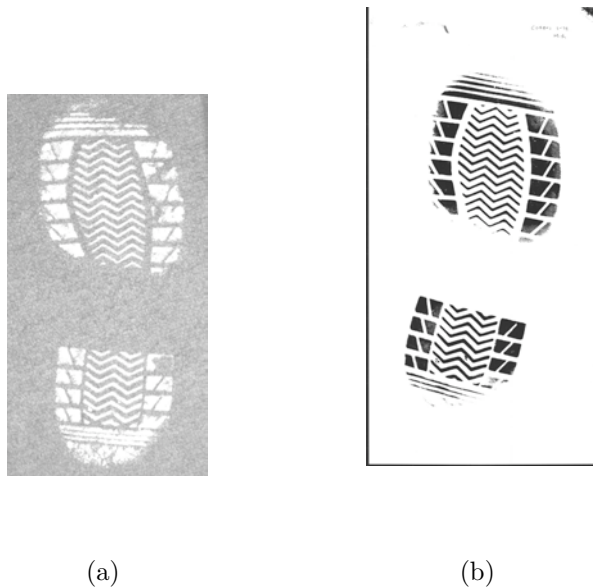


Figure 2.1: Simulated crime scene and known prints: (a) print on carpet with powder, and (b) chemical print.

2. *Photographs of Outsoles*: A web crawler was written to download outsoles of shoes in commercial vendor websites. About 10,000 such images were downloaded. An example of the types of photographs available is given in Fig. 2.2.
3. *Crime scene database*: This database consists of 350 crime scene images and over 5,000 known prints. The known prints were created by taking impressions of footwear outsoles provided by footwear vendors. Sample crime scene images are shown in Fig. 2.3, and samples from the known set are given in Figure 2.4. The ground truth for the crime scene database is in the form of Excel spread-sheets as shown in Fig. 2.5.



Figure 2.2: Shoe photographs of outsides and uppers on commercial website. Model shown is Nike Air Force 1 which is most often encountered in U. S. crime scenes.

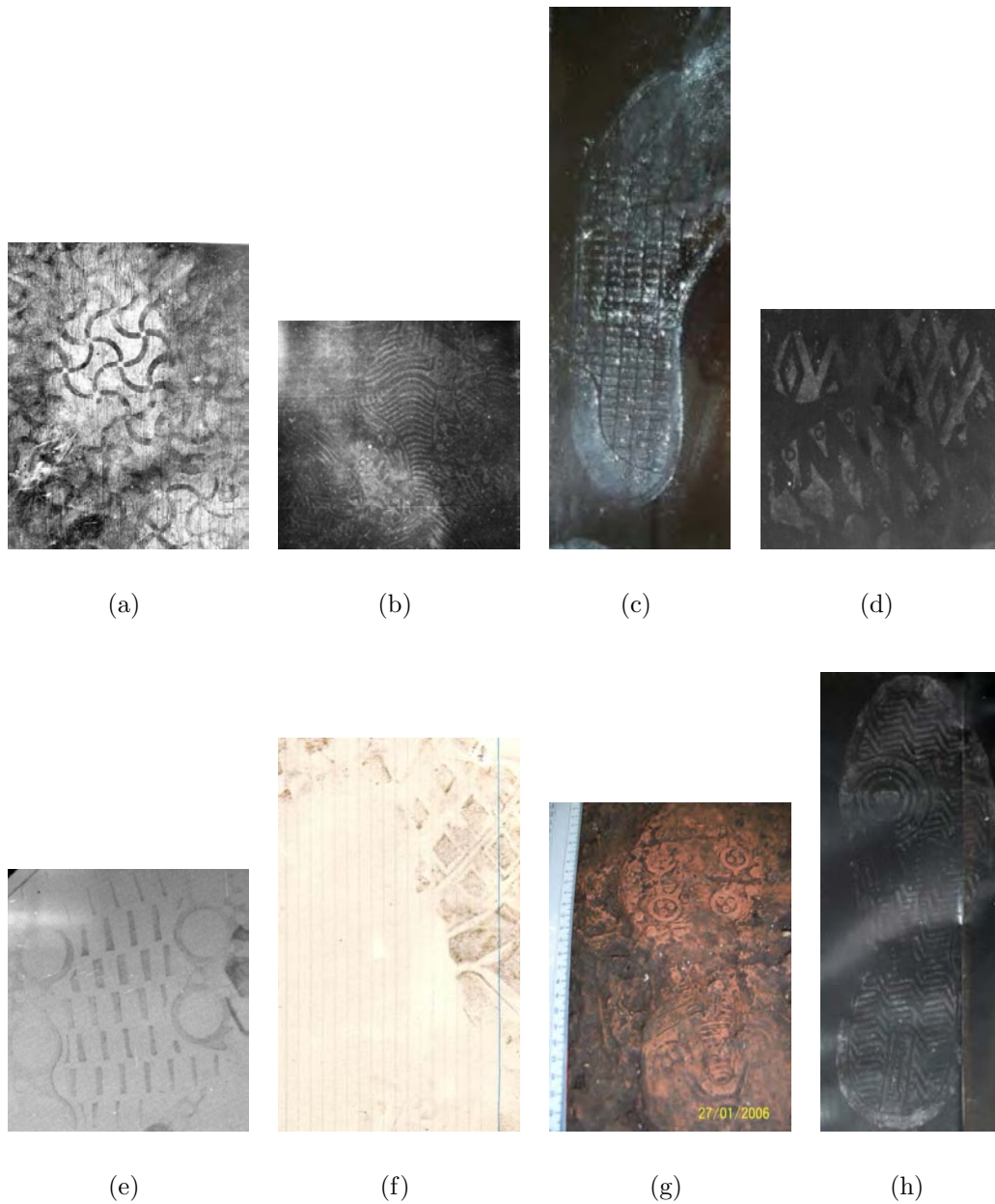


Figure 2.3: Some crime scene images in database.



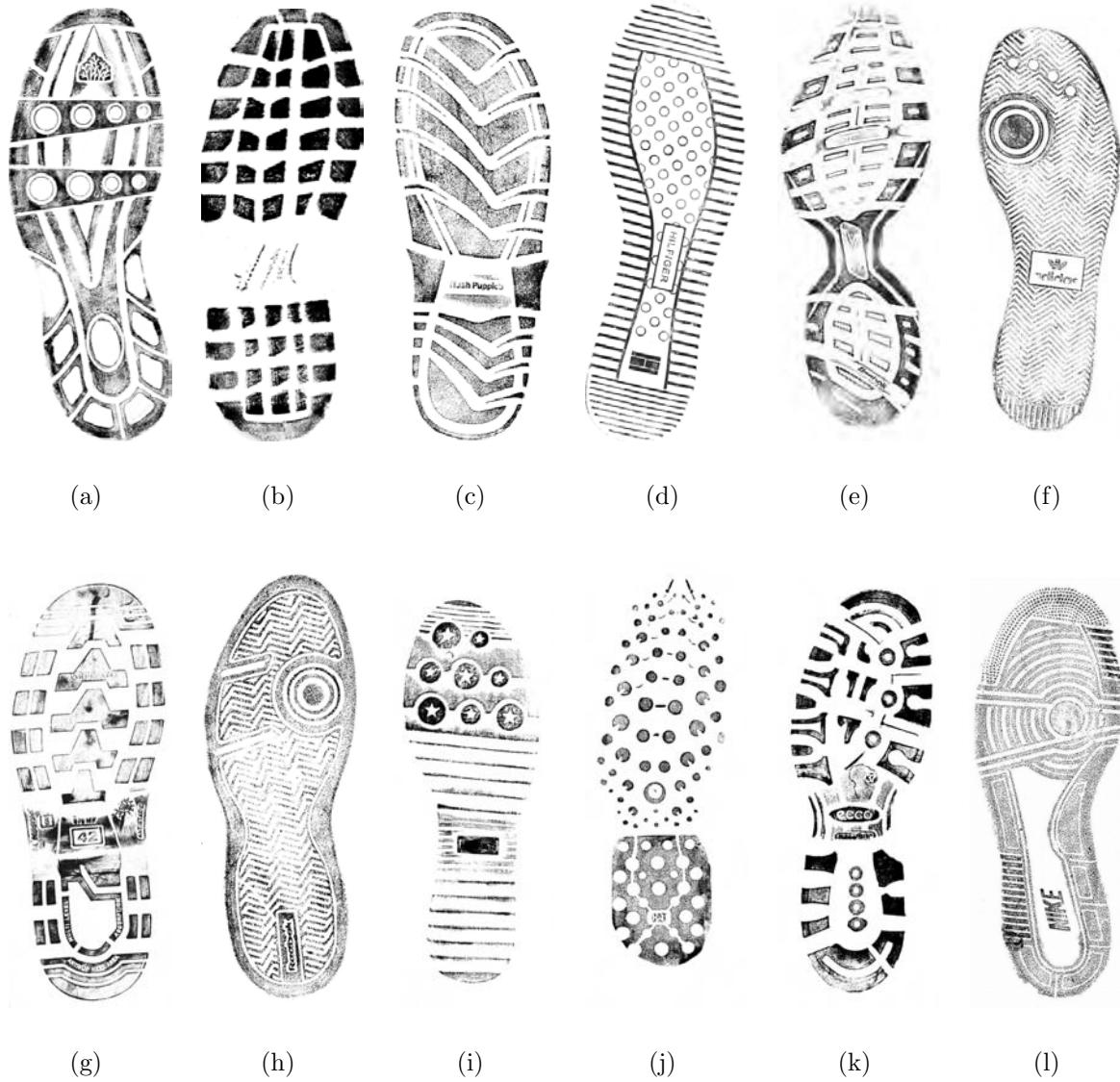


Figure 2.4: Some known prints in database.

There are multiple prints from the same crime scene, e.g., in the first set 194 crime scene images are from 176 crime scenes and 144 crime scene images in the second are from 126 crimes. Each crime scene image of the 50 crime scene images in the first dataset came from a different crime scene. Among these there are multiple shoe prints such as two partial shoe marks from the same crime scene, same marks taken at different illumination, same marks taken at different angles/orientation etc. We plan to combine the partial shoe marks at the image level for the shoe marks which has some degree of overlap between them. For shoe marks taken at different illumination and different orientation, the best image— one that is rich with features— would be interactively chosen by a human operator.

The resolution of database images varies from 72 dpi to 150 dpi. Crime scene image resolution varies from 72 dpi to 240 dpi. The crime scene image dataset contains an equal number of color and gray-scale images. Only 3% of the database images are direct photographs of the outsole of brand new shoes. The database images can be broke down as follows. 97% are gray scale images. they are actually prints. 3% are color images, which are direct photographs of the outsole of the shoes on the market. Very few (less than 0.1%) are binary images.

### 2.2.2 Retrieval system design

An overview of the processes necessary for a system to retrieve closest matches to a query image are shown in Figure 2.6. image enhancement operations are used on both crime scene and known images. Examples of such techniques are edge detection or contextually based image pixel labeling. Next, we build a feature representation for the image either by extracting them from the entire image or by detecting local patterns found in outsoles. The design should attempt to integrate several levels of analysis: (i) global shoe properties: heavily worn or brand new, shape, size etc., (ii) detailed and distinctive local features should be utilized to increase the discriminative power in order to confirm a match. Each level requires a different variety of image analysis techniques from robust geometric and texture feature detectors to detailed correlation of distinctive minutiae and their spatial arrangement.

A similarity measure appropriate to the feature description is used in the comparison of two images. Based on experiments with several approaches the final method chosen was a graph representation where each node denotes a single geometrical primitive, such as a circle, an ellipse, a line segment, with attributes describing unary features of this primitive; each attributed edge between a pair of nodes represents spatial relationships between them. Thus the problem of image retrieval and matching is converted to an attributed graph matching problem. It involves establishing correspondence between the nodes of the two graphs. Retrieving the most similar prints to an impression can be made faster by clustering the database prints beforehand.

### 2.2.3 Image pre-processing

The matching of crime scene impressions to known prints largely depends on the quality of the extracted image from the crime scene impression. Thus the first step in dealing with

2.2. METHODS

CHAPTER 2. RESEARCH NARRATIVE

Answers for "Crimes and Matches (144 images from 126 crimes Jan-Dec-07)" Test Data					
Test Image No	Brand	Model(s)	ImageID	Group Brand	Group ImageID
1	K-Swiss	Rinzler, Rinzler Jewel, Rinzler SP	66907		
2	Adidas	Thalcke	22262		
3	Lacoste	Solano, Kurat	53151		
4	Umbro	Kingston	62834		
5	Fuma	Race Car J, Rep Car Mid J, Speed Car	15412		
6	Nike	Charis	450	Adidas	141
7	Nike	Air Max Huarsche 2	42280		
8	Nike	Brun Canalis, Blazer Suede	27636	Fape Jeans	19477
9	Ascent	LK2049V	45562	Umbro	7376
10	Adidas	360 Shell	22065	Kowiss	54494
11	Giorgio	Logiter Boot	28847		
12	Vans	Larzer, Lori	4459E	Rugged Trail	41203
13	Nike	Air Force I, Air Force Low	42251	Whamplur	28738
14	Reebok	Classic Leather Mustang	20422		
15	Blox	Blaze, Hausman	26925	Path	31703
16	Lacoste	Score, Score V, Score Strap	53147		
17	Nike	Air Force I, Air Force Low	42251		
18	1992	Hemp, Gothic, Shields (Gross)	58998	Efense	75317
19	GrSport	Contractor, Director, Combat	16205		
20	FDQ	Ingleton	51241	Adlan	30522
21	H-Tec	Miami	39071	MAX2	71727
22	Columbia	Blacknick, Thundercoat	19471	H-Tec	1982
23	Joe Boxer	Skate 47460	38372	Pyneess	35143
24	Nike	Air Max Deluxe E	8150	Nike	37550
25	Wilson	Prostaff 1000, Prostaff 710	40614	Wilson	45615
26	Globe	Monta, Flux, Falcon	21012		
27	K-Swiss	Olsen, Farias	51165		
28	Frank Wright	Weiler	1753		
29	Thom McKn		64656		
30	Lee		20446		

Figure 2.5: Ground Truth associated with database images.



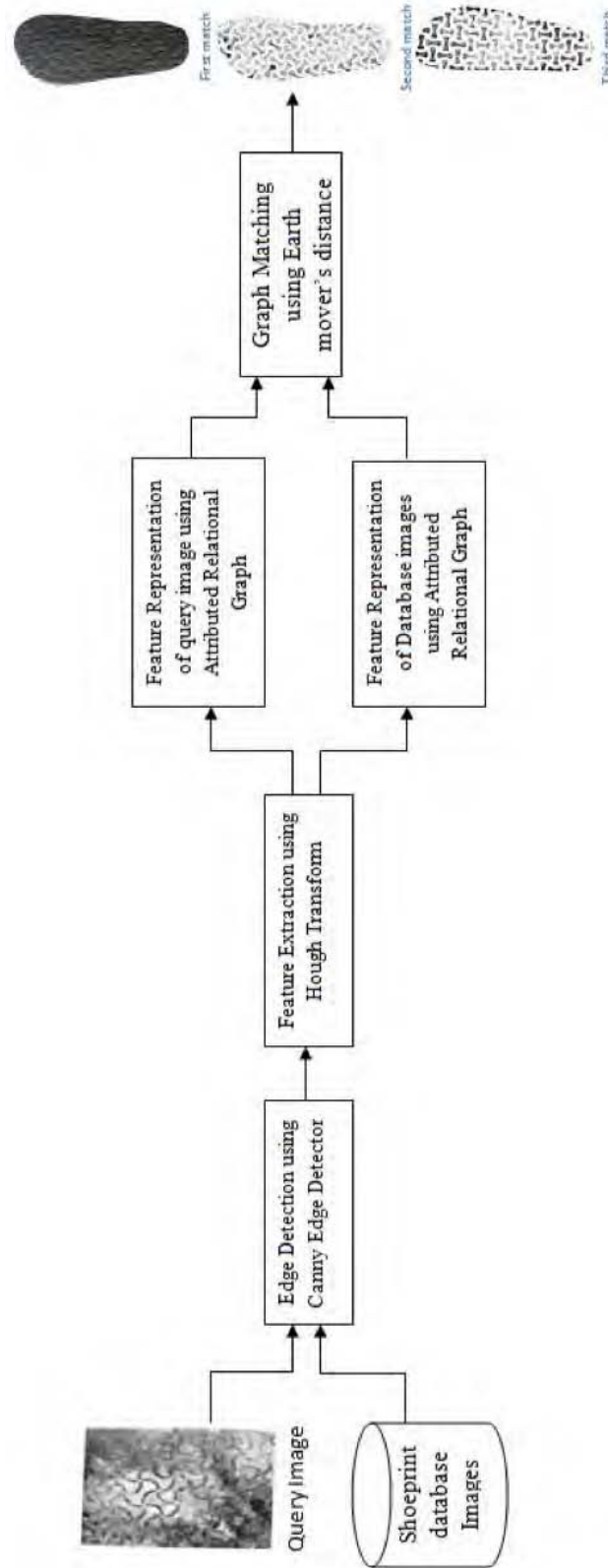


Figure 2.6: System flow in retrieval from sole-print image database.

both crime scene prints and database prints is that of processing them in a way that makes further processing more effective and/or efficient. Two approaches were tried: image labeling and edge detection.

### Image pixel labeling

The shoe-print image enhancement problem is formulated as an image labeling problem. Different pixels or regions of the image are labeled as foreground (shoeprint) or background. The labeling problem is naturally formulated as a machine learning task and has several different approaches. Different machine learning strategies can generate a labeled image.

One simple method is global thresholding. A threshold value is selected and all pixels with an intensity lower than this value are marked as background and all pixels with higher values are marked as foreground. Different strategies for determining the global thresholding value exist. A simplistic method, for example, models the intensities as a histogram with the assumption of two main intensity peaks (foreground and background), selecting a middle point as the threshold. A more sophisticated method is Otsu thresholding [36]. One major drawback of global thresholding algorithms is their inability to cope with images that have a variety of intensity. A latent print on carpet, for example, is often difficult to threshold with global thresholding since when the background is completely below the chosen threshold value, large portions of the shoeprint will also be missing.

The thresholding algorithm to determine whether a pixel is part of the foreground or background using contextual information from other pixels is based on *conditional random fields (CRFs)* [37]. A similar approach was used for an analogous problem in handwriting labeling [38]. The model exploits the inherent long range dependencies that exist in the latentprint and hence is more robust than approaches using neural networks and other binarization algorithms.

The probabilistic CRF model is given below.

$$P(\mathbf{y}|\mathbf{x}, \theta) = \frac{e^{\psi(\mathbf{y}, \mathbf{x}; \theta)}}{\sum_{\mathbf{y}'} e^{\psi(\mathbf{y}', \mathbf{x}; \theta)}} \quad (2.1)$$

where  $\mathbf{y}_i \in \{\text{Shoeprint, Background}\}$  and  $\mathbf{x}$  : Observed image and  $\theta$  : CRF model parameters. It is assumed that an image is segmented into  $3 \times 3$  non-overlapping patches. The patch size is chosen to be small enough for high resolution and big enough to extract enough features. Then

$$\psi(\mathbf{y}, \mathbf{x}; \theta) = \sum_{j=1}^m \left( A(j, y_j, \mathbf{x}; \theta^s) + \sum_{(j,k) \in E} I(j, k, y_j, y_k, \mathbf{x}; \theta^t) \right) \quad (2.2)$$

The first term in equation 2.2 is called the state term and it associates the characteristics of that patch with its corresponding label.  $\theta^s$  are called the state parameters for the CRF model. Analogous to it, the second term, captures the neighbor/contextual dependencies by associating pair wise interaction of the neighboring labels and the observed data.  $\theta^t$  are called

the transition parameters of the CRF model.  $E$  is a set of edges that identify the neighbors of a patch. A 24 neighborhood model was used.  $\theta$  comprises of the state parameters,  $\theta^s$  and the transition parameters,  $\theta^t$ .

The association potential can be modeled as

$$A(j, y_j, \mathbf{x}; \theta^s) = \sum_i (f_i \cdot \theta_{ij})$$

where  $f_i$  is the  $i^{th}$  state feature extracted for that patch and  $\theta_{li}$  is the state parameter. The state features,  $f_i$  are transformed by the  $\tanh$  function to give the feature vector  $\mathbf{h}$ . The transformed state feature vector can be thought analogous to the output at the hidden layer of a neural network. The state parameters  $\theta^s$  are a union of the two sets of parameters  $\theta^{s1}$  and  $\theta^{s2}$ .

The interaction potential  $I(\cdot)$  is generally the inner product between the transition parameters  $\theta^t$  and the transition features  $f_t$ . The interaction potential is defined as follows:

$$I(j, k, y_j, y_k, \mathbf{x}; \theta^t) = \sum_l (f^l(j, k, y_j, y_k, \mathbf{x}) \cdot \theta_l^t)$$

Features of a shoe-print might vary according to the crime scene. It could be powder on a carpet, mud on a table etc. So generalization of the texture of shoe-prints is difficult. So we resort to the user to provide the texture samples of the foreground and background from the image. The sample size is fixed to be  $15 \times 15$  which is big enough to extract information and small enough to cover the print region. There could be one or more samples of foreground and background. The feature vector of these samples are normalized image histograms. The two state features are the cosine similarity between the patch and the foreground sample feature vectors and the cosine similarity between the patch and the background sample feature vectors. Given normalized image histogram vectors of two patches the cosine similarity is given by

$$CS(P_1, P_2) = \frac{P_1 * P_2}{|P_1||P_2|} \quad (2.3)$$

The other two state features are entropy and standard deviation. Given the probability distribution of gray levels in the patch the entropy and standard deviation is given by

$$E(P) = - \sum_i^n p(x_i) * \log(p(x_i)) \quad (2.4)$$

$$STD(P) = \sqrt{\sum_i^n (x_i - \mu)^2} \quad (2.5)$$

The transition feature is the cosine similarity between the current patch and the surrounding 24 patches.

For the purpose of baseline comparison, pixels of the same images were labeled using Otsu

thresholding and neural network methods, with the results shown in Fig. 2.7. The CRF approach tends to outperform both by exploiting dependency between the current patch and its neighborhood.

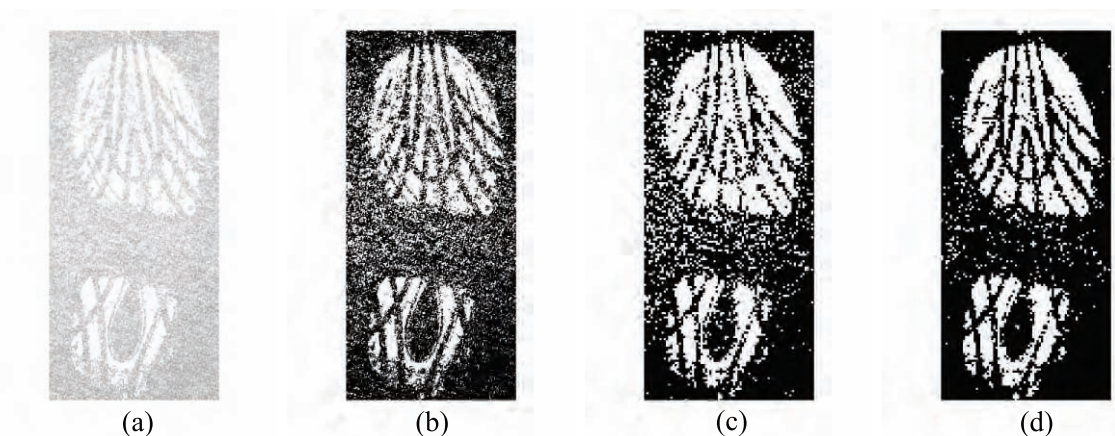


Figure 2.7: Results of image pixel labeling: (a) latent image, (b) Otsu thresholding, (c) neural network thresholding and (d) CRF segmentation.

A third method for separating foreground and background images is adaptive thresholding. In this method, a single threshold value is not selected for the entire image. Instead, the threshold value is dynamically determined throughout the image. This method has the advantage of being able to cope with larger changes in background, such as variations in background material (carpet, flooring, etc.) and lighting. Such images often lack the separation of peaks necessary to use global thresholding. Smaller sub-images are much more likely to be more uniform than the image overall. It selects the threshold value for each individual pixel based on the local neighborhood's range of pixel intensities. For some  $n$  pixels around a given pixel, the thresholding value is calculated via mean, median, mean-C, etc. and used to determine whether a single pixel is part of the foreground or background, with different selections of sampling giving different results. After tuning the method to shoe-prints, this method gives high quality results at reasonable resolution. Some sample images are shown in Figure 2.8.

### Edge detection

Rather than labeling pixels in the gray-scale image to convert to a binary image, an alternative is to use edge detection as the starting point. This has a firm basis in biological vision and has been studied extensively. Among various edge detectors the Canny edge detector [39] has been shown to have many useful properties. It is considered to be the most powerful edge detector since it uses a multi-stage algorithm consisting of noise reduction, gradient calculation, non-maximal suppression and edge linking with hysteresis thresholding. The detected edges preserve the most important geometric features on shoe outsoles, such as straight line segments, circles, ellipses. The results of applying the Canny edge operator to

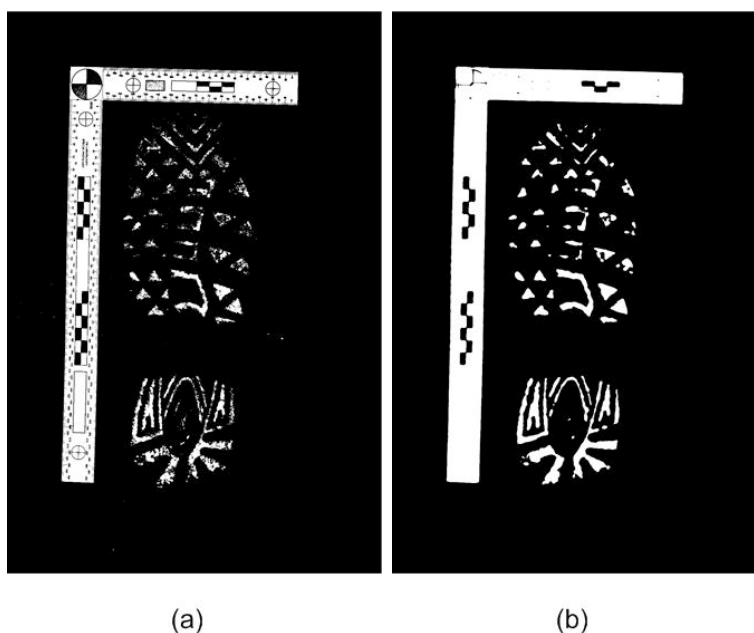


Figure 2.8: Adaptive thresholding results: (a) original impression (b) enhanced image using adaptive thresholding.

crime scene images is shown in Fig. 2.9. Results with some database images are shown in Fig. 2.10.

Prior to edge detection, morphological operations are performed on database images [40]. The morphological operations are: dilation, erosion and filling holes in the binary image<sup>5</sup>. The result is a more exact region boundary that improves the quality of edge detection. Morphological operations play a vital role in fetching the exact contours of the different shapes like line, ellipse and circle. We perform morphological operations (dilation and erosion) to make the interior region of the boundary uniform and then extract the boundary using Canny edge detection. Since the interior region is uniform, canny edge detector does not detect any edges inside the boundary and it improves the quality of edge detection. Specifically, each database shoe-print is processed in the following order: Edge detection  $\rightarrow$  Dilation  $\rightarrow$  Erosion  $\rightarrow$  Flood fill  $\rightarrow$  Complement. This procedure is illustrated using a sample print in the Fig. 2.11(a-f). As shown in Fig. 2.11(g), the edge image of the enhanced print has much better quality for feature extraction.

Dilation and erosion make the interior region of the boundary uniform and then extract the boundary using edge detection. Since the interior region is uniform the edge detector does not detect any edges inside the boundary. Edge detection showing the intermediate results of morphological operations is shown in Figure 2.12. Database Prints are subject to the sequence: Edge Detection, Morphological Operation and Edge Detection. Crime Scene Prints are subjected to only Edge Detection. For crime scene prints, because of their poor

<sup>5</sup><http://www.mathworks.com/access/helpdesk/help/toolbox/images/imfill.html><http://www.academictutorials.com/graphics/graphics-flood-fill.asp>



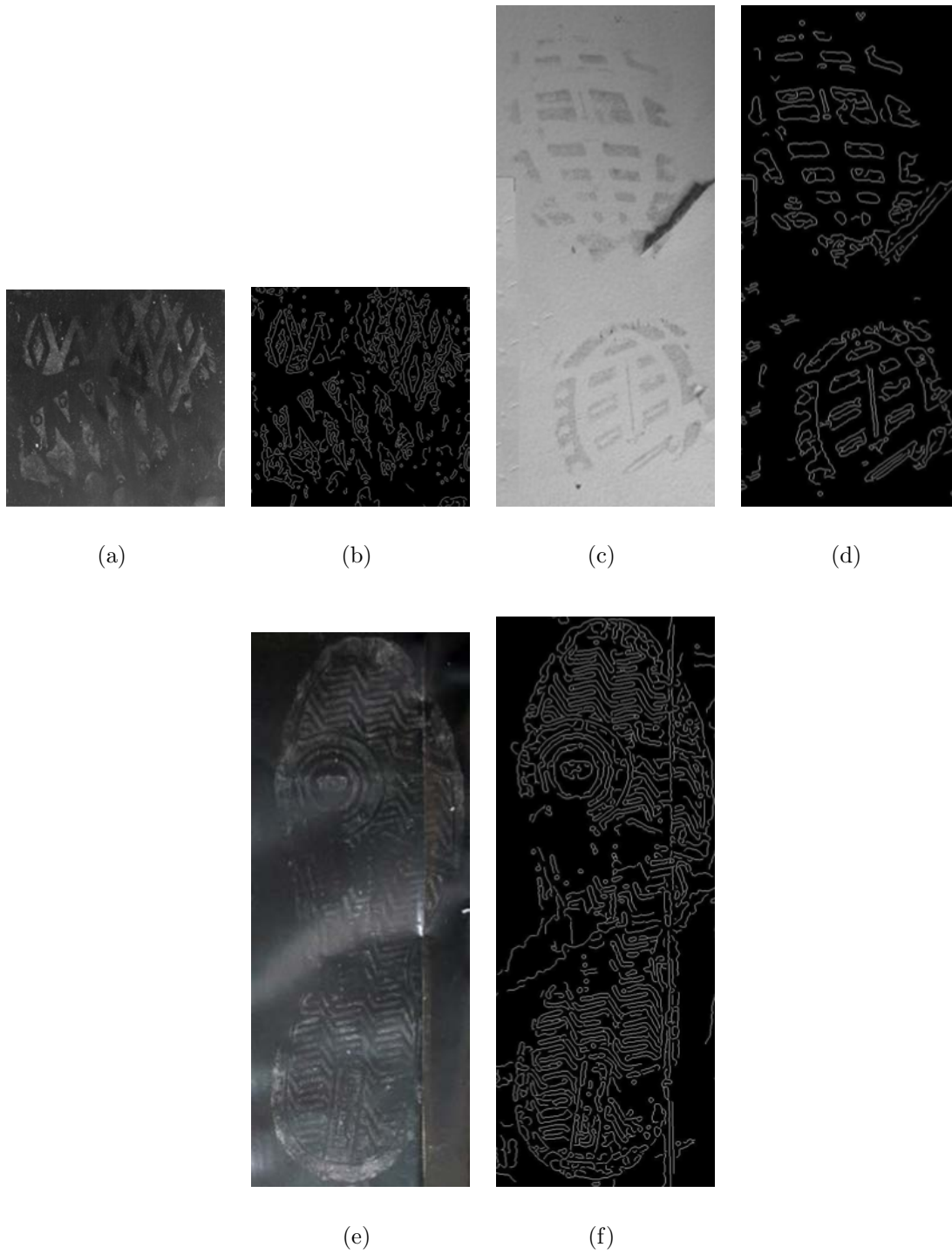


Figure 2.9: Results of edge detection on crime scene images: (a), (c) and (e) are originals, (b),(d) and (f) are corresponding edge images.

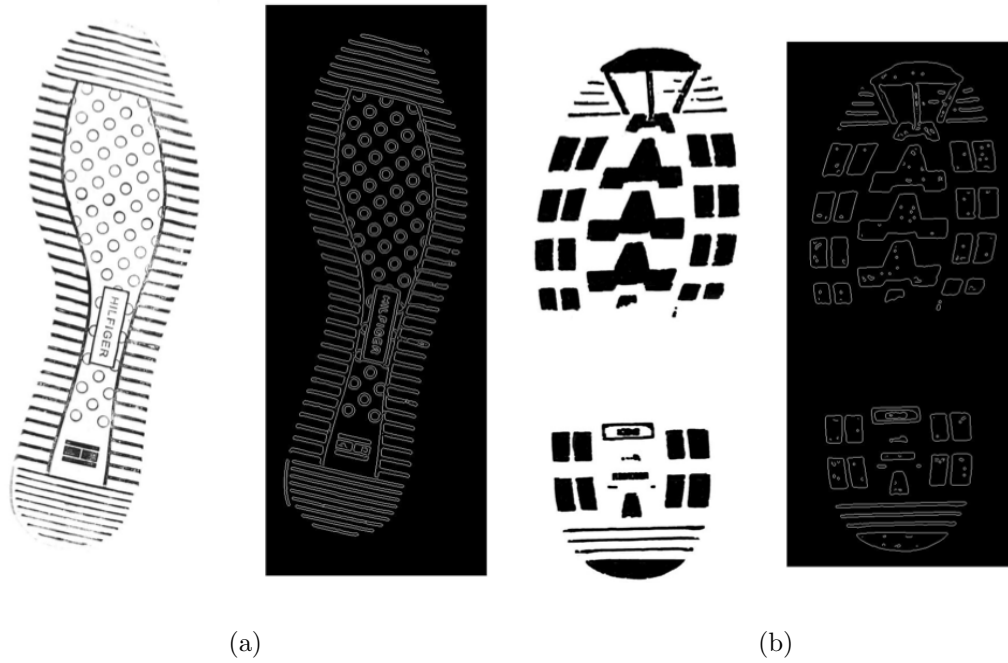


Figure 2.10: Results of edge detection on data base images.

quality, we directly extract features from the edge image of original image. It takes 4-5 seconds to process one image on a desktop computer.

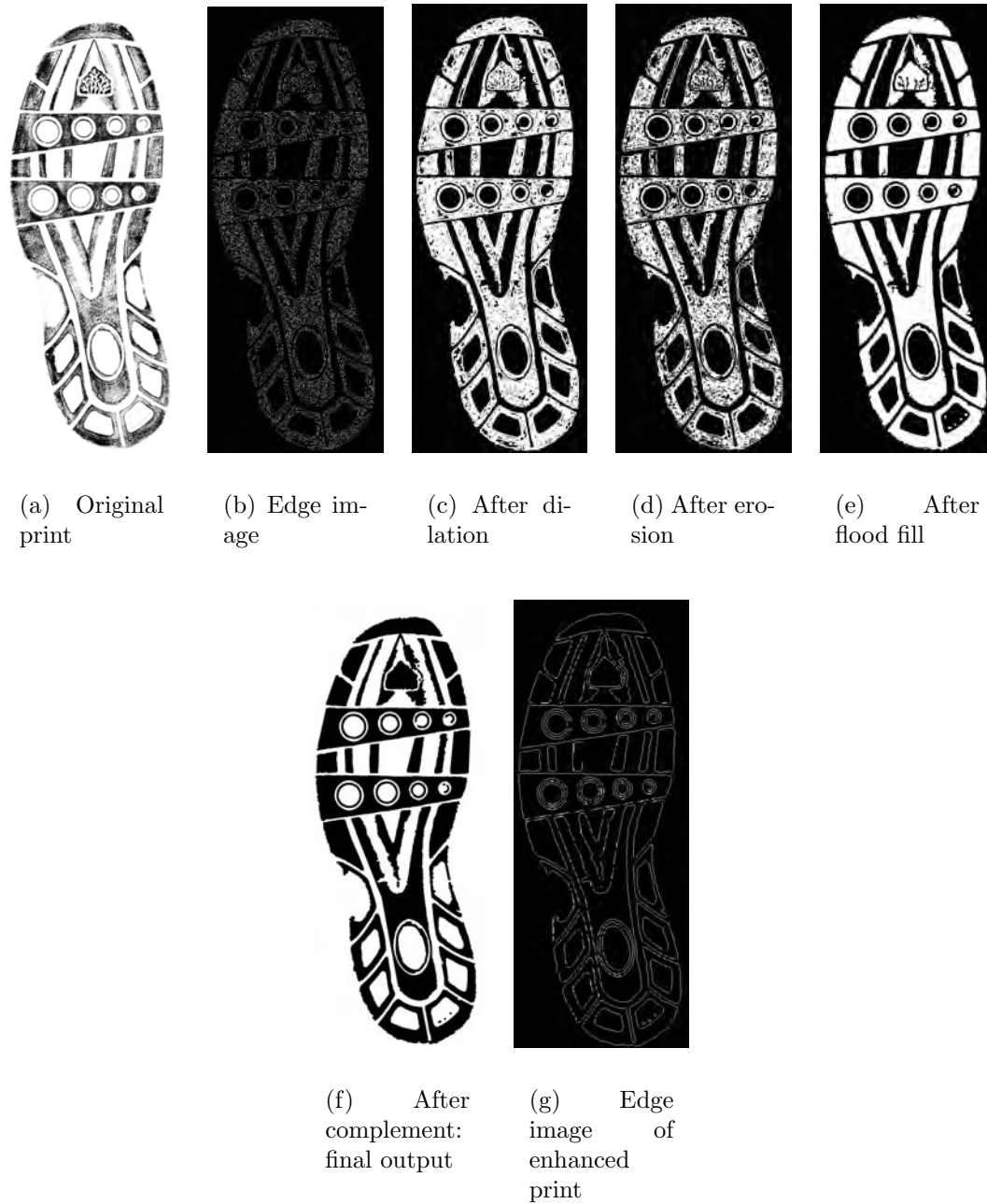
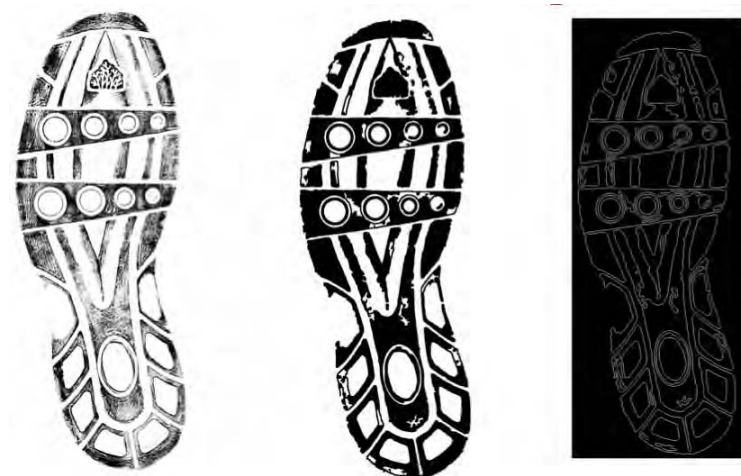


Figure 2.11: Morphological Operations for Shoe-print Enhancement.



(a)



(b)

Figure 2.12: Results of edge detection showing intermediate morphological operations on data base images.

### 2.2.4 Feature extraction

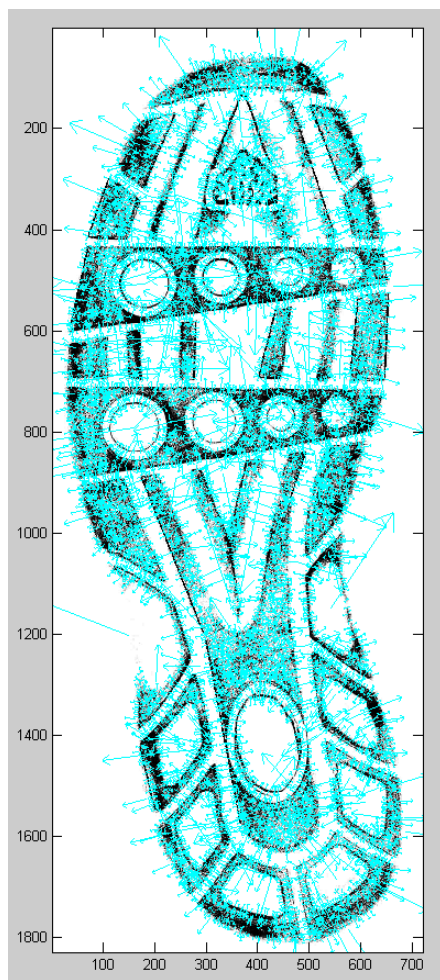
The extraction of suitable features for discrimination is the fundamental step of pattern recognition. The features extracted should discriminate between different outsoles but as well be invariant to various geometrical transformations. Once a set of features are determined there is also a need for a suitable measure of similarity between feature sets.

Color, texture and shapes of primitive elements can be used to distinguish images in general [41]. However color features are absent since acquired impression prints are gray-scale images. Textures are sensitive to acquisition methods and susceptible to wear while shapes are resistant to wear and present over a long period of time. Shape features are also robust against occlusion and incompleteness, i.e., the wear or variation of a local region on the outsole will be less likely to affect shape features in other regions.

Three different feature types, and associated similarity measurement methods, were tried for their suitability with footwear print images:

1. *GSC*: Features previously used with success in document analysis tasks of handwriting recognition and writer verification are GSC (gradient, structural, concavity) features which detect local, intermediate and global features (see Fig. 2.13) [42]. The basic unit of an image is the pixel and we are interested in its relationships to neighbors at different ranges from local to global. In a sense, we are taking a multi-resolution approach to feature generation. GSC features are generated at three ranges: local, intermediate and global. In the basic approach the feature vector consists of 512 bits corresponding to gradient (192 bits), structural (192 bits), and concavity (128 bits) features. Each of these three sets of features rely on dividing the scanned image into a  $4 \times 4$  region. Gradient features capture the frequency of the direction of the gradient, as obtained by convolving the image with a Sobel edge operator, in each of 12 directions and then thresholding the resultant values to yield a 192-bit vector. The structural features capture, in the gradient image, the presence of corners, diagonal lines, and vertical and horizontal lines, as determined by 12 rules. Concavity features capture, in the binary image, major topological and geometrical features including direction of bays, presence of holes, and large vertical and horizontal patterns. The input shoe-print is represented as two  $4 \times 4$  regions or a fixed-dimensional (1028-bit) binary feature vector. The similarity between two GSC feature vectors is computed using a correlation measure.
2. *SIFT*: A commonly used feature known as the scale invariant feature transform, or SIFT (Scale Invariant Feature Transform) [43]. It is an algorithm to extract and describe invariant features from images that can be used to perform reliable matching between different views of an object or scene. It consists of four major steps including scale-space extrema detection, key point localization, orientation assignment and keypoint descriptor construction. Specifically, in the scale space constructed by convolving the input image with a Gaussian function and resampling the smoothed image, maxima and minima are determined by comparing each pixel in the pyramid to its 26 neighbors (in a  $3 \times 3$  cube). These maxima and minima in the scale space are called as key points, which are in turn described by a 128-dimensional vector: a normalized





(a)

0.0158	0.2521	0.0433	0.0217	0.0276	0.0197	0.0473	0.0039
0.1260	0.2599	0.0315	0.0000	0.0000	0.0020	0.0039	0.0000
0.0551	0.0768	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0177	0.0256	0.0276	0.0729	0.0709	0.0886	0.1339	0.0158
0.2599	0.1733	0.0079	0.0217	0.0256	0.0236	0.0138	0.0729
0.2599	0.0985	0.0236	0.0000	0.0000	0.0000	0.0000	0.0138
0.0000	0.0039	0.0020	0.0000	0.0000	0.0000	0.0000	0.0000
0.0098	0.0059	0.0256	0.0748	0.0591	0.1635	0.1201	0.0394
0.2481	0.0670	0.1024	0.0748	0.0295	0.0118	0.0374	0.1674
0.0729	0.2599	0.2599	0.0650	0.0236	0.0000	0.0000	0.0059
0.0059	0.1871	0.0473	0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0197	0.0236	0.0748	0.2599	0.0906	0.0354	0.0020
0.0059	0.0886	0.1024	0.1182	0.0985	0.0039	0.0059	0.0098
0.0394	0.1753	0.1024	0.0926	0.1359	0.0236	0.0197	0.0197
0.0610	0.2599	0.0118	0.0000	0.0000	0.0000	0.0039	0.0079

(b)

Figure 2.14: SIFT Representation: (a) key points where each blue arrow shows key point orientation, and (b) descriptor for one key point.



Table 2.1: Distribution of geometric patterns in footwear outsole prints.

<b>Fundamental Patterns</b>	<b>No. of Prints</b>
Line segments	3397
Lines & Circles	812
Lines & Ellipses	285
Only Circles/Arcs	73
Lines, Circles & Ellipses	37
Only Ellipses	15
Circles & Ellipses	5
Texture	410
<b>Total - 5034 prints</b>	

Features based on primitive shapes worked better than SIFT in retrieval as is described later in this report (see Fig. 2.42).

### 2.2.5 Geometrical patterns

Patterns of outsoles usually contain small geometrical patterns involving short straight line segments, circles and ellipses. An analysis of 5,034 outsole prints revealed that 67% have only line segments (some examples are shown in Fig. 2.15, where the line segments have a minimum length of 25 pixels), 1.5% have only circles (Fig. 2.16), 0.004% have only ellipses (Fig. 2.17), and 24% are combinations of lines, circles and ellipses. The principal combination of shapes are lines-circles which constitute 16% (Fig. 2.18), lines-ellipses constitute 6% (Fig. 2.19), circles-ellipses-0.1% (Fig. 2.20) and lines-circles-ellipses-0.7% (Fig. 2.21). Texture patterns (Fig. 2.22) constitute the remaining 8%. The complete distribution is given in Table 2.1. This analysis shows that the three basic shapes are present in 92% of outsole prints. Furthermore, patterns other than circles and ellipses can be approximated by piecewise lines.

In fact when projected on to a plane, most man-made objects can be represented as combinations of straight line and ellipse segments. Mathematically, straight line segments and circles are special cases of ellipses. An ellipse with zero eccentricity is a circle and an ellipse with eccentricity of 1 is a straight line; where the eccentricity of an ellipse is defined as  $\sqrt{1 - (b/a)^2}$  where  $a$  and  $b$  are the lengths of the semi-major and semi-minor axes.

While an ellipse detector alone can capture 92% of the primitive shapes, we choose to use specialized detectors for straight lines and circles since they are more efficient. The feature extraction approach is to detect the presence, location and size of three basic shapes: *straight line segments*, *circles/arcs* and *ellipses*. Since all three are geometrical shapes with simple parametric representations, they are ideal for the application of a robust method of detecting shapes.

The Hough transform[44] is a method to automatically detect basic geometrical patterns in noisy images. It detects features of a parametric form in an image by mapping foreground pixels into parameter space, which is characterized by an  $n$  dimensional accumulator array, where  $n$  is the number of parameters necessary to describe the shape of interest. Each



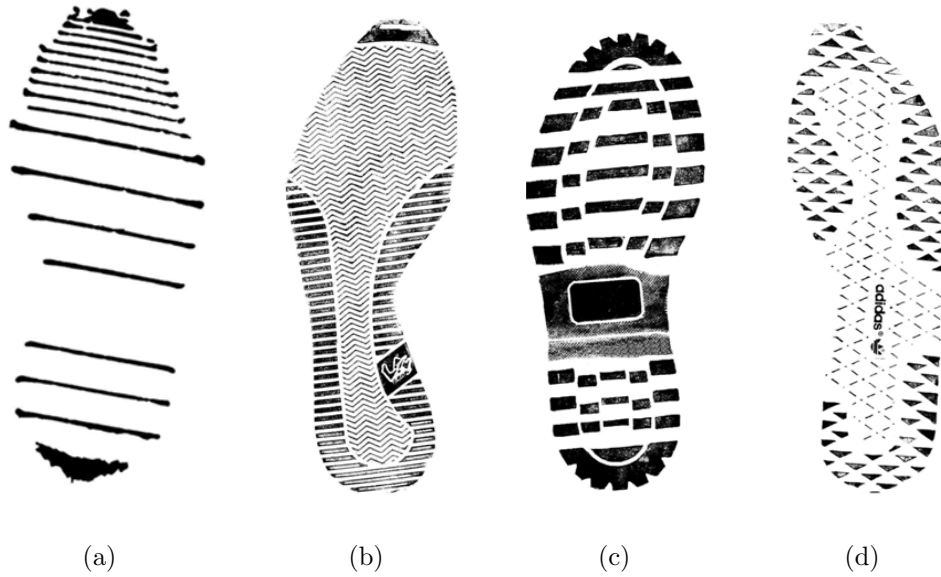


Figure 2.15: Footwear outsole patterns containing line segments only.

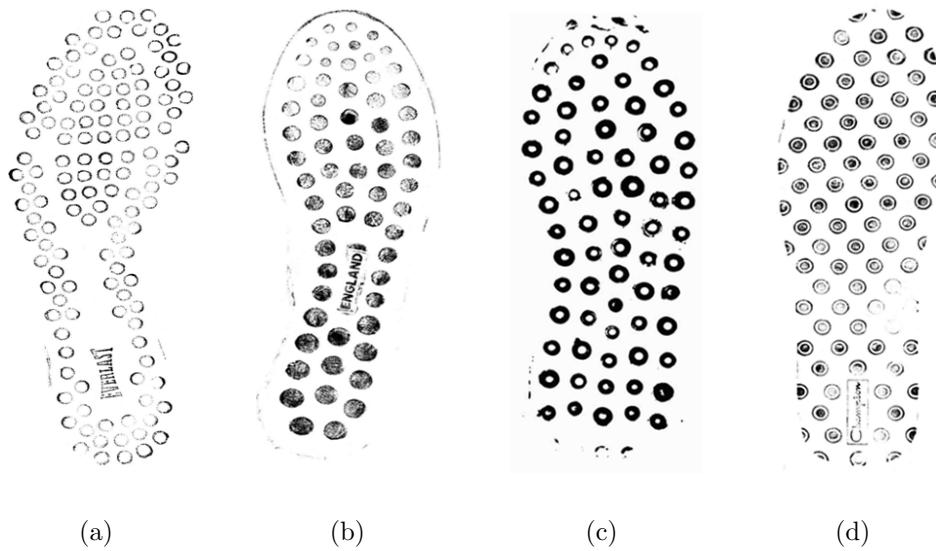


Figure 2.16: Footwear outsole patterns containing circles only.

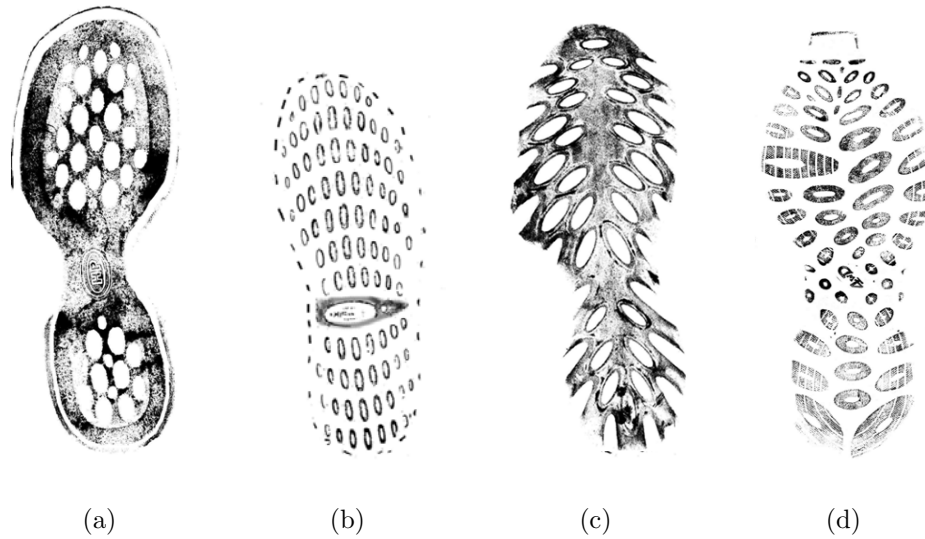


Figure 2.17: Footwear outsole patterns containing ellipses only.

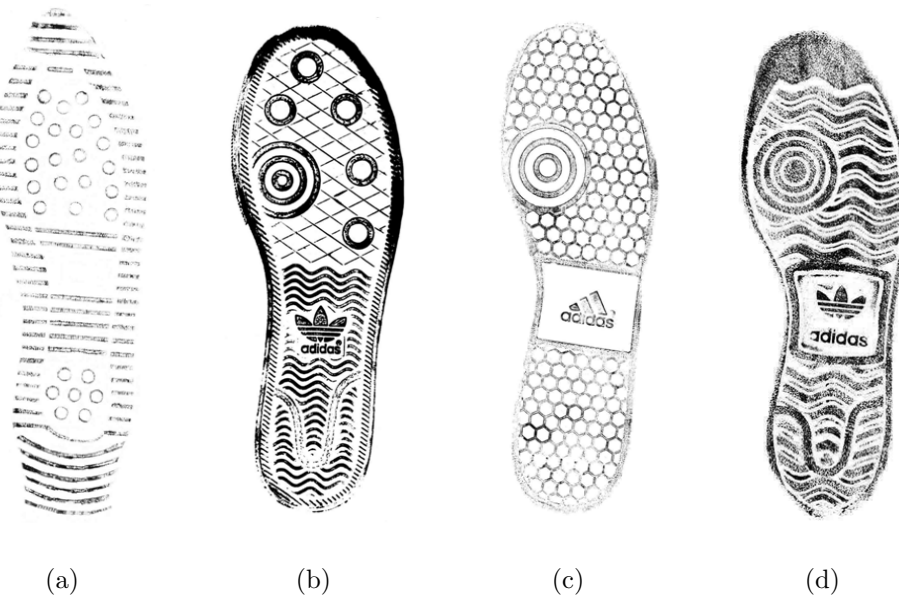


Figure 2.18: Footwear outsole patterns containing lines and circles.

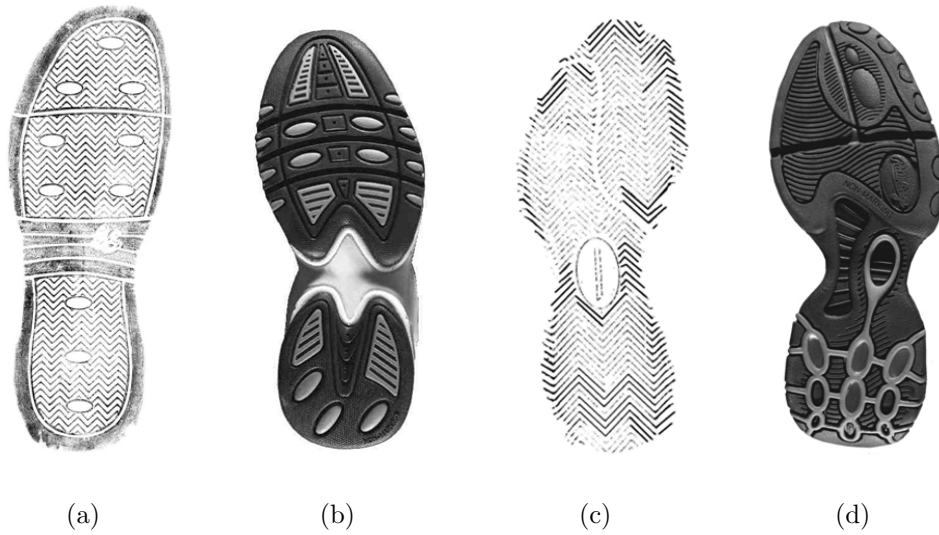


Figure 2.19: Footwear outsole patterns containing lines and ellipses.

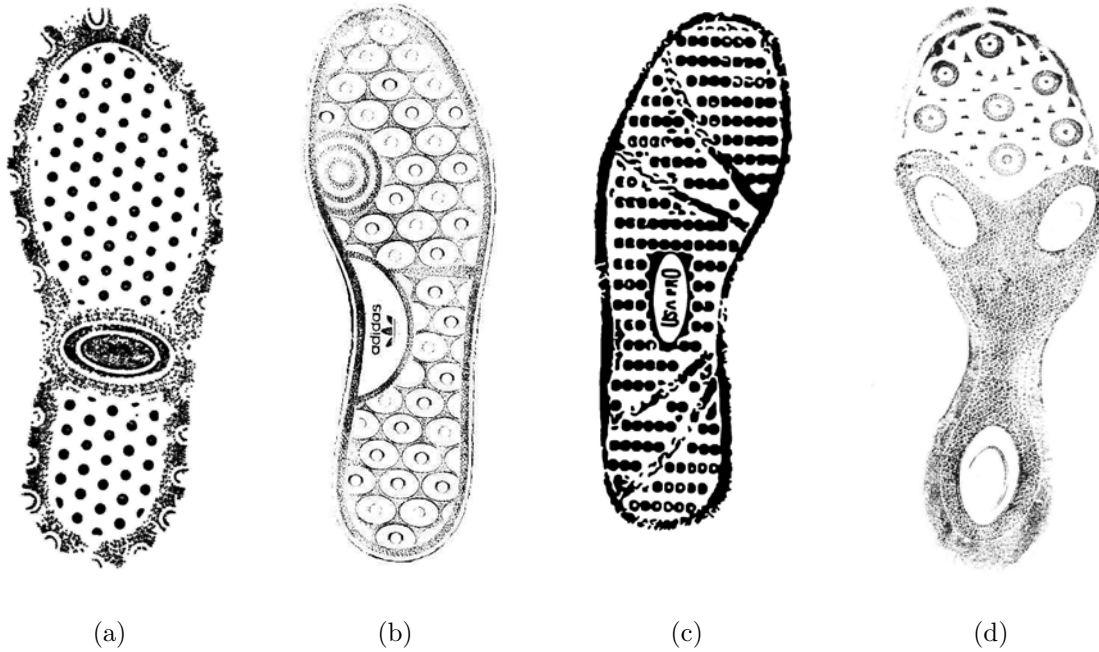


Figure 2.20: Footwear outsole patterns containing circles and ellipses.

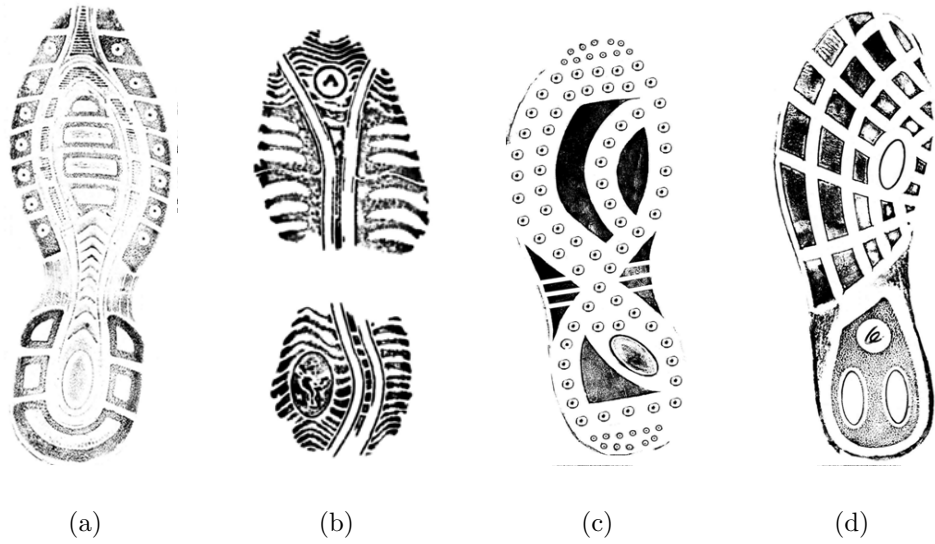


Figure 2.21: Footwear outsole patterns containing lines, circles and ellipses.

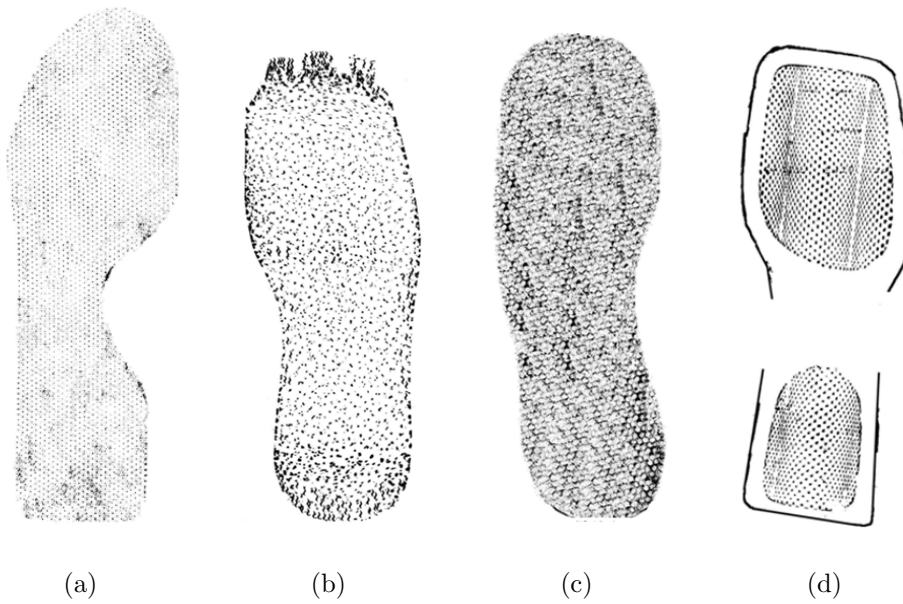


Figure 2.22: Footwear outsole patterns containing texture only.

significant pixel from the shape of interest would cast a vote in the same cell of an accumulator array, hence all pixels of a shape gets accumulated as a peak. The number of peaks corresponds to the number of shapes of interest in the image.

The Hough transform was originally designed for detecting straight lines in cloud chamber photographs. Later it was generalized to circles and ellipses. It has found success in many applications such as detecting cancerous nodules in radiological images and structure of textual lines in document images[45].

1. *Line Segments:* Using the polar coordinate system, a straight line can be represented by two parameters  $r$  and  $\theta$ . The Hough transform maps each pixel in the Cartesian  $x$ - $y$  plane to a 2-dimensional accumulator array using the transformations defined by  $x = r\cos\theta$  and  $y = r\sin\theta$ . The values of  $r$  and  $\theta$  at which the accumulator elements peak represent the presence of straight lines.
2. *Circles:* It involves building a 3-dimension accumulator array corresponding the center coordinates and the radius. Gradient orientation is used to limit the generation of spurious votes. Further, spatial constraints are used to identify spurious circles. Gradient orientation is used to limit the generation of spurious votes[46]. Further, spatial constraints are used to eliminate spurious circles.
3. *Ellipses* In a Cartesian plane, an ellipse can be described by its centre  $(p, q)$ , length of the semi-major axis  $a$ , length of the semi-minor axis  $b$  and the angle  $\theta$  between the major axis and the  $x$ -axis. Thus five parameters  $(p, q, a, b, \theta)$  are required to uniquely describe an ellipse[47]. These five parameters demand a five-dimensional accumulator which is computationally expensive but the Randomized Hough transform (RHT) [48] for ellipse detection, described next, is more efficient.

**Algorithm RHT:** Randomized Hough Transform

- (a) Pick three foreground pixels  $p_1, p_2$  and  $p_3$  randomly and fit a tangent at each of the picked point, namely  $t_1, t_2$  and  $t_3$
- (b) Find intersection of the tangent pairs  $(t_1, t_2)$ , and  $(t_2, t_3)$
- (c) Find straight line passing through midpoint of pixels  $p_1$  and  $p_2$  and the intersection of their tangents. Repeat the same step with pixels  $p_2$  and  $p_3$ . The intersection of the two lines gives the centre of the ellipse
- (d) Shift ellipse center to the origin to get rid of parameters  $w_4$  and  $w_5$  in conic equation

$$w_1x^2 + w_2xy + w_3y^2 + w_4x + w_5y + w_6 = 0 \quad (2.6)$$

- (e) Find coefficients  $w_1, w_2$  and  $w_3$  in conic equation by substituting the co-ordinates of the three picked points and by solving the system of linear equations.

The RHT cannot be used directly for ellipse detection in outsole print images. This is because there are around 50,000 foreground pixels in a print of typical size  $600 \times 800$  and picking three random foreground pixels will never narrow down to the right ellipse.

The solution proposed is to determine connected components in the edge image. To reduce computation certain connected components are eliminated based on the eccentricity property of the region enclosed by each component. Within a connected component three pixels are randomly selected and bad ones eliminated using gradient information. Then each connected component is scanned for an ellipse using RHT. The fraction of foreground pixels that satisfies the ellipse equation are determined. Spurious pixels are eliminated by comparing gradient direction and orientation. The complete algorithm follows.

**Algorithm ED:** Ellipse Detection

**Input:** Edge image after removing all the on-circle pixels  $BW$  and original image  $I$

**Output:** Detected ellipses and their parameters

- (a) Compute gradient orientation of  $I$
- (b) Find the connected components and their eccentricities  $e$  in  $BW$
- (c) Eliminate the connected components with  $e < 0.3$  or  $size < 20$  pixels (noise)
  - for each** connected component **do**
    - i. Randomly pick three pixels
    - ii. Compute the standard deviation of gradient orientation at each pixel's  $7 \times 7$  neighbor and get  $s_1$ ,  $s_2$  and  $s_3$
    - iii. **if** ( $s_1 \in [minS, maxS]$ ) & ( $s_2 \in [minS, maxS]$ ) & ( $s_3 \in [minS, maxS]$ ) **then**
      - A. Apply RHT and find parameters  $(p, q, a, b, \theta)$  of the ellipse
      - B. Find candidate foreground pixels that satisfy ellipse equation
 
$$\frac{((x-p) \cos \theta + (y-q) \sin \theta)^2}{a^2} + \frac{((y-q) \cos \theta + (x-p) \sin \theta)^2}{b^2} = 1 \quad (2.7)$$
      - C. Find analytical derivative  $D$  at each candidate pixel using
 
$$\frac{(\frac{-2}{a^2})(x-p) \cos \theta \sin \theta + (y-q) \sin^2 \theta + (\frac{2}{b^2})(y-q) \cos^2 \theta - (x-p) \cos \theta \sin \theta}{(\frac{-2}{a^2})(x-p) \cos^2 \theta + (y-q) \cos \theta \sin \theta + (\frac{2}{b^2})(y-q) \cos \theta \sin \theta - (x-p) \sin^2 \theta} \quad (2.8)$$
      - D. If difference between  $D$  and tangent of gradient orientation is below threshold  $T_1$ , declare it as a true ellipse pixel
      - E. If ratio of number of true ellipse pixels to circumference of ellipse is above threshold  $T_2$ , declare component as ellipse
    - end if**
  - end for**

The fraction of true ellipse pixels to ellipse perimeter is a measure of ellipse quality. Ellipses detected before and after elimination of spurious ones, in Steps D-E, are shown in Figure 2.23.

Final results of extracting circles, ellipses and straight line segments in both crime scene and data base prints are shown in Figures 2.24 and 2.25 respectively.

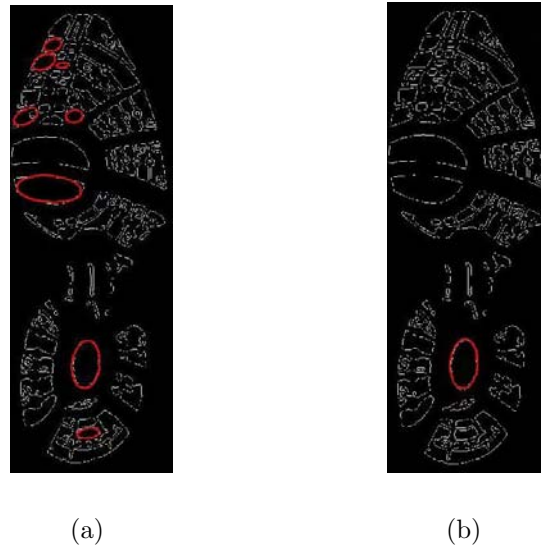


Figure 2.23: Elimination of spurious ellipses using gradient orientation. (a) before elimination, and (b) after elimination.

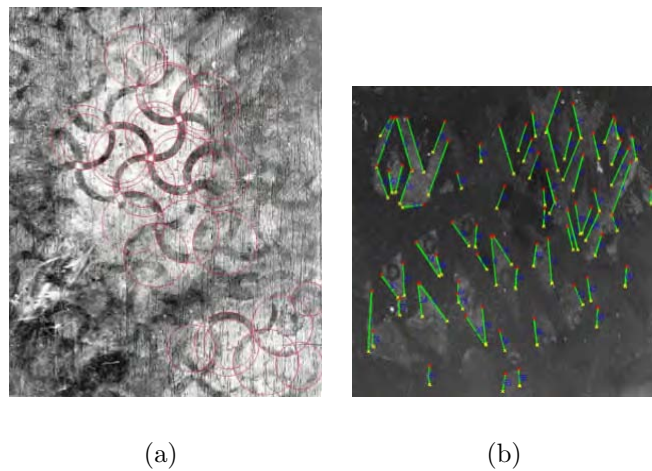


Figure 2.24: Shapes automatically detected in crime scenes: (a) print where circles (red) are prominent, and (b) print where line segments (green) are prominent.

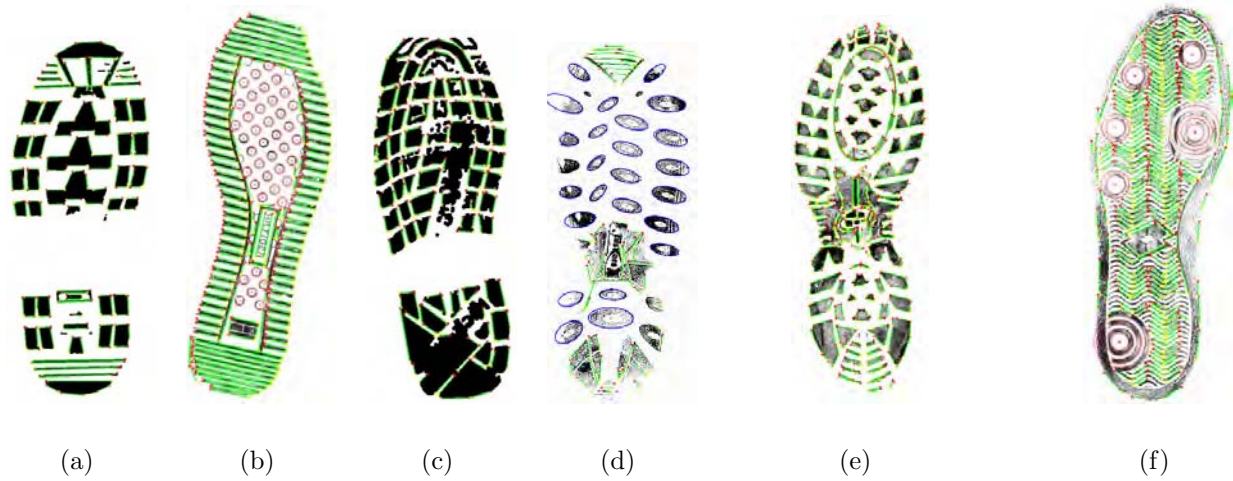


Figure 2.25: Shapes detected in database prints: lines, circles and ellipses are shown in green, red and blue respectively.



## 2.2.6 Graph representation

Structural representations have long been used in computer vision to represent complex objects and scenes for image matching [49]. Graph representations have a great advantage over feature vectors because of they can explicitly model the relationship between different parts and feature points [50].

After detecting their presence, the impression image is decomposed into a set of primitives. To obtain a structural representation of these primitives, an *attributed relational graph* (*ARG*) [51, 52] is built. An *ARG* is a directed graph that can be represented as a 3-tuple  $(V, E, A)$  where  $V$  is the set of vertices, also called nodes,  $E$  is the set of relations (edges) and  $A$  is the set of attributes. Each edge describes the spatial relationship between a pair of nodes. The attributes include node attributes (unary) and edge attributes (binary).

There are three types of nodes, corresponding to lines (L), circles (C) and ellipses (E), and nine types of edges: line-to-line (L2L), line-to-circle (L2C), line-to-ellipse (L2E), circle-to-circle (C2C), circle-to-ellipse (C2E), ellipse-to-ellipse (E2E), circle-to-line (C2L), ellipse-to-line (E2L) and ellipse-to-circle (E2C). Attributes of nodes and edges should be defined such that they are scale/rotation invariant, and capture spatial relationships such as distance, relative position, relative dimension and orientation.

### Attributes of vertices

Three attributes are defined for nodes which represent the basic shapes detected<sup>6</sup>.

1. *Quality* is the ratio of the number of points on the boundary of the shape (perimeter pixels) to the perimeter of the shape.
2. *Completeness* is the standard deviation of the angles of all on-perimeter pixels with respect to the center of circle/ellipse, *std*, normalized as *std*/180. If a wide range of angles are present, implying that most of the shape is represented, there will be more angles represented and this value is high, while a partial figure will have smaller diversity of angles and this value will be low. While the range of angles is 0 to 360 for circles and ellipses, for a straight line there are only two angles with respect to the center, 0 and 180.
3. *Eccentricity* is the degree of elongation, defined as the square root of 1 minus square of ratio of minor to major axes. For a circle eccentricity is 0 and for a straight line eccentricity is 1.

Node attributes are summarized in Table 2.2.

### Attributes of relations

Edge attributes are dependent upon the pair of shapes they connect. They use the relative position definitions between lines, circles and ellipses shown in Fig. 2.27. Some attributes with value  $x$  are normalized to the range  $[0,1]$  by using the sigmoid function

---

<sup>6</sup>In the implementations described later, node attributes were implemented for circles/ellipses but not for straight lines.

Table 2.2: Node Attribute Definitions

<b>Attributes</b>	<b>Definition</b>
<i>Eccentricity</i>	$\sqrt{1 - \frac{b^2}{a^2}}$ (see definitions of $a, b$ below)
<i>Quality</i>	Ratio of no. of on-shape pixels to circumference (equal to the length for lines)
<i>Completeness</i>	Std. Dev. of angle that on-shape pixels make with center

(a) Node Attributes of Circle, Ellipse and Line

<b>Symbols</b>	<b>Definition</b>	<b>Symbols</b>	<b>Definition</b>
$L$	Line segment	$len$	Length
$C$	Circle	$N$	Normalized
$E$	Ellipse	$m$	mid-point of a line segment
$cen$	center	$e$	eccentricity
$d_E$	Euclidean distance	$ \cdot $	absolute value
$rd$	Relative distance	$a$	Semi-major axis of ellipse
$ro$	Relative orientation	$b$	Semi-minor axis of ellipse
$rs$	Relative size	$ER$	Geometric mean: $\sqrt{a \times b}$
$\theta$	Orientation	$max$	maximum
$r$	Radius	$min$	minimum

(b) Symbol definitions

$$s(x) = x/\sqrt{1+x^2} \quad (2.9)$$

which is also depicted in Figure 2.26.

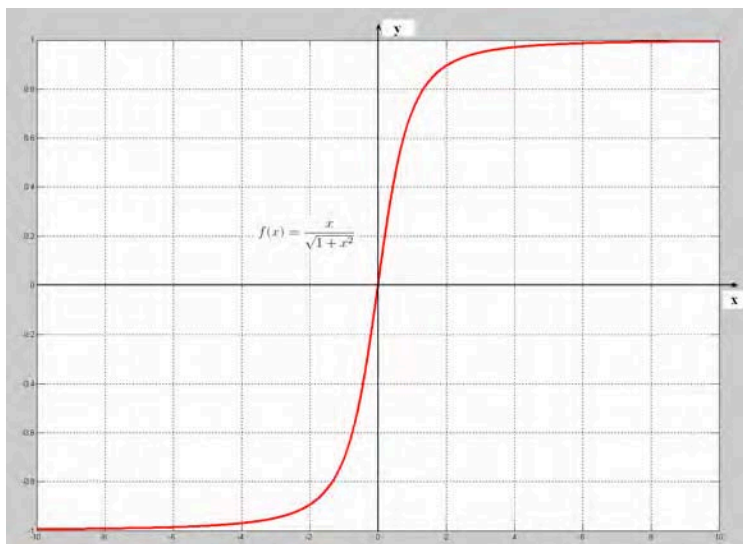


Figure 2.26: Normalization of some attributes is done using the squashing function  $f(x) = x/\sqrt{1+x^2}$ . When only positive vales of  $x$  are input, the attribute is normalized to have values in the interval  $[0,1]$ .

1. *Line-to-Line (L2L)*: There are five attributes for a pair of straight line segments  $L_1$  and  $L_2$ : (i) *normalized relative angle*  $N - \alpha$ : acute angle between  $L_2$  and  $L_1$ , normalized as  $\alpha/180$  (ii) *normalized relative size*  $N - rs$ : the length of  $L_1$  divided by that of  $L_2$ , normalized as  $len(L_1)/(len(L_1) + len(L_2))$ , (iii) *relative distance*  $rd$ : defined as the length of the line segment connecting the mid-points  $M_1, M_2$  of these two lines, divided by the length of  $L_2$ , normalized as  $len(M_1M_2)/(len(M_1M_2) + len(L_2))$ . (iv) *perpendicular distance*  $pd$ : distance of mid-point of  $L_1$  to  $L_2$  divided by the sum of their lengths. (v) *relative position*  $rp_1, rp_2$ , which describes how the two lines are connected or to be connected. If the point where they meet is located at  $O$  on both lines, this attribute is calculated by dividing the shorter one of the two sub-segments  $OA$  and  $OB$  by the longer one of them. If the point of intersection lies outside of the two line segments, it is calculated by dividing the length of  $OB$  by minus length of  $OA$ . This attribute is normalized as  $(rp + 1)/2$ .
2. *Circle-to-Circle (C2C)*: There are three attributes for a pair of circles  $C_1$  and  $C_2$ : (i) *Normalized relative size*  $N - rs$ : : the radius of  $C_1$  divided by that of  $C_2$ , normalized as  $r(C_1)/(r(C_1) + r(C_2))$  (ii) *relative distance*  $rd_1$ : the distance between two centers divided by the sum of the radiuses of  $C_1$  and  $C_2$ , normalized as  $s(rd_1)$  using Eq. 2.9, (iii) *relative distance*  $rd_2$ : the distance between two centers divided by the difference between the radiuses of  $C_1$  and  $C_2$ , normalized as  $s(rd_2)$ .

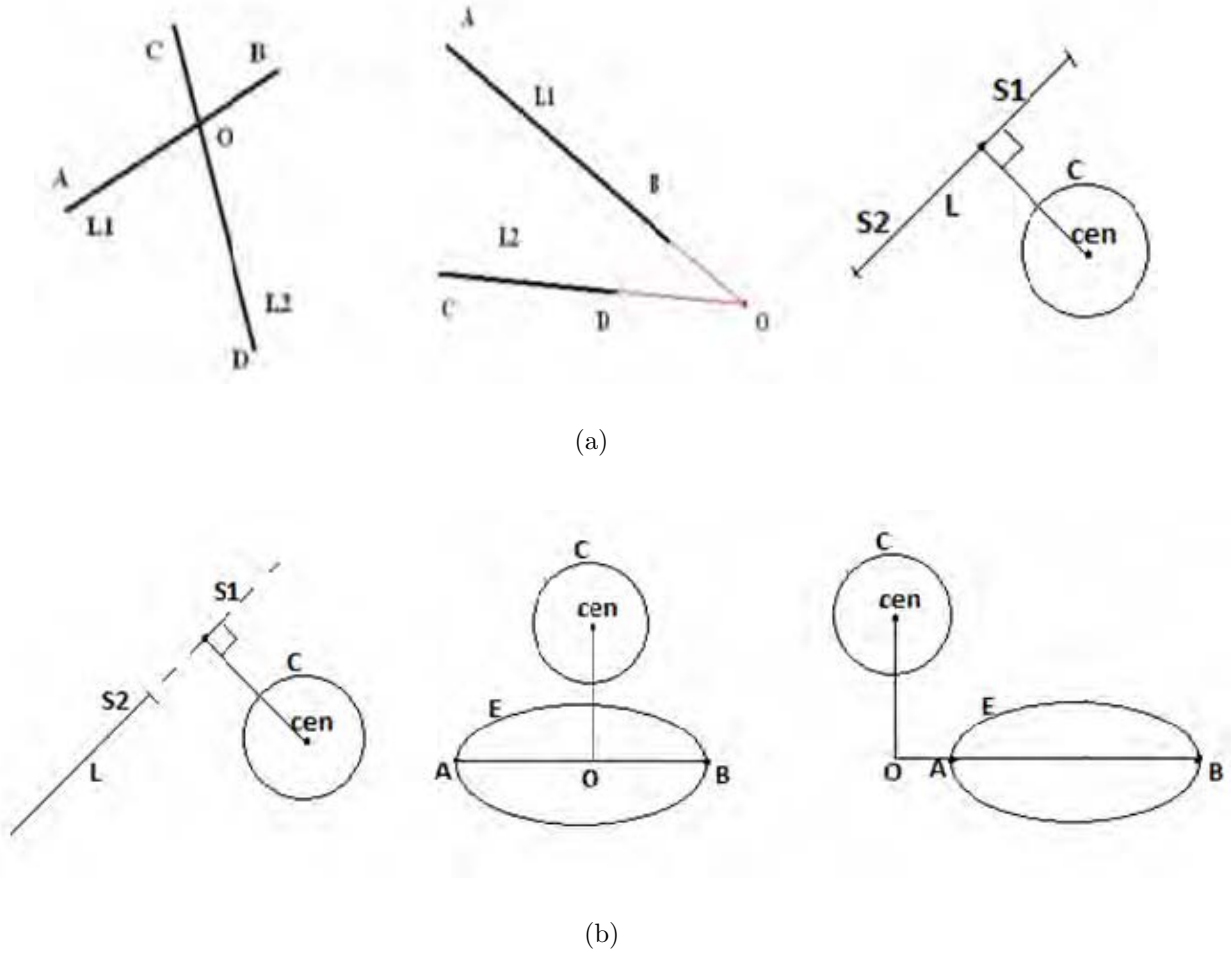


Figure 2.27: Relative position definitions for line-line, line-circle and circle-ellipse.

3. *Line-to-Circle (L2C)*: There are three attributes between a line segment  $L$  and circle  $C$ : (i) *Normalized relative size  $N - rs$* : the length of  $L$  divided by the radius of  $C$ , normalized as  $len(L)/(len(L) + r(C))$ , (ii) *relative distance  $rd$* : the distance from the center of  $C$  onto  $L$  or its extension divided by the radius of  $C$ , normalized as  $s(rd)$  (iii) *relative position  $rp$* : the perpendicular line passing through the center of  $C$  intersects  $L$  at some point  $P$ .  $P$  divides  $L$  into two sub-segments  $S_1$  and  $S_2$ . The relative position is defined as the ratio between  $min(S_1, S_2)$  and  $max(S_1, S_2)$ .
4. *Line-to-Ellipse (L2E) Attributes*: There are four attributes between a straight line segment  $L$  and ellipse  $E$ : (i) *Normalized relative size  $N - rs$* :  $len(L)$  divided by the sum of equivalent radius (ER) of ellipse (geometric mean of lengths of semi-axes) and  $len(L)$ , (ii) *relative distance  $rd$* : ratio of distance between center of ellipse and mid-point of line to sum of ER and  $len(L)$ , (iii) *relative position  $rp$* : relative position of major axis of ellipse and the line, (iv) *Normalized relative orientation  $N - ro$* : relative angle of major axis of ellipse and the line.
5. *Circle-to-Line (C2L) Attributes*: There are two attributes between a circle  $C$  and line segment  $L$ : (i) *relative size  $rs$* : the ratio of radius of  $C$  and the length of  $L$ , normalized as  $C.radius/(L.length + C.radius)$  (ii) *relative distance  $rd$* : distance from the center of  $C$  onto  $L$  or its extension divided by the radius of  $C$ , normalized as  $s(rd)$ .
6. *Circle-to-Ellipse (C2E) Attributes*: There are three attributes similar to that for Line-to-Ellipse.
7. *Ellipse-to-Ellipse (E2E) Attributes*: There are six attributes between ellipses  $E_1, E_2$ : (i) *eccentricity ratio  $er$* : ratio of eccentricity of  $E_1$  to sum of their eccentricities, (ii) *eccentricity difference  $ed$* : normalized difference of eccentricities, (iii) *relative distance  $rd$*  is similar to relative distance of circle to ellipse, (iv) *Normalized relative size  $N - rs$* : is similar to circle-to-circle, (v) *Normalized relative orientation  $N - ro$*  is the absolute difference between the orientations divided by 90, (vi) *relative position  $rp$* : relative position of the two major axes.

The cases of ellipse-to-line (E2L) and ellipse-to-circle (E2C) are defined similar to the above. All attributes are normalized to the interval [0.1]. The edge attribute definition are summarized in Tables 2.3 and 2.4; the weights shown for each attribute are defined using sensitivity analysis described in Section 2.2.8.

### Graph connectivity

So as to handle missing nodes or incorrectly detected nodes, which may arise due to noise, occlusion and incompleteness, a *fully-connected graph* is used. If for the sake of computational efficiency we consider only local relationships, as is often done in Markov models, it would lead to poor results since the only image components discernible in a print may be those at the extremities.

This means that there is a directed edge from each node to all nodes including itself; a node is connected to itself because we can use a general formula for computing the cost

Table 2.3: Edge Attribute Definitions

Attribute	Symbol	Definition	Normalizn	Weight
Normalized relative angle	$N - \alpha$	$\frac{ L1.\theta - L2.\theta }{180}$	already normalized	0.4472
Normalized relative size	$N - rs$	$\frac{L1.len}{L1.len + L2.len}$	already normalized	0.4472
Relative distance	$rd$	$\frac{d_E(L1.m, L2.m)}{L1.len + L2.len}$	$s(rd)$	0.4472
Perpendicular distance	$pd$	$\frac{d_E(L1.m, L2)}{L1.len + L2.len}$	$s(pd)$	0.4472
Relative position	$rp_1$	$\frac{\min(OA, OB)}{\max(OA, OB)}$	$\frac{rp_1 + 1}{2}$	0.4472
Relative position	$rp_2$	$-\frac{\min( OA ,  OB )}{\max( OA ,  OB )}$	$\frac{rp_2 + 1}{2}$	0.4472

(a) Line-to-Line (L2L)

Symbol	Definition	Normalizn	Weight
$N - rs$	$\frac{C1.r}{C1.r + C2.r}$	-	0.7071
$rd_1$	$\frac{d_E(C1.cen, C2.cen)}{C1.r + C2.r}$	$s(rd_1)$	0.7071
$rd_2$	$\frac{d_E(C1.cen, C2.cen)}{\max( C1.r - C2.r , 10^{-3})}$	$s(rd_2)$	0.0

(b) Circle-to-Circle (C2C)

Symbol	Definition	Normalizn	Weight
$N - rs$	$\frac{L.len}{C.r + L.len}$	-	0.5774
$rd$	$\frac{d_E(C.cen, L)}{C.r}$	$s(rd)$	0.5774
$rp$	$\frac{\min(S1, S2)}{\max(S1, S2)}$	$\frac{rp + 1}{2}$	0.5774

(c) Line-to-Circle (L2C)

Symbol	Definition	Normalizn	Weight
$N - rs$	$\frac{L.len}{E.ER + L.len}$	-	0.5
$rd$	$\frac{d_E(E.cen, L.m)}{L.len + E.ER}$	$s(rd)$	0.5
$rp_1$	$\frac{\min(OA, OB)}{\max(OA, OB)}$	$\frac{rp_1 + 1}{2}$	0.5
$rp_2$	$-\frac{\min( OA ,  OB )}{\max( OA ,  OB )}$	$\frac{rp_2 + 1}{2}$	0.5
$N - ro$	$\frac{ L.\theta - E.\theta }{180}$	-	0.5

(d) Line-to-Ellipse (L2E)

Symbol	Definition	Normalizn	Weight
$N - rs$	$\frac{C.r}{C.r + L.len}$	-	0.5774
$rd$	$\frac{d_E(C.cen, L)}{C.r}$	$s(rd)$	0.5774
$rp$	$\frac{\min(S1, S2)}{\max(S1, S2)}$	$\frac{rp + 1}{2}$	0.5774

(e) Circle-to-Line (C2L)

Table 2.4: Edge Attribute Definitions (Continued)

Attribute	Symbol	Definition	Normalization	Weight
Normalized relative size	$N - rs$	$\frac{C.r}{C.r+E.ER}$	-	0.5774
Relative distance	$rd$	$\frac{d_E(C.cen,E.cen)}{C.r+E.ER}$	$s(rd)$	0.5774
Relative position	$rp$	$\frac{\min(OA,OB)}{\max(OA,OB)}$	$\frac{rp+1}{2}$	0.5774

(a) Circle-to-Ellipse (C2E)

Attribute	Symbol	Definition	Normalizn	Weight
Eccentric. ratio	$er$	$\frac{E1.e}{E1.e+E2.e}$	-	0.2236
Eccentric. diff.	$ed$	$\frac{E1.e-E2.e+1}{2}$	-	0.2236
Rel. distance	$rd$	$\frac{d_E(E1.cen,E2.cen)}{E1.ER+E2.ER}$	$s(rd)$	0.4472
Norm. rel. size	$N - rs$	$\frac{E1.ER}{E1.ER+E2.ER}$	-	0.4472
Norm. rel. orient.	$N - ro$	$\frac{ E1.\theta-E2.\theta }{90}$	-	0.4472
Rel. position	$rp$	$rp(E1.maj - axis, E2.maj - axis)$	$\frac{rp+1}{2}$	0.4472

(b) Ellipse-to-Ellipse (E2E)

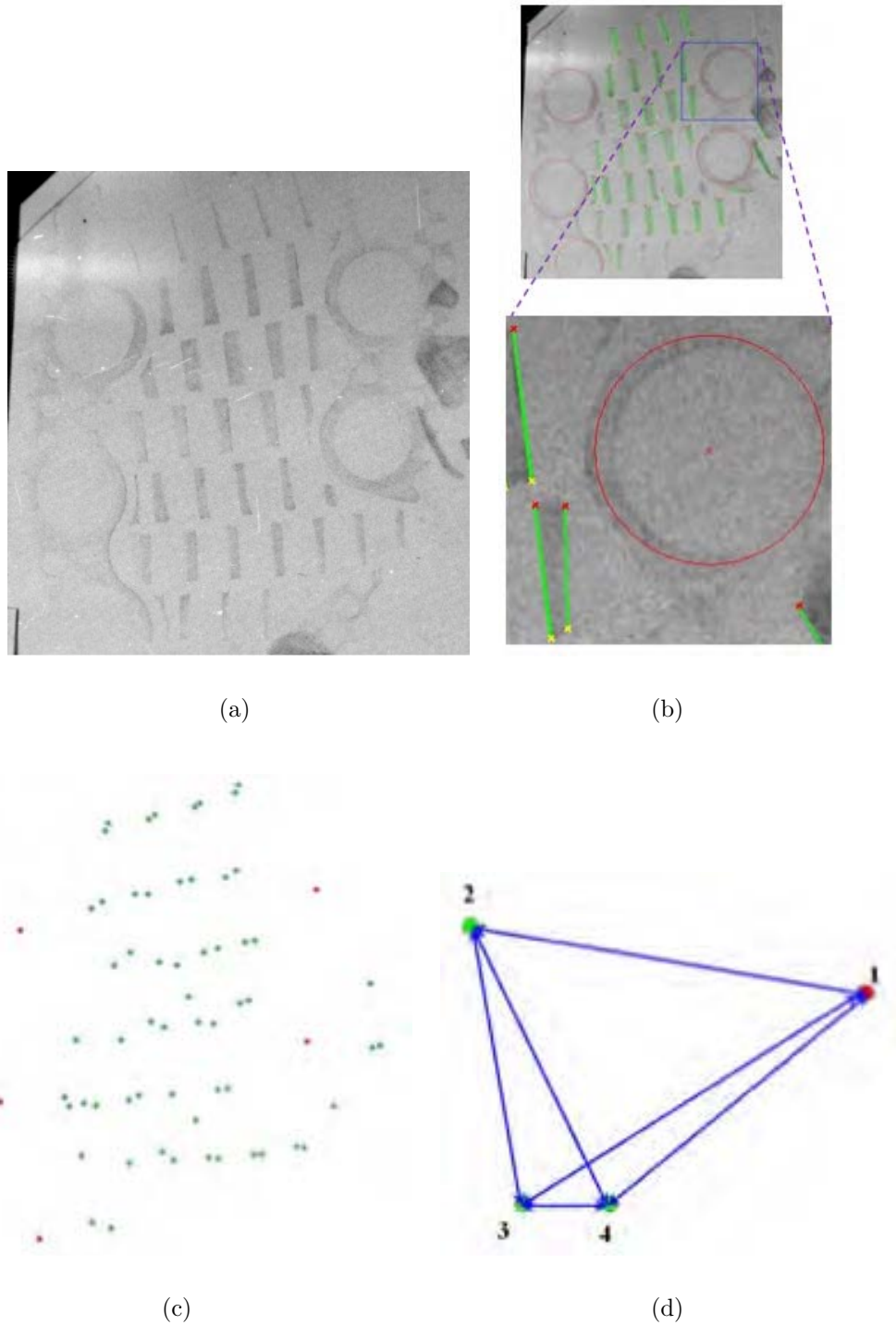


Figure 2.28: Attribute relational graph of a crime scene print: (a) print image, (b) detected circles and straight lines with magnified sub-image showing three straight lines to left of circle, (c) centers of 61 straight line segments (green points) and 5 circles (red), and (d) sub-graph for the three straight lines and circle in (b) whose attributes are given in Table 2.5.



Table 2.5: Node and Edge Attributes for subgraph in Figure 2.28(d).

Nodes and Edges	Attributes
Node 1	[0.0000, 0.7468, 0.5699]
Node 2	N/A
Node 3	N/A
Node 4	N/A
$E_{11}$	[0.5000, 0.0000, 0.0000]
$E_{12}$	[0.4285, 0.1976, 0.5989]
$E_{13}$	[0.4593, 0.1976, 0.3195]
$E_{14}$	[0.4809, 0.1387, 0.2316]
$E_{21}$	[0.5715, 0.1976, 0.5989]
$E_{22}$	[0.0000, 0.5000, 0.0000, 0.0000, 0.0200]
$E_{23}$	[0.0000, 0.5312, 0.0584, 0.0000, 0.0200]
$E_{24}$	[0.0323, 0.5527, 0.0609, 0.0146, 0.0626]
$E_{31}$	[0.5407, 0.1976, 0.3195]
$E_{32}$	[0.0000, 0.4688, 0.0584, 0.0000, 0.0200]
$E_{33}$	[0.0000, 0.5000, 0.0000, 0.0000, 0.0200]
$E_{34}$	[0.0324, 0.5217, 0.0091, 0.0090, 0.1018]
$E_{41}$	[0.5191, 0.1387, 0.2316]
$E_{42}$	[0.0323, 0.4473, 0.0609, 0.0085, 0.0901]
$E_{43}$	[0.0324, 0.4783, 0.0091, 0.0091, 0.0903]
$E_{44}$	[0.0000, 0.5000, 0.0000, 0.0000, 0.0200]

between two graphs. Thus in a directed graph with  $N$  nodes there will be  $N + 2(N(N - 1)/2) = N^2$  edges. The number of attributes at each edge depends on the types of nodes it connects.

The ARG for a crime scene print is shown in Fig. 2.28; the values of node and edge attributes for a portion of the subgraph with four nodes are given in Table 2.5.

### 2.2.7 Graph distance

Central to both retrieval and identification is a method for computing similarity between images. Equivalently, the inverse of similarity is a distance measure. The choice of similarity or distance measure is important since it influences the retrieval result, uncertainty of match, and the quality of clusters in partitioning the database for efficiency.

Image retrieval applications typically employ histogram (or probability density) distance measures. Bin-by-bin distance measures such as Euclidean distance (or its generalization known as the Minkowski distance), and Kulback-Leibler divergence are perceptually unsatisfactory. Cross-bin measures are quadratic similarity and Kolmogorov-Smirnov distance; the former is perceptually unsatisfactory, while Kolmogorov-Smirnov is only applicable to one-dimensional problems. A cross-bin distance metric, known as the Wasserstein metric, or the Earth Mover’s Distance (EMD) has become most popular in content-based image retrieval [53]. Advantages of EMD include: allowing for partial matches, ability to efficiently handle high-dimensional feature spaces and closeness to perceptual similarity when applied to image histograms. Since EMD computation forms an essential part of attribute relational graph matching, we describe it first.

#### Earth Mover’s Distance

The goal of EMD is to determine the least amount of work that is needed to transform one distribution into the other. It has to take into account the weights found in each of the two sets of bins and the ground distances between bins. From this it has to find the flow from each bin of the first histogram to the bins of the second histogram. Once the optimal flow is found the total amount transferred is the EMD.

Consider the evaluation of the distance between two signatures (also histograms or piles of dirt)  $P_1 = \{P_{1i} | i = 1, \dots, n_1\}$  and  $P_2 = \{P_{2j} | j = 1, \dots, n_2\}$ . The elements, or bins,  $P_{1i}$  have corresponding weights (or “supplies”)  $W_1 = [w_{1i}]$  and similarly  $P_{2j}$  has weights  $W_2 = [w_{2j}]$ .

The ground distance matrix  $C = [c_{ij}]$  specifies ground distance for all pairs of bins,  $c_{ij}$ . The flow matrix  $F = [f_{ij}]$ , where  $f_{ij}$  denotes the amount of “supplies” transferred from  $P_{1i}$  to  $P_{2j}$ , which *minimizes* the overall work is determined as follows

$$WORK(W_1, W_2, C, F) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} c_{ij} f_{ij} \quad (2.10)$$

which is subject to the following constraints:

$$f_{ij} \geq 0, \quad 1 \leq i \leq n_1, 1 \leq j \leq n_2, \quad (2.11)$$

$$\sum_{j=1}^{n_2} f_{ij} \leq w_{1i}, \quad 1 \leq i \leq n_1, \quad (2.12)$$

$$\sum_{i=1}^{n_1} f_{ij} \leq w_{2j}, \quad 1 \leq j \leq n_2, \quad (2.13)$$

and

$$\sum_{i=1}^{n_1} \sum_{j=1}^{n_2} f_{ij} = \min\left(\sum_{i=1}^{n_1} w_{1i}, \sum_{j=1}^{n_2} w_{2j}\right). \quad (2.14)$$

Constraint 2.11 allows moving “supplies” from  $P_1$  to  $P_2$  and not vice versa. Constraint 2.12 limits the amount of “supplies” that can be sent by the nodes in  $P_1$  to their weights. Constraint 2.13 limits the nodes in  $P_2$  to receive no more “supplies” than their weights. Constraint 2.14 forces to move the maximum amount of “supplies” possible. This amount is referred to as the total flow in the transportation problem.

This is a linear programming problem which is solved efficiently by the transportation simplex algorithm [54]. Once the flow matrix is found, the distance is defined as the overall matching cost normalized by the sum of all the weights transferred from  $P_1$  to  $P_2$ .

$$D(P_1, P_2) = \frac{\sum_{i=1}^{n_1} \sum_{j=1}^{n_2} c_{ij} f_{ij}}{\sum_{i=1}^{n_1} \sum_{j=1}^{n_2} f_{ij}} \quad (2.15)$$

### Attribute Relational Graph Distance

The task at hand is ARG matching rather than histogram matching. ARG matching requires an assignment algorithm which yields not only a correspondence between two sets of vertices but also the similarity between them. Some assignment algorithms are nearest neighbor search, Hausdorff distance and bipartite matching. However the EMD has shown itself to be versatile in solving the assignment problem [55]. The bins are replaced by vertices and relations between vertices. Both vertices (nodes) and relations (edges) have attributes associated with them. The vertices also have associated weights with them, which are useful in performing assignment.

A completely connected ARG is formally defined as  $P = (V, R, n)$  where  $V = \{V_i | 1 \leq i \leq n\}$  is the set of nodes and  $R = \{R_{ij} | 1 \leq i, j \leq n\}$  is the set of relations between nodes. Each node has a weight and an attribute vector,  $V_i = (w_i, \mathbf{v}_i)$  and each relation  $R_{ij}$  has an attribute vector  $\mathbf{r}_{ij}$ .

We wish to compute the distance between two ARGs,  $P_1 = (V_1, R_1, n_1)$  and  $P_2 = (V_2, R_2, n_2)$ . This involves finding an appropriate mapping  $M$  between the two sets of nodes according to the basic philosophy of a given assignment algorithm. The cost or ground distance matrix is  $C = [c_{ij}]$  where  $c_{ij} = c(V_{1i}, V_{2j} | V_{1i} \in V_1, V_{2j} \in V_2)$ . The unit cost between  $V_{1i}$  and  $V_{2j}$  is evaluated based on the similarity of the spatial configurations at the two nodes, which is explained later in this discussion.

By providing identical weights the nested structure of EMD can handle the case of sub-graph matching, i.e.,

$$w_{1i} = w_{2j} = \frac{1}{\max(n_1, n_2)}, 1 \leq i \leq n_1, 1 \leq j \leq n_2. \quad (2.16)$$

Unlike EMD, a node from  $P_1$  can only transfer its weight to a single node in  $P_2$ , which is known as the one-to-one constraint, i.e., enforce a one-to-one correspondence, each node  $i$  in the first ARG can only match one node  $j$  in the second ARG or be left unmatched.

Since the weight transferred from  $i$  to  $j$ ,  $f_{ij}$ , can only assume the value of either  $\frac{1}{\max(n_1, n_2)}$  or 0 we can rewrite Eq 2.15 as

$$D(P_1, P_2) = \frac{\frac{1}{\max(n_1, n_2)} \sum_{\{(i,j)|f_{ij}>0\}} c_{ij}}{\sum_{\{(i,j)|f_{ij}>0\}} f_{ij}} \quad (2.17)$$

The total number of correspondence pairs between the two ARGs is  $\min(n_1, n_2)$ . So the whole flow transferred from  $P_1$  to  $P_2$  is  $\frac{\min(n_1, n_2)}{\max(n_1, n_2)}$ . Substituting this term for the denominator,

$$D(P_1, P_2) = \frac{\sum_{\{(i,j)|f_{ij}>0\}} c_{ij}}{\min(n_1, n_2)} \quad (2.18)$$

Now for the determination of cost between two vertices. For a given pair of vertices in two graphs, how one vertex is different from the other depends not only on the vertex, but also on how they relate to their neighbors in terms of distance, orientation, position etc. This means that the similarity between two vertices should be evaluated based on the similarity between an attributed relational sub-graph rooted at the first vertex and another attributed relational sub-graph rooted at the second vertex. The attributed relational sub-graph is an *Attributed Tree*(AT) [56]. This leads to a nested structure of ARG matching or *EMD computation*. It consists of inner and outer steps.

For the outer step, the unit matching cost between nodes  $V_{1i}$  and  $V_{2j}$ , is defined as

$$c(V_{1i}, V_{2j}) = D(AT_{V_{1i}}, AT_{V_{2j}}), \quad (2.19)$$

where  $AT_{V_{1i}}$  and  $AT_{V_{2j}}$  are attributed trees rooted at  $V_{1i}$  and  $V_{2j}$  in the two ARGs. The attributed tree  $AT_{V_{1i}}$  consists of the vertex  $V_{1i}$ , all the vertices it is connected to, together with the relations in common. In the case of a completely connected graph there are  $n_1 - 1$  such relations.

To calculate the distance between the two trees  $AT_{V_{1i}}$  and  $AT_{V_{2j}}$ , we need to build the inner cost matrix  $C = [c_{i\hat{j}}]$  whose elements correspond to pairwise node-to-node ( $V_{1\hat{i}}$  to  $V_{2\hat{j}}$ ) distances in the two trees. The *inner cost* between  $V_{1\hat{i}}$  and  $V_{2\hat{j}}$  is calculated as follows. It takes into account not only the unary attributes of the vertices but also the binary relations of the related edges.

$$c(V_{1\hat{i}}, V_{2\hat{j}}) = \alpha d_E(\mathbf{v}_{1\hat{i}}, \mathbf{v}_{2\hat{j}}) + (1 - \alpha) d_E(\mathbf{r}_{1\hat{i}}, \mathbf{r}_{2\hat{j}}), \quad (2.20)$$

where  $\alpha$  is a weight coefficient in the interval  $[0, 1]$  reflecting the relative importance of the difference of node attributes and the difference of edge attributes in the evaluation of inner cost between two nodes,  $d_E$  stands for Euclidean distance and  $\mathbf{r}_{1\hat{i}}$  is the attribute vector of edge between  $V_{1i}$  and  $V_{1\hat{i}}$ .

The nodes  $V_{1i}$  and  $V_{2j}$  have one of three possible labels  $L$ ,  $C$  or  $E$  corresponding to line, circle, or ellipse. Thus there are 9 combinations of labels for  $(V_{1i}, V_{2j})$ . A line and a circle or an ellipse do not match regardless of their attributes and neighbors; while a circle and ellipse can match. Thus the unit matching cost for non-matching label pairs is  $c(L, C) = c(L, E) = 1$ . For the others the node-to-node inner costs are determined using

Eq. 2.20.

Algorithm PD, which summarizes the discussion above, computes the distance between two ARGs.

**Algorithm PD:** Distance between prints represented as ARGs

**Input:** Two ARGs,  $P_1 = (V_1, R_1, n_1)$ ,  $P_2 = (V_2, R_2, n_2)$

where  $V_1 = \{(V_{1i}, w_{1i}, \mathbf{v}_{1i}) | i = 1, ..n_1\}$ ,  $V_2 = \{(V_{2j}, w_{2j}, \mathbf{v}_{2j}) | j = 1, ..n_2\}$

and  $R_1 = \{(R_{1is}, \mathbf{r}_{1is}) | i, s = 1, ..n_1\}$ ,  $R_2 = \{(R_{2jt}, \mathbf{r}_{2jt}) | j, t = 1, ..n_2\}$

**Output:**  $D(P_1, P_2)$

1. Set each component  $w_{1i} = w_{2j} = \frac{1}{\max(n_1, n_2)}$ ,  $1 \leq i \leq n_1$ ,  $1 \leq j \leq n_2$
2. Compute outer cost matrix C
  - for**  $k_1 = 1$  to  $n_1$
  - for**  $k_2 = 1$  to  $n_2$
  - if** ( $isCompatible(V_{1k_1}, V_{2k_2}) == false$ )
  - $c(k_1, k_2) = 1$
  - else**
  - Build attributed trees rooted at  $V_{1k_1}$  and  $V_{2k_2}$ , *viz.*,  $AT_{V_{1k_1}}$  and  $AT_{V_{2k_2}}$
  - for**  $i = 1$  to  $n_1$
  - for**  $j = 1$  to  $n_2$
  - if** ( $isCompatible(V_{1i}, V_{2j}) == false$ )
  - $c'(i, j) = 1$
  - else**
  - $c'(i, j) = \alpha * d_E(\mathbf{v}_{1i}, \mathbf{v}_{2j}) + (1 - \alpha) * d_E(\mathbf{r}_{1k_1i}, \mathbf{r}_{2k_2j})$
  - end if**
  - end for**
  - end for**
  - $c(k_1, k_2) = D(AT_{V_{1k_1}}, AT_{V_{2k_2}})$
  - end if**
  - end for**
  - end for**
3. Compute flow matrix  $F$  using simplex method to minimize  $WORK(W_1, W_2, C, F)$
4. Return  $D(P_1, P_2) = \sum_{\{(i,j) | f_{ij} > 0\}} c_{ij} / \min(n_1, n_2)$

The function *isCompatible* checks to see if the node types are compatible, e.g., if one is a line and other is a circle then they are incompatible, as follows:

**Function** *isCompatible*(Node  $V_1$ , Node  $V_2$ )

**if** ( $V_1.label == 'L'$ ) & ( $V_2.label == 'C'$  ||  $V_2.label == 'E'$ )

**return** false;

```

else if( $V_2.label == 'L'$ ) & ( $V_1.label == 'C'$ ||  $V_1.label == 'E'$ )
    return false;
else return true;
end if

```

### Example of Distance Computation

As an illustrative example, consider two simple prints and their graphs shown in Fig. 2.29. Print  $P_1$  has five imperfect elements: three circles, an ellipse and a straight line, its ARG has five nodes  $\{V_{11}, ..V_{15}\}$ . Print  $P_2$  has six imperfect elements: two circles, one ellipse and three straight line segments, its ARG has six nodes  $\{V_{21}, ..V_{26}\}$ . Thus the number of edges in their ARGs are  $2 \times \binom{5}{2} = 20$  and  $2 \times \binom{6}{2} = 30$  respectively. Their corresponding node and edge attribute tables are shown in Tables 2.6 and 2.7 respectively.

To find the overall cost matrix between the ARGs of  $P_1$  and  $P_2$ , the unit matching cost between  $V_{1i}$  and  $V_{2j}$  is to be determined. For illustration, the method to compute the unit matching cost between  $V_{11}$  and  $V_{21}$  alone is described. The two attributed trees (shown in Fig.2.29) rooted at  $V_{11}$  and  $V_{21}$  are constructed so that all the neighbors of  $V_{11}$  and  $V_{21}$  are also considered during cost computation.

The cost matrices are shown in Table 2.8. The  $5 \times 6$  inner cost matrix is obtained for each node of  $P_1$  by comparing its tree of five descendants (including itself) with each of the six nodes of  $P_2$ .

The unit matching cost between  $V_{11}$  and  $V_{21}$  is another distance. Hence an inner cost matrix (shown in Table 2.8(a))  $c'_{ij}$  between the two trees is to be determined. If  $i^{th}$  node is a line and  $j^{th}$  node is a circle/ellipse then  $c'_{ij} = 1$  else it is computed using  $c'_{ij} = \alpha d_E(\mathbf{v}_{1i}, \mathbf{v}_{2j}) + (1 - \alpha)d_E(Q * \mathbf{r}_{11i}, Q * \mathbf{r}_{21j})$  where  $d_E$  stands for Euclidean distance,  $\alpha$  was set to 0.5 and  $Q$ , given in Table 2.3 and 2.4, was determined using sensitivity analysis, where \* denotes element-wise vector multiplication, which is discussed fully in (Section 2.2.8).

Once the inner cost matrix is determined, each node is assigned a weight of 1/6. Then the flow matrix  $[f_{ij}]$  is initialized to a feasible solution to determine the overall cost  $\sum_{i=1}^{n_1} \sum_{j=1}^{n_2} f_{ij}c_{ij}$ . This process is repeated iteratively to find the optimal flow matrix which minimizes the overall cost. Eventually, the cost between  $V_{11}$  and  $V_{21}$  is obtained using Eq 2.15.

Similarly the unit matching cost between the rest of  $V_{1i}$  and  $V_{2j}$  is calculated to get an outer cost matrix shown in Table 2.8(b). Once the outer cost matrix is determined, the flow matrix  $f_{ij}$  is computed as described earlier. Finally, the distance between  $P_1$  and  $P_2$  is obtained using Eq 2.15. Using the attribute tables shown this evaluates to 0.5674.

The process of similarity computation in a more realistic scenario involving actual footwear prints is shown in Figure 2.30. In this case the distance evaluates to a much smaller value of 0.0835 indicating a finer degree of match.

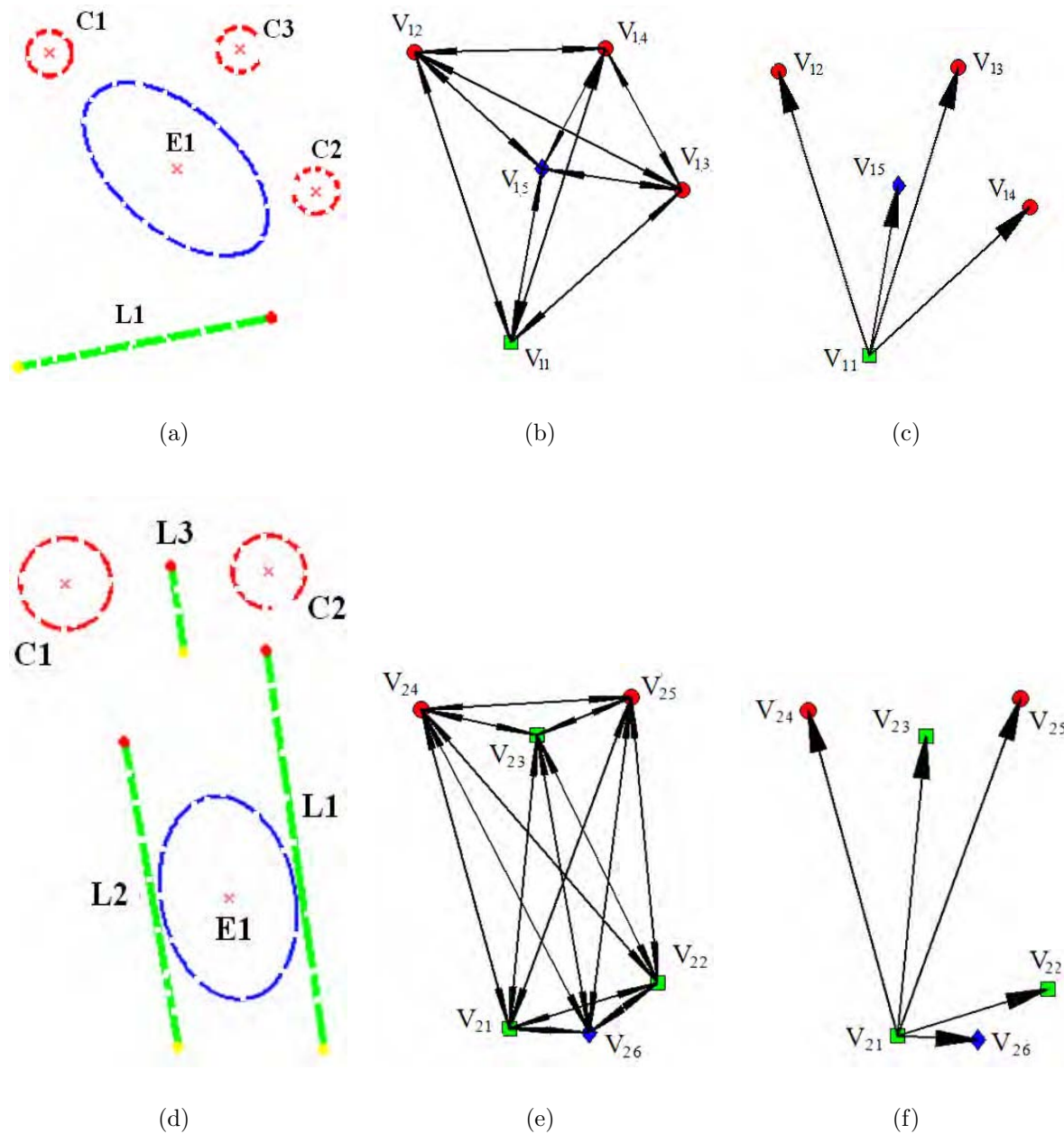


Figure 2.29: Illustration of distance computation between two simple prints: (a) print  $P_1$  with five primitive elements, (b) attributed relational graph of  $P_1$  with vertices  $V_{11}..V_{15}$ , (c) attributed tree rooted at  $V_{11}$ , (d) print  $P_2$  with six elements, (e) attributed relational graph of  $P_2$  with vertices  $V_{21}..V_{26}$  and (f) attributed tree rooted at  $V_{21}$ . Using the attribute values shown in Tables 2.6 and 2.7 the distance evaluates to 0.5674.

Table 2.6: Node and edge attribute values of print  $P_1$  shown in Fig. 2.29 (a)

Circle No.	Eccent.	Quality	Completeness
C1	0	0.8692	0.5755
C2	0	0.8325	0.5795
C3	0	0.7835	0.5661

(a) Node Attributes (Circles)

Ellipse	Eccent.	Quality	Completeness
E1	0.8365	0.8972	0.8214

(b) Node Attribute (Ellipse)

	$N - rs$	$N - rd_1$	$N - rd_2$
C1-C2	0.5	0.8245	1.0
C1-C3	0.5	0.6541	1.0
C2-C1	0.5	0.5748	1.0
C2-C3	0.5	0.8245	1.0
C3-C1	0.5	0.6541	1.0
C3-C2	0.5	0.5748	1.0

(c) Edge Attributes (Circle-Circle)

	$N - rs$	$N - rd$	$N - rp$
L1-C1	0.9219	0.9502	0.8829
L1-C2	0.9219	0.7378	0.3132
L1-C3	0.9219	0.8245	0.4639

(d) Edge Attributes (Line-Circle)

	$N - rs$	$N - rd$	$N - rp$
C1-L1	0.0781	0.9502	0.8829
C2-L1	0.0781	0.7378	0.3132
C3-L1	0.0781	0.9383	0.4639

(e) Edge Attributes (Circle-Line)

	$ed$	$N - rs$	$N - rd$	$N - rp$
C1-E1	0.8365	0.2111	0.1641	0.3009
C2-E1	0.8365	0.2111	0.1357	0.5086
C3-E1	0.8365	0.2111	0.1302	0.8866

(f) Edge Attributes (Circle-Ellipse)

	$ed$	$N - rs$	$N - rd$	$N - rp$
E1-C1	0.8365	0.7889	0.1641	0.3009
E1-C2	0.8365	0.7889	0.1357	0.5086
E1-C3	0.8365	0.7889	0.1302	0.8866

(g) Edge Attributes (Ellipse-Circle)

	$ed$	$N - rs$	$N - rd$	$N - rp$	$N - ro$
E1-L1	0.1635	0.2405	0.0517	0.2312	0.5845

(h) Edge Attributes (Ellipse-Line)

	$ed$	$N - rs$	$N - rd$	$N - rp$	$N - ro$
L1-E1	0.1635	0.7595	0.0517	0.3324	0.5845

(i) Edge Attributes (Line-Ellipse)



Table 2.7: Node and edge attributes of print  $P_2$  shown in Fig. 2.29 (d)

Circle No.	Eccentricity	Quality	Completeness
C1	0	0.8332	0.5441
C2	0	0.7596	0.5839

(a) Node Attributes (Circles)

Ellipse No.	Eccentricity	Quality	Completeness
E1	0.7564	0.8524	0.6913

(b) Node Attribute (Ellipse)

	$N - rs$	$N - rd$	$N - rp$
L1-C1	0.8996	0.6415	0.3286
L1-C2	0.9181	0.0123	0.3577
L2-C1	0.8733	0.0430	0.1984
L2-C2	0.8960	0.6959	0.2213
L3-C1	0.6606	0.3656	0.5794
L3-C2	0.7087	0.4106	0.7587

(c) Edge Attributes (L2C)

	$N - rs$	$N - rd$	$N - rp$
C1-L1	0.1004	0.6415	0.3286
C1-L2	0.1267	0.0430	0.1984
C1-L3	0.3394	0.3656	0.5794
C2-L1	0.0819	0.0123	0.3577
C2-L2	0.1040	0.6959	0.2213
C2-L3	0.2913	0.4106	0.7587

(d) Edge Attributes (C2L)

	$ed$	$N - rs$	$N - rd$	$N - rp$
C1-E1	0.7564	0.3515	0.2702	0.1050
C2-E1	0.7564	0.3025	0.2698	0.1162

(e) Edge Attributes (C2E)

	$ed$	$N - rs$	$N - rd$	$N - rp$
E1-C1	0.7564	0.6485	0.2702	0.1050
E1-C2	0.7564	0.6975	0.2698	0.1162

(f) Edge Attributes (E2C)

	$ed$	$N - rs$	$N - rd$	$N - rp$	$N - ro$
E1-L1	0.2436	0.1707	0.0169	0.0288	0.0285
E1-L2	0.2436	0.2112	0.0197	0.0202	0.0062
E1-L3	0.2436	0.4866	0.1708	0.1968	0.0285

(g) Edge Attributes (E2L)

	$ed$	$N - rs$	$N - rd$	$N - rp$	$N - ro$
L1-E1	0.2436	0.8293	0.0169	0.0419	0.0285
L2-E1	0.2436	0.7888	0.0197	0.0212	0.0062
L3-E1	0.2436	0.5134	0.1708	0.2341	0.0285

(h) Edge Attributes (L2E)

	$N - rs$	$N - rd_1$	$N - rd_2$
C1-C2	0.5556	0.3875	0.9809
C2-C1	0.4444	0.3875	0.9809

(i) Edge Attributes (C2C)

	$N - \alpha$	$N - rs$	$N - rp$	$N - rd$	$N - pd$
L1-L2	0.0111	0.5654	0.0258	0.0211	0.0210
L1-L3	0.0000	0.8216	0.0200	0.0547	0.0168
L2-L1	0.0111	0.4346	0.0239	0.0211	0.0209
L2-L3	0.0111	0.7797	0.0326	0.0722	0.0168
L3-L1	0.0000	0.1784	0.0200	0.0547	0.0168
L3-L2	0.0111	0.2203	0.0212	0.0722	0.0192

Table 2.8: Cost Matrices in comparing  $P_1$  and  $P_2$ .

0.1061	0.0000	0.3898	1.0000	1.0000	1.0000	0.5084	0.5133	0.5515	1.0000	1.0000	1.0000
1.0000	1.0000	1.0000	0.9701	0.7419	1.0000	1.0000	1.0000	1.0000	0.4999	0.5123	0.9990
1.0000	1.0000	1.0000	0.7918	0.1452	1.0000	1.0000	1.0000	1.0000	0.3873	0.3880	0.9990
1.0000	1.0000	1.0000	0.9545	0.3675	1.0000	1.0000	1.0000	1.0000	0.5167	0.5055	0.9990
1.0000	1.0000	1.0000	1.0000	1.0000	0.4214	1.0000	1.0000	1.0000	0.9687	0.9860	0.4406

(a) Inner Cost matrix for nodes  $V_{11}$  and  $V_{21}$

(b) Outer Cost Matrix

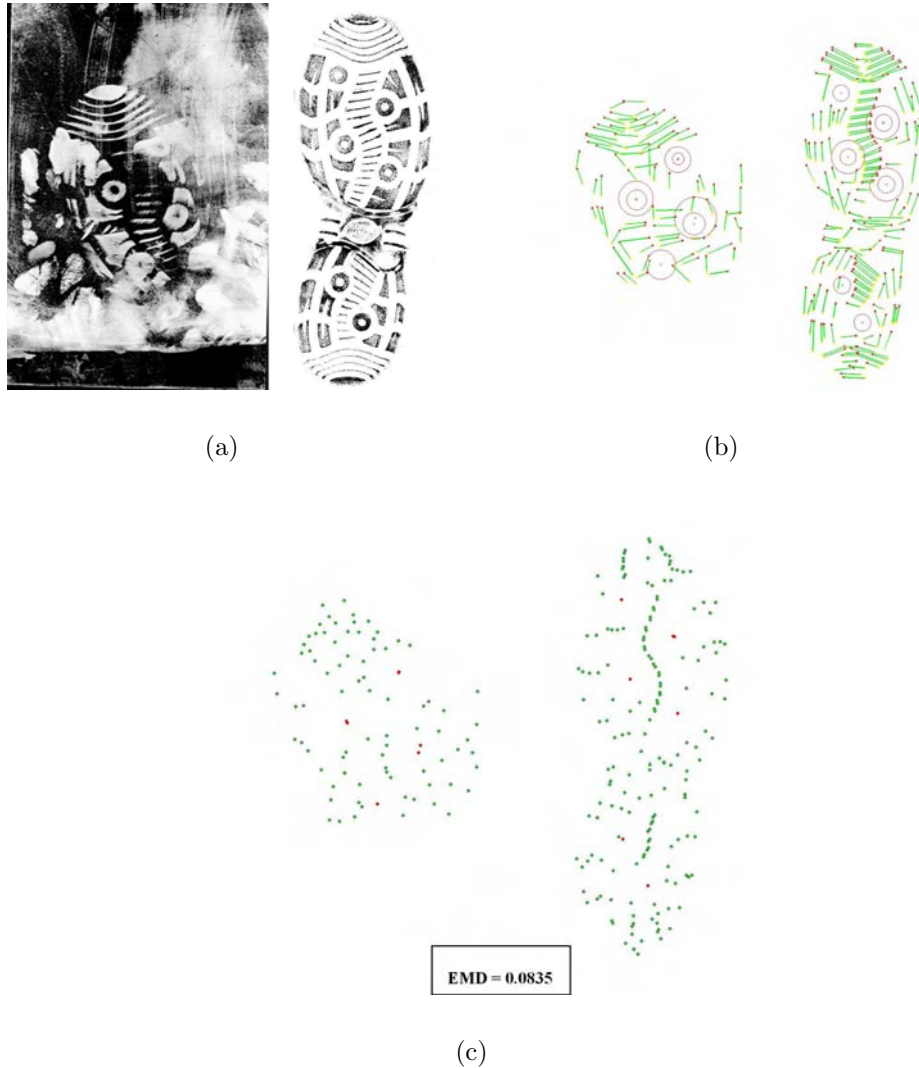


Figure 2.30: Similarity between a crime scene and a known print:(a) inputs, (b) detected lines and circles, and (c) graphs, where only nodes are shown for clarity, and similarity value.

## 2.2.8 Sensitivity analysis

Sensitivity analysis [57] is a system validation technique which can be used to determine robustness of the distance measure when the inputs are slightly disturbed. Its application here is to determine as to how sensitive the distance measure is to changes with respect to attributes (which were defined in Tables 2.3 and 2.4). Results of these experiments are shown as graphical plots in Fig.2.31, where 1,400 prints were used in the experiments.

There are fifteen graphs shown (labelled  $a - o$ ), each of which is a plot of distance with respect to one of the features. The first four correspond to how the distance between two lines (L2L) changes as each of five features are varied; they correspond to the features:  $N - \alpha$ ,  $N - rs$ ,  $N - rd + N - pd$ , and  $rp$ . They are followed by three for distance between two circles (C2C):  $N - rs$ ,  $N - rd_1$ ,  $N - rd_2$ , and one for circle completeness. There are six for distance between two ellipses (E2E):  $ed$ ,  $N - rs$ ,  $N - rd$ ,  $N - ro$ ,  $rp$  and  $rd$ .

A linear change is consistent with human perception whereas nonlinear behavior needs justification for its suitability. A linear correlation is seen between distance and most attributes indicating their acceptability. The exceptions are the four plots ( $b, j, n, o$ ) which are explained as follows:

( $b$ ) L2L  $N - rs$ : distance drops after reaching a peak, e.g., say two lines  $l_1$  and  $l_2$ , where  $l_1.len > l_2.len$ , as  $l_1$  becomes shorter  $l_1.len < l_2.len$  is reached and in that case the algorithm will switch their roles to minimize distance,

( $j$ ) C2C  $ed$ : similar reason as ( $b$ ),

( $n$ ) E2E  $rp$ : initially increases fast and then saturates because when the major axes of two ellipses is far, the rate of change in  $rp$  becomes increasingly small,

( $o$ ) C2C  $rd$ : when the radius  $r$  of one of the two circles vary randomly within 15%, the change of distance is always below 0.025.

Distance between ARGs has different sensitivities for different sets of attribute values. So we introduce a weight vector  $Q$  to take into account the difference in sensitivities. The calculation of inner cost is modified as follows

$$c(V_{1\hat{i}}, V_{2\hat{j}}) = \alpha d_E(\mathbf{v}_{1\hat{i}}, \mathbf{v}_{2\hat{j}}) + (1 - \alpha) d_E(Q * \mathbf{r}_{1\hat{i}}, Q * \mathbf{r}_{2\hat{j}}) \quad (2.21)$$

where  $*$  is the element-wise or Hadamard (Schur) product between two vectors.

Consider two synthetic prints shown in Figure 2.32, each of which contains two identical ellipses, i.e.,  $n_1 = n_2 = 2$ . We assume that  $\alpha$  equals 0.5, which means that node attributes and edge attributes are weighed equally. Using algorithm PD we arrive at the following distance between the two prints

$$D(P_1, P_2) = 0.25(d_E(\mathbf{v}_{11}, \mathbf{v}_{21}) + d_E(\mathbf{v}_{12}, \mathbf{v}_{22}) + d_E(Q * \mathbf{r}_{112}, Q * \mathbf{r}_{212})) \quad (2.22)$$

Following similar reasoning for two prints with arbitrary  $n_1$  and  $n_2$  nodes, with  $n_1 \leq n_2$ , we have

$$D(P_1, P_2) = \frac{\alpha \{ \sum_{i=1}^{n_1} n_1 d_E(\mathbf{v}_{1n_i}, \mathbf{v}_{2n_i}) + 2 \sum_{\{(i,j)|i < j\}} d_E(Q * \mathbf{r}_{1ij}, Q * \mathbf{r}_{2ij}) \}}{n_1^2} \quad (2.23)$$

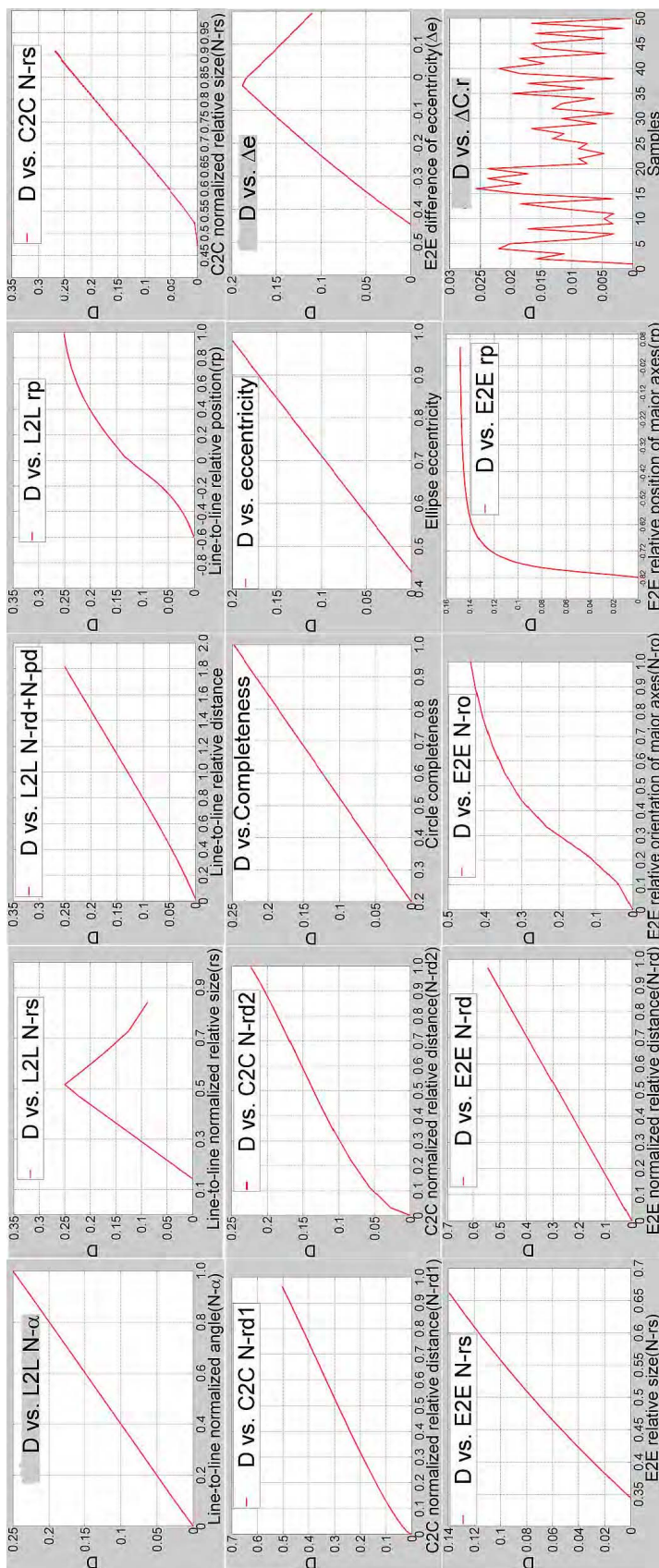


Figure 2.31: Sensitivity of distance measure to variations in each attribute, showing that  $D$  is *insensitive* to small errors in feature extraction. There are fifteen graphs— from top to bottom, left to right: labelled (a)-(e), (f)-(j), (k)-(o). A linear correlation is seen between  $D$  and most attributes indicating that  $D$  *consistently* measures human perceptual distance. The exceptions are (b, j, n, o) which are explained as follows: (b)  $D$  drops after reaching peak, e.g., say two lines  $l_1$  and  $l_2$ , where  $l_1.len > l_2.len$ , as  $l_1$  becomes shorter  $l_1.len < l_2.len$  is reached and in that case the algorithm will switch their roles to minimize  $D$ , (j) Similar reason as (b), (n)  $D$  initially increases fast and then saturates because when the major axes will switch their roles to minimize  $D$ , (o) Similar reason as (b), (n) increasingly small, (o) when the radius  $r$  of one of the two circles vary randomly within 15%, the change of  $D$  is always below 0.025.

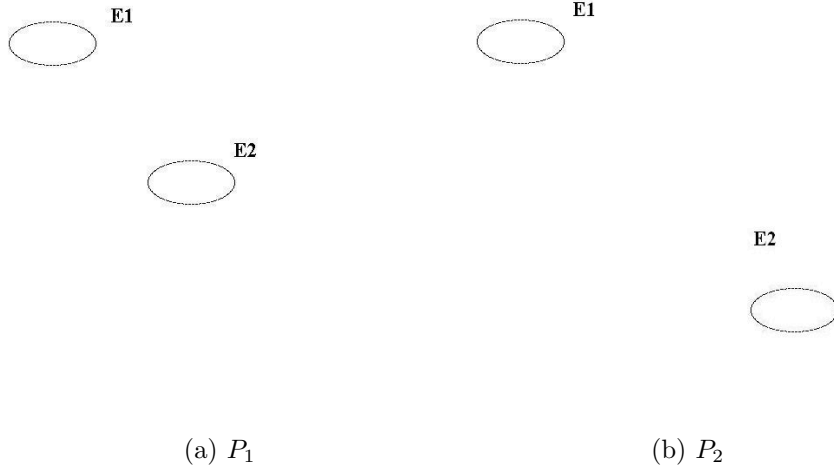


Figure 2.32: Example of two 2-node prints used in sensitivity analysis.

Here we assume that we can make a print become totally different from itself by changing each value of all the attributes (including node and edge attributes) from one extreme to the other, which means  $\Delta r = 1$ , where  $r$  is an edge attribute. Suppose that  $P_1$  and  $P_2$  are such two prints whose distance  $D(P_1, P_2)$  equals 1. Assuming that the difference of node attributes in each correspondence pair, as well as the difference of edge attributes in each correspondence pair, take *equal* responsibility for the distance  $D(P_1, P_2)$ . The number of shares of such responsibility is  $n_1^2 + n_1(n_1 - 1) = n_1(2n_1 - 1)$ .

$$d_E(Q * \mathbf{r}_{1is}, Q * \mathbf{r}_{2jt}) = \frac{n_1}{\alpha(2n_1 - 1)}, 1 \leq i, s \leq n_1, 1 \leq j, t \leq n_2. \quad (2.24)$$

We can rewrite the above equality as follows.

$$\sqrt{\sum_{k=1}^m Q_k (r_{1isk} - r_{2jtk})^2} = \frac{n_1}{\alpha(2n_1 - 1)}, \quad (2.25)$$

where  $m$  is the dimension of the edge attribute vector.

Finally we arrive at the values of weight vector as follows.

$$Q_k = \frac{\sqrt{\left[\frac{n_1^2}{\alpha n_1(2n_1-1)}\right]^2}}{1} = \frac{2n_1}{(2n_1 - 1)\sqrt{m}}, \forall k \in \{1, 2, \dots, m\} \quad (2.26)$$

For large  $n_1$ ,  $\frac{2n_1}{(2n_1-1)} \approx 1$ , thus we have  $Q_k \approx \frac{1}{\sqrt{m}}$ . When  $n_1 = 2$ ,  $Q_k = \frac{4}{3\sqrt{m}}$ . This indicates that we can determine the weights  $\{Q_k, k = \{1, \dots, m\}\}$  by first deriving the value of  $Q_k$  in the case of 2-nodes, then multiplying it by  $\frac{3}{4}$ . The contribution of *each* edge attribute for all pairs of nodes to distance can be calculated as  $\frac{\frac{2n_1}{(2n_1-1)*\sqrt{n}} * 1 * n_1(n_1-1) * \alpha}{n_1^2} = \frac{n_1-1}{(2n_1-1)\sqrt{n}}$ .

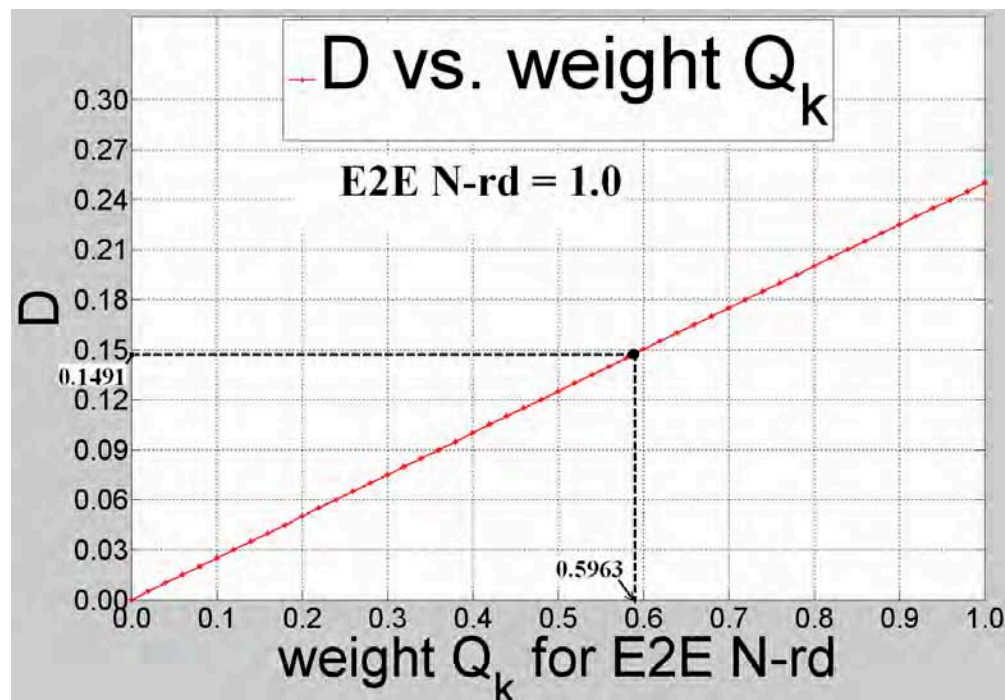


Figure 2.33: Plot of distance  $D$  against scale factor  $Q_k$  for attribute  $E2E N-rd$ . Prints with only two nodes were used in the experiments.  $N-rd(P_1) = 0$ ,  $N-rd(P_2) = 1$ .

From the previous definition of edge attributes as shown in Table 2.3 and 2.4,  $E2E$  (Ellipse-to-Ellipse) edge has 6 attributes. Since the attribute  $er$  and  $ed$  bear apparent dependency between each other, the number of independent attributes for  $E2E$  edge is 5, i.e.  $m = 5$ . We chose the attribute *Normalized relative distance*  $N-rd$  to conduct the experiments on prints with *only two* nodes to determine the relationship between the distance  $D$  and  $N-rd$  as well as the weight  $Q_k$ . This attribute is supposed to contribute to the final distance an amount of  $\frac{2-1}{(2*2-1)\sqrt{5}} = 0.1491$ .

In the experiments, the maximum change of  $N-rd$  is 1. Our goal is to find the value of  $Q_k$  s.t. the distance reaches 0.1491 when the change of  $N-rd$ ,  $\Delta N-rd$  is equal to 1 (We set the  $N-rd$  of  $P_1$  to be 0, so  $\Delta N-rd = N-rd(P_2)$ ). From Figure 2.33, we can get the value of  $Q_k$  as 0.5963, which is the same as what we would get if we substitute 5 for  $n$  in Eq 2.26. Finally, we obtain the weight  $Q_k$  with the attribute of  $E2E N-rd$  for prints with  $n_1$  nodes ( $n_1$  large) by multiplying 0.5963 by  $\frac{3}{4}$  to get 0.4472.

All the weights have been determined by means of both experiments and mathematical derivations and summarized in Tables 2.3 and 2.4.

The resulting weights are shown in the last columns in the definitions of edge attributes in Tables 2.3 and 2.4. They are used as the weight vector  $Q$  in distance computation as shown in Eq. 2.21.



### 2.2.9 Clustering

The process of comparing ARGs is extremely slow. The computational complexity of distance computation is  $O(n_1 n_2 \max(n_1, n_2))$ . This is essentially cubic in the number of nodes  $n$  [55]. Thus it is necessary to use approximate methods to speed-up the computation. One approach is to eliminate several edge evaluations. Another is to cluster the known data set such that not all comparisons need to be made against the database. The clustering approach was developed as described here.

Clustering algorithms can be generally divided into partition-based, density-based and hierarchical based methods [58]. Algorithms like  $k$ -means, hierarchical clustering, and expectation maximization requires similarity matrix consisting of pair-wise distance between every footwear prints in dataset. Building similarity matrix is computationally expensive for a large dataset. Further, the ARG representing a footwear print has 200-300 nodes on average and nodes can vary considerably in terms of relative size, position etc. This makes the feature space very *sparse* and therefore similar footwear prints tend to stay close to each other and dissimilar ones stay apart. Hence, to cluster the entire dataset we use *recurring patterns* as *fixed* cluster centers [59].

Recurring patterns [3, 6] such as waves and concentric circles are common in footwear outsoles and each of them can be used to represent a group of similar prints. These patterns are simple in structure and the graphs constructed, from these patterns, have many fewer nodes. Hence, recurring patterns can be used as query to fetch similar database prints. This drastically reduces the computation required and does not require a similarity matrix. Although this clustering requires domain knowledge to determine recurring patterns, it avoids the problems of deciding the number of clusters beforehand (unlike  $k$ -means).

From visual inspection of 1000 prints, 20 recurring patterns (shown in Fig. 2.35) were determined and used as cluster representatives. For each database footwear print, we computed its FPD to each pattern and then assigned it to the nearest cluster representative. These cluster representatives are similar to cluster means in  $k$ -means algorithm but these "means" are fixed. This efficiency is achieved by exploiting sparseness of the feature space. Most prints have a resolution of either 150 or 72 DPI.

*Step 1 (Morphology)*: The first step in the feature extraction is to perform morphological operations such as dilation and erosion. This makes the interior region of the boundary *uniform* and hence the Canny edge detector [60] does not detect any edges inside the boundary. This helps to enhance the quality of the edge image. Result of step 1 for a sample footwear print is shown in Figure 2.36.

*Step 2 (Hough Transform)*: SHT is used to detect circles in footwear prints. Pixels of detected circles shown in Fig. 2.37(a) are removed from the edge image and fed as input for ellipse detection using RHT. Pixels of detected ellipses shown in Fig. 2.37(b) are removed from the edge image and the output is fed as input for line detection in Fig. 2.37(c). Features are extracted in the order: circle, ellipse and line. This is because circles are degenerated ellipses and arbitrary shapes in footwear print are approximated by piecewise lines. Fig. 2.37(d) sums up all the features.

*Step 3 (ARG)*: For each detected feature, node attributes like completeness & quality of circle, eccentricity of the ellipse etc. are computed. Further, edge attributes like relative

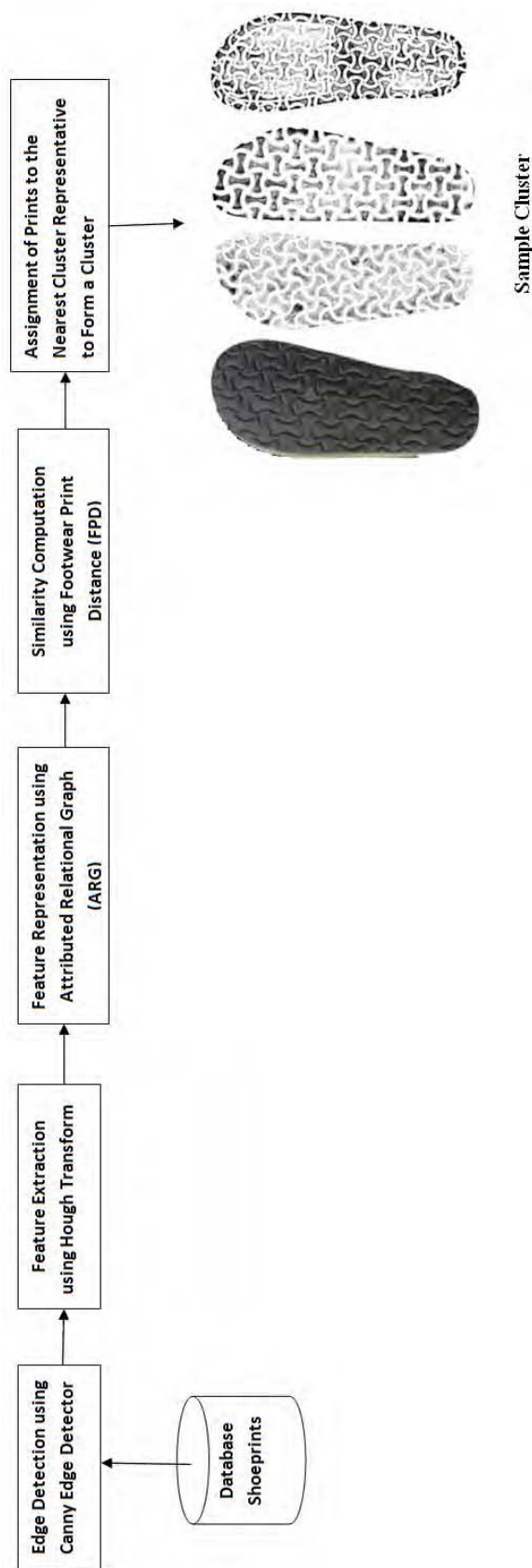


Figure 2.34: Process flow for clustering footwear outsole prints in database.



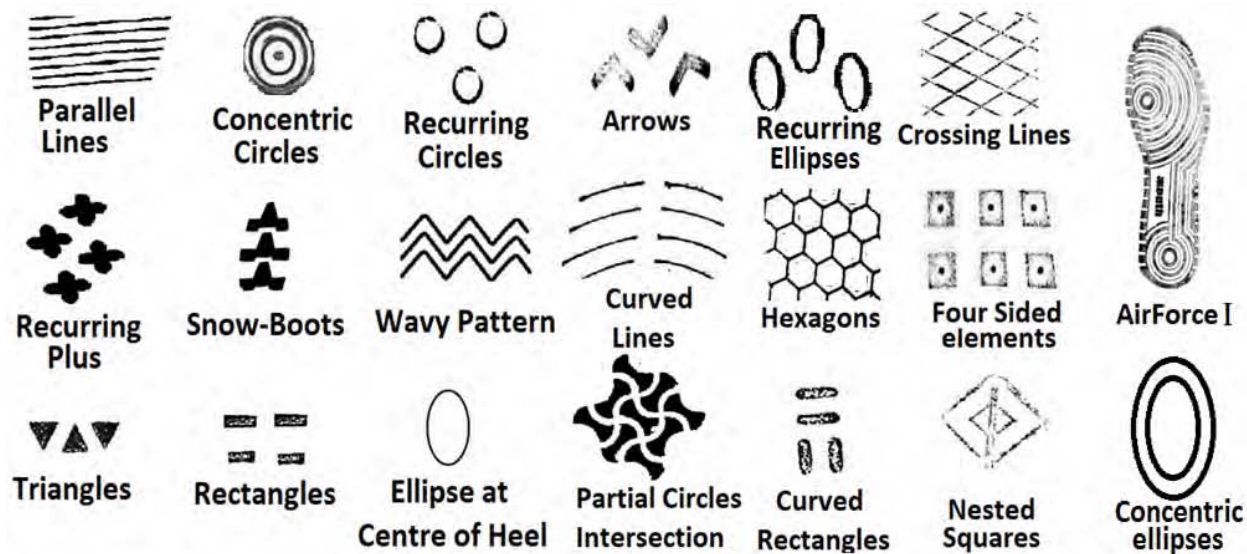


Figure 2.35: Canonical Patterns for Clustering.

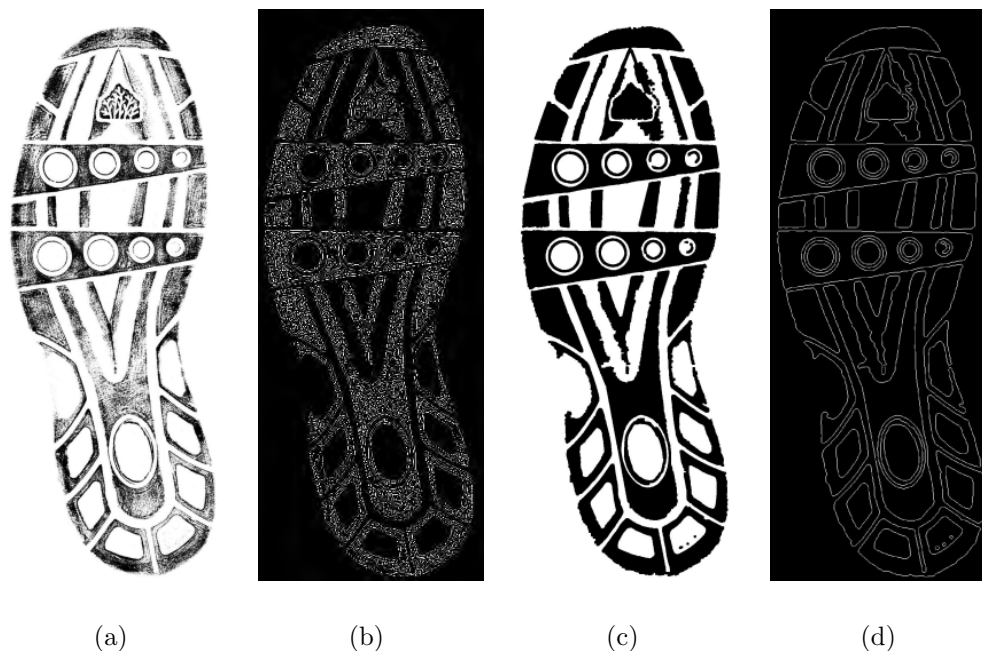


Figure 2.36: Clustering Step 1(Morphology): (a) Original Gray-scale Image, (b) Edge Image of (a), (c) Result of Morphological Operation on (a), (d)Edge Image of (c).

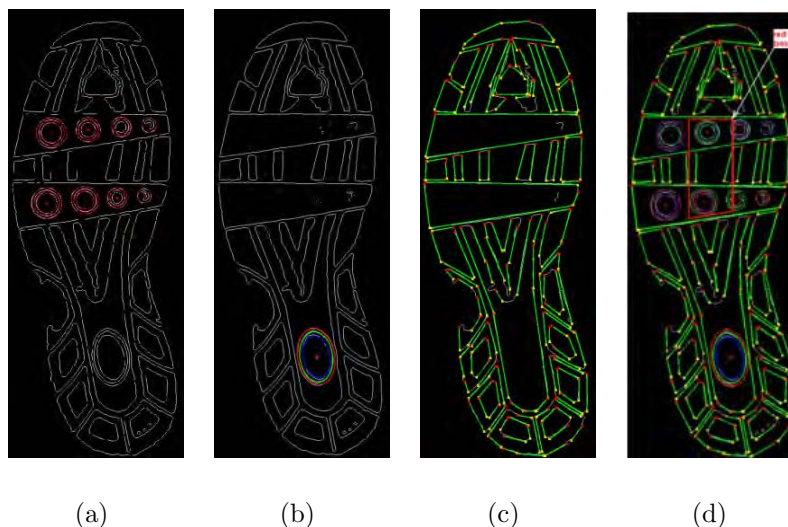


Figure 2.37: Clustering Step 2 (Hough Transform). Extracting features in the sequence circle→ellipse→line: (a) Circles. (b) Ellipses. (c) Line Segments. (d) All features. Red box indicates a small region in the footwear print.

distance & position between nodes are calculated and finally an ARG is constructed. One such ARG is shown in Fig. 2.38.

*Step 4(Distance):* FPD between each database print and every cluster representative is calculated. Then each print is assigned to the nearest representative, for which the FPD is below threshold  $T$ . If FPD between a print and cluster representatives are greater than  $T$ , then the print remains as a single cluster. In our experiments,  $T$  was set to 0.15.

The method was evaluated on 1000 footwear prints. The system assigned 550 footwear prints to one of 20 clusters whereas the remaining 450 prints were so unique that each of them was a cluster by itself. Two sample clusters based on the canonical patterns of Fig. 2.35 are shown in Fig. 2.39.

Retrieval performance was measured by the  $F$ -measure, the weighted harmonic mean of precision and recall. Its value is also a measure of clustering's accuracy.

$$F = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (2.27)$$

The Precision vs. Recall curve and the maximum  $F$ -measure are shown in Fig. 2.40.

One advantage of this clustering method is huge reduction in computation. For a database of 1000 prints, existing clustering algorithms would require  ${}_{1000}C_2 = 499,500$  FPD computations to build the similarity matrix. However, our clustering method would take  $1000 \times k$  FPD computations, where  $k$  is the number of recurring patterns. In our case,  $k = 20$ , so the computation is reduced by 96%. This efficiency is achieved without compromising the accuracy or recall rate (shown in Fig. 2.39).

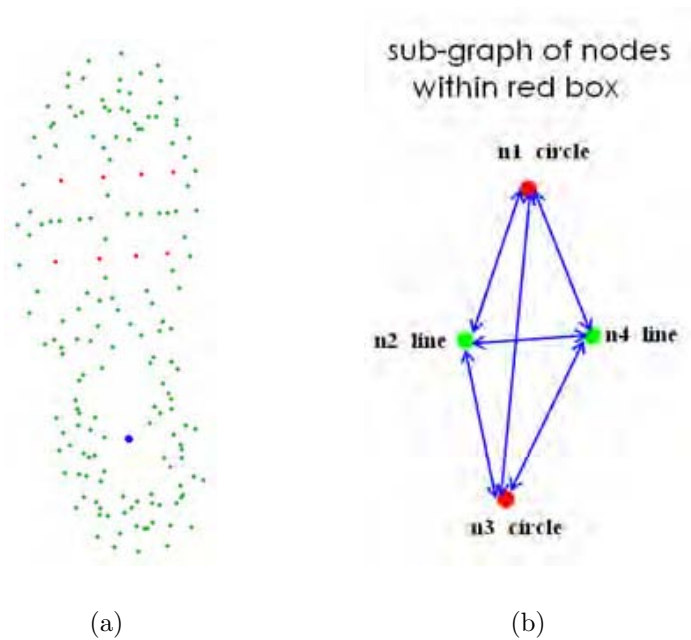


Figure 2.38: Clustering Step 3 (ARG): (a) nodes in graph corresponding to image in Figure 2.37 with edges omitted due to complete connectivity, (b) subgraph for region enclosed in the red box of Figure 2.37 (d). Red and green dots represent circles and lines respectively.

2.2. METHODS

CHAPTER 2. RESEARCH NARRATIVE










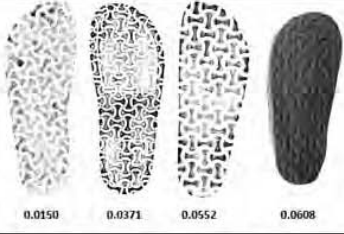
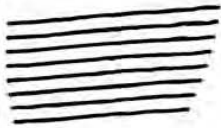

Cluster Name	Cluster Representative	Total number of prints in the cluster	Sample prints
Wavy Pattern		145	 0.0040    0.0472    0.0473    0.0498    0.0617
Snow Boots Pattern		5	 0.0635    0.0822    0.0947    0.0957    0.0998
Concentric Circles		20	 0.0000    0.0319    0.0693    0.0957    0.0969
Recurring Plus		6	 0.0881    0.0932    0.0953    0.0982    0.0995
Partial Circles Interleaves		4	 0.0150    0.0371    0.0552    0.0608
Parallel Lines		25	 0.0050    0.01214    0.0124    0.0594    0.0634

Figure 2.39: Sample clusters based on using the canonical patterns in Fig. 2.35.

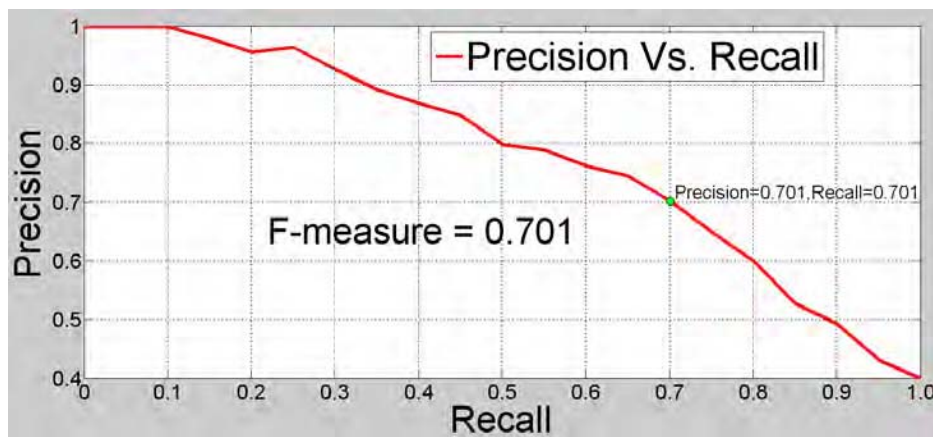


Figure 2.40: Retrieval Performance: F-measure corresponds to its maximum value.



### 2.2.10 Retrieval performance evaluation

An evaluation of the end-to-end performance of soleprint image retrieval was performed in terms of both retrieval performance and speed. A preliminary set of experiments were performed using a dataset of 50 crime scene shoeprints as query and a database of 1066 known shoeprints. Each known shoeprint has meta data information such as brand and model of the shoe. In the clustered database, each crime scene mark was used as a query and the closest prints were retrieved. The crime scene mark was matched against every cluster representative to find the closest cluster. In the closest cluster, crime scene mark was matched against each of the prints and the top  $n$  matches from the cluster were retrieved. Sample retrieval results for two crime scene marks are shown in Fig.2.41.

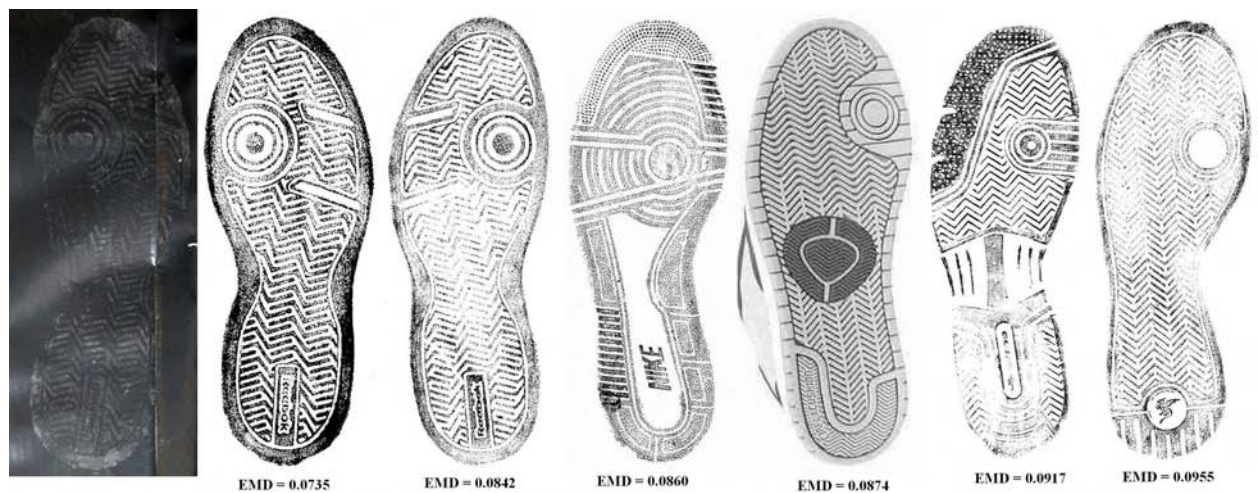
In terms of overall performance, the performance metric known as the *cumulative match characteristic (CMC)* was chosen as it can answer the question [14] “what is the probability of finding a match in the first  $n$  percent of database images?”. The probability of a match (*cumulative match score*) was estimated by the proportion of times during the tests when a match shoeprint appears in the first  $n$  percent of sorted database. Also, this metric is suitable when there is only one matching shoeprint in the database. The CMC curve of the system *before* clustering and of SIFT are shown in Fig.2.42.

For a baseline comparison, experiments were also performed using the SIFT feature descriptor. SIFT [43] being a robust image retrieval algorithm. SIFT is the apparent method used in Google’s Similar Image search. The resulting CMS is much better than the CMS of SIFT. Dardi et al. [21], the only ones to report accuracy with crime scene marks, state 72% accuracy. The algorithm proposed here achieves an accuracy of 92% for crime scene marks & an accuracy of 100% for degraded prints. Tests with crime scene marks have an error of 0.08% but from the confidence interval of sample size we found that true error ranges between 0.03% and 0.18% for the used sample size. A comparison with the state-of-the art techniques as described in the literature reveals that the proposed system outperforms each of them (see Table 2.9).

The ratios of finding a correct match within the first 1, 2, 5, 10, 20, 50 images in the sorted database are listed in Figure 2.42(b). SIFT performs only slightly better than randomly selecting an image. This is possibly because SIFT features are not preserved among different shoes of the same class and throughout the lifetime of a single shoe. The proposed ARG-EMD method greatly outperforms SIFT. From the curve we can see that, if looking at the very first 0.1% database images, we still have the probability of 0.426 to find a correct match. This means only reviewing the first 10 images for a database of 10,000 images. This is because our method extracts geometric shapes such as circles, lines, which are the most durable and preservable features, and the attributed relational structure constructed based on primitive features is able to capture the distinctiveness of each shoeprint pattern, and have the desirable scale, rotation invariances. Finally ARG-EMD matching allows partial matching in a natural way and is robust to the change of the relational structure. In addition to the quantitative analysis, we present some retrieval results in Figures 2.41 to show how these results are close to perceptual similarity.



(a)



(b)

Figure 2.41: Results of automatic retrieval with two queries shown as the left-most images followed on the right by the top database entries retrieved. It can be seen that the top choices are similar to human perception.

Table 2.9: Comparison of ARG-FPD approach with the state-of-the-art

State-of-the-art	Full print			Partial print			Experiments with Scene of Crime (SoC)	Shortcoming	Dataset
	CMS @ 1%	CMS @ 5%	CMS @ 10%	CMS @ 1%	CMS @ 5%	CMS @ 10%			
de Chazal et al. (2005)	64	87	90	50	70	77	-	Lacks Scaling invariance	475 prints from For. Sci. Lab., Ireland
Zhang et al. (2005)	85.4	95	97.44	-	-	-	-	Not tested with partials	512 prints from F & F
Pavlou et al. (2006)	86	90	93	85	90	92	-	Not tested with real SoC marks	368 prints of For. Sci. Serv, UK
Crookes et al. (2007)	100	100	100	100	100	100	-	Tested with synthesized SoCs	500 clean prints, 50 degraded
Crookes et al. (2007)	100	100	100	100	100	100	-	Lacks rotational invariance	100 clean prints, 64 synthetic
Gueham et al. (2008)	-	-	-	-	-	95.68	-	Tested with 100 prints	100 F & F prints
Dardi et al. (2009)	-	-	-	-	-	-	CMS @ 10% is 73% CMS @ 5% is 40% CMS @ 1% is 10%	Tested with 87 known prints and 30 SoCs	87 known and 30 real SoC ENSFI
Srihari et al. ARG-FPD (2010)	100	100	100	100	100	100	CMS @ 10% is 92% CMS @ 5% is 90% CMS @ 1% is 70%	-	1400 degraded, 1000 known & 50 real SoC



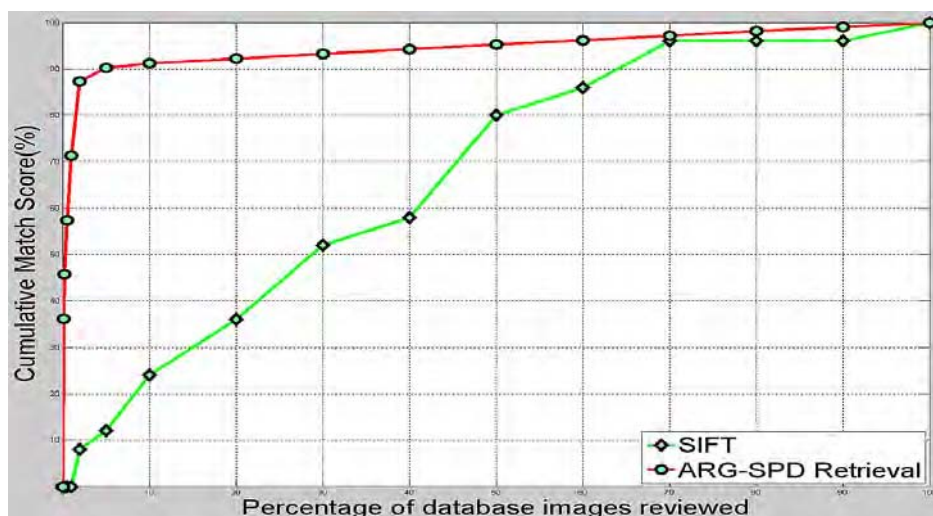


Figure 2.42: Cumulative Match Characteristic of ARG-EMD and SIFT.

Table 2.10: Retrieval speed before and after clustering.

No of crime scene marks	Ave time before	Ave time after
50	120 mins	8 mins

## Speed

The speed of comparison before and after clustering is given in Table 2.10. CMC curve remains the same before and after clustering but clustering makes significant improvement in retrieval speed.

Speed of clustering was improved by adding a pre-filtering procedure that computes the Euclidean distance between global feature vectors of the query shoeprint and each database shoeprint, and ignores those database images which are too far from the query to be a potential match

The time for processing a crime scene image and a database image depends on the number of nodes in each graph constructed on each image. On average, it takes 30 seconds to compute one FPD. Thus for a single query and 1,000 database entries, it takes 20-30 minutes. In a large shoe-print database, the efficiency(speed) of retrieving a query print becomes important. Effective indexing techniques should be designed to enter standard shoeprint prototypes.

The focus for speed-up can be determined from the following break-down: (i) Hough transform: 15s (detect lines) + 60s (detect circles) = 75s, (ii) graph construction: 0.1s, and (iii) FPD distance computation for each pair: 30s. Distance computation is slow since it involves a nested structure. The number of FPDs that need to be computed is the product of the number of nodes in the query and the database entry.

Speed can be improved by: (i) reducing the number of nodes by merging two detected lines which are associated with a single straight boundary (to be done), (ii) using pre-filtering to enhance the speed performance e.g., computing the Euclidean distance between global

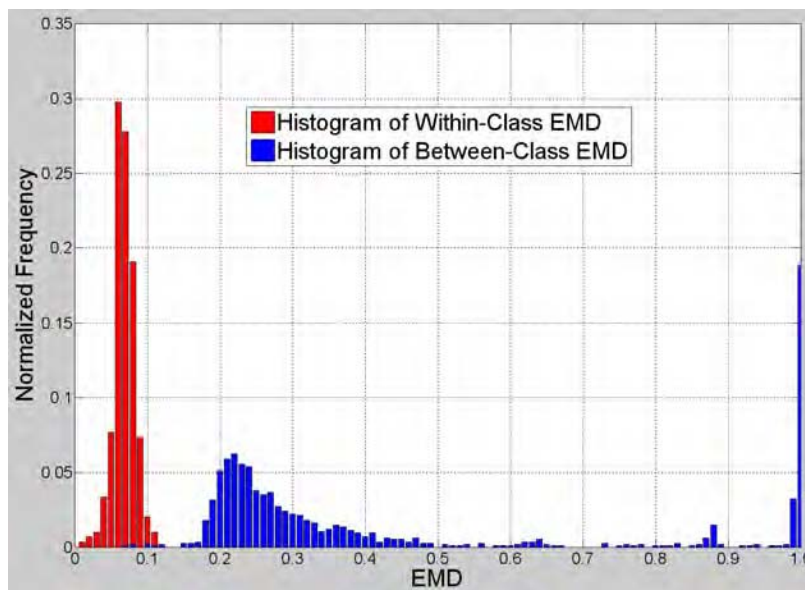


Figure 2.43: Probabilistic Model: intra- and inter-class distributions of distance allow computing likelihood ratios.

feature vectors of the query print and each database print, and ignoring those database prints too far from the query to be a potential match, (iii) relaxing full connectivity in graph by triangulation, and (iv) other improvements.

### 2.2.11 Uncertainty computation

In the previous sections an approach has been proposed to compute the similarity between a query print and a database print. The similarity measure is not only useful for database retrieval but can also form the important first step for measuring the degree of uncertainty. To evaluate the strength of the evidence, it is necessary to either transfer the distance/similarity measure or the number of matching features into a probability measure.

The likelihood ratio is the ratio of the two probabilities of the evidence given two competing hypotheses:  $H$  – the crime scene print is created by the same footwear as the known print and  $\bar{H}$  – the crime scene print is not from the known. This ratio can be expressed as:  $LR = \frac{P_r(E|H,I)}{P_r(E|\bar{H},I)}$ . where  $E$  is the evidence given by the crime scene mark, and  $I$  is all the background information relevant to this case. This approach can be decomposed into the following three steps: (i) estimate the within-class and between-class shoeprint variability as measured by graph distance (see Figure 2.43), (ii) compute the LR for the evidence, and (iii) convert the LR into a verbal scale which can be conveyed to different shoeprint examiners.

*Degradation Model:* Features present in a footwear print can be classified into two categories: manufacturing and acquired. Manufacturing features are those that comes from the manufacturing process. Acquired features are features that have been acquired during the lifetime of the shoe, such as wear pattern and damaging features.

*Within-class variability* measures the variance of features of multiple shoeprints from the

same shoe. To be able to simulate different variations caused by wears, incompleteness and the change of medium and illumination, we apply image degradation models multiple times on each database image to produce a set of degraded shoeprints. Our matching algorithm will be applied to calculate the distance between each pair of within-class shoeprints. Then we will be able to build a probability distribution of within-class distance.

*Between-class variability* measures the variance of features of multiple shoeprints that are from different classes. This distribution can be built following a similar way as for within-class variability.

Given a distance between the crime scene mark and a test mark made by the suspect's shoe, we can compute how likely is the distance given the hypothesis that the two marks are from the same source, as well as how likely is the distance given the hypothesis that the two marks are from different sources. The ratio of these two likelihood is then calculated to get the LR. The distribution of LRs, determined from a learning set, can be used to convert the LR value into an opinion scale.

### 2.2.12 User interface

A preliminary system to retrieve known images from given a crime scene image has been put together for demonstration purposes. It runs under the Windows operating system. The opening splash screen is shown in Figure 2.44(a). There are user options for opening image files and for performing the search. An example of the results of matching a query crime scene print against a set of knowns is shown in Figure 2.44(b). Note that the query image has present in it several cracks in the sidewalk on which the impression was created, which does not affect the results of matching. In this demonstration the features of the knowns and the query were pre-computed. Computing the features takes about fifteen minutes per image on a Windows PC using the current implementation in MATLAB.

## 2.3 Conclusions

The research has articulated tasks in footwear outsole impression evidence matching that are suitable for automation. Computational methods were defined and studied for several sub-problems. They include methods for enhancing the quality of images, extracting features useful for comparison, determining the degree of similarity between evidence and known, efficient implementation of algorithms to retrieve closest matches in a database, and computation of match uncertainty. Major findings in each of these areas are summarized below.

### 2.3.1 Discussion of findings

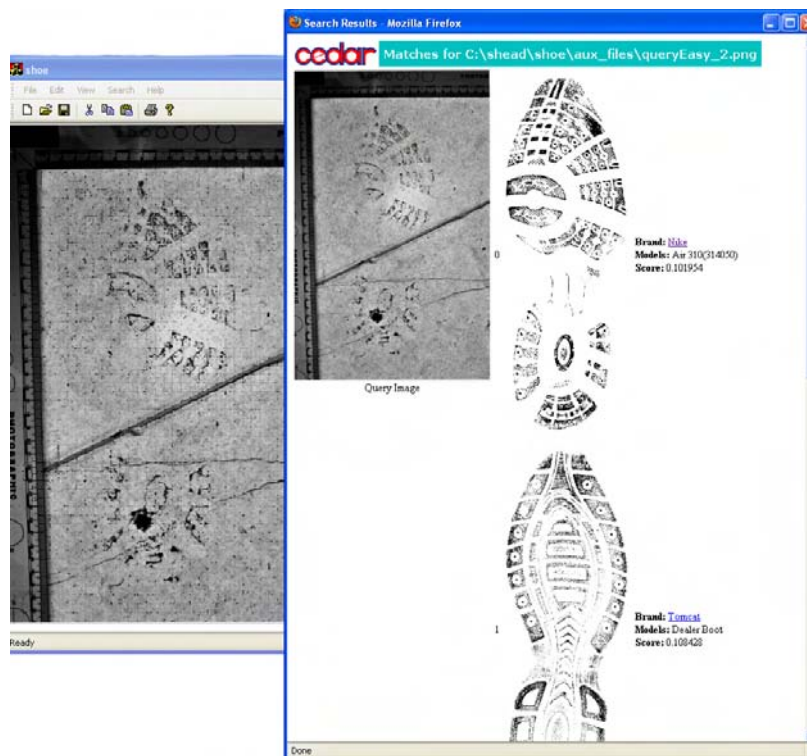
1. *Image enhancement methods.* Several algorithms for extracting foreground pixels from the crime scene image were studied. Among these a method based on utilizing statistical dependencies between nearby pixels worked better than simpler thresholding

## 2.3. CONCLUSIONS

## CHAPTER 2. RESEARCH NARRATIVE



(a)



(b)

Figure 2.44: User Interface: (a) opening screen with tool bar, and (b) a query and results.

## 2.3. CONCLUSIONS

## CHAPTER 2. RESEARCH NARRATIVE

algorithms used in applications such as document processing. Morphology based algorithms also performed well.

2. *Methods of representing footwear outsole patterns.* Three different methods of extracting features were investigated and compared. Features based on detecting component geometric shapes, principally as ellipses of different eccentricities, is more reliable than either global image features used in questioned document examination (such as GSC) and in three-dimensional scenes (such as SIFT). Experimental evaluation with crime scene images suggest that geometric shapes are robust to various degradations and that an attributed relational structure is a robust descriptor.
3. *Computing similarity between patterns.* A similarity measure based on comparing node and edge attributes in ARGs is robust with respect to small variations in the input features.
4. *Clustering of patterns.* By clustering known images into cognitively similar patterns, higher efficiency is achieved in retrieval.
5. *Performance evaluation.* a graph matching algorithm based on a nested distance computation structure can achieve stable and prominent performance even under moderate change of the topology of the graph.

### 2.3.2 Implications for policy and practice

1. Automated tools, such as those developed here, have the potential to assist footwear impression evidence examiners in searching through large data bases to find footwear impression type. This should enable more use of footwear impression evidence in crime scene investigation.
2. Measures of similarity developed will not only allow retrieval of known prints in the *investigative phase* but will also be useful in the *prosecutorial phase*. A quantitative similarity measure will allow quantification of uncertainty in presenting evidence in the courtroom. This will allow moving away from individualization testimony involving absolutes, which is a point underscored in the NAS report [61].
3. Project has synergy with other areas of impression evidence, specifically questioned document examination and latent print analysis. The graph structure method described will be particularly useful when the impression evidence involves man-made geometrical patterns.

### 2.3.3 Implications for further research

The following are areas that needs to further studied:

1. Statistical machine learning approaches can be used effectively in several phases of the task. This research has developed algorithms for image processing, feature extraction, similarity computation, and clustering. While CRFs were shown to be useful

for extracting foreground image pixels, the model needs to be further studied for its practicality. More sophisticated clustering algorithms such as *EM* could be tried.

2. A standardized database of crime scene marks will allow researchers to measure the performance of their systems.
3. The variability of the similarity measure needs to be further studied, e.g., different sizes of the query image, increased number of footwear models, etc. The distribution of the similarity measure is also central to the quantification of uncertainty. The similarity measure is analogous to other such measures in impression evidence. For instance the similarity between two latent prints is measured using the Bozorth matcher, which is also based on a graph representation of minutiae.
4. Similarity computation is very slow due to  $O(N^3)$  computation in the number primitive shapes. Thus more efficient implementations that perform approximate matching need to be designed.
5. The use of the similarity metrics in the computation of likelihoods for the prosecution and defense hypotheses need to be studied so as to provide uncertainty measures in comparison.
6. Most of the code in this research was implemented in MATLAB, a programming environment suitable for exploratory research. These need to be further refined and converted to C++ for speed and efficiency.
7. A preliminary user interface for a system for image retrieval was developed. The developed algorithms need to be integrated so that the system can be a practically useful tool for the forensic footwear examiner.

## 2.4 Dissemination

### 2.4.1 Publications

The following papers were published:

1. V. Ramakrishnan, M. Malgireddy and S. N. Srihari, "Shoe-print Extraction from Latent Images using CRF," *Computational Forensics: Proceedings of International Workshop on Computational Forensics, Washington DC*, Springer LNCS 5158, 2008, pp. 105-112.
2. V. Ramakrishnan and S. N. Srihari, "Extraction of Shoe-print Patterns from Impression Evidence using Conditional Random Fields," *Proceedings of International Conference on Pattern Recognition, Tampa, FL*, IEEE Computer Society Press, 2008, pp. 1-4.
3. Y. Tang, S. N. Srihari and H. Kasiviswanathan, "Similarity and Clustering of Footwear Prints," *Proceedings of International Symposium on Foundations and Practice of Data Mining, GrC 2010*, August 2010, San Jose, CA

4. Y. Tang and S. N. Srihari, "Footwear Print Retrieval System for Real Crime Scene Marks," *Proceedings of International Workshop on Computational Forensics, IWCF*, November 2010, Tokyo, Japan

### 2.4.2 Presentations

Portions of this research were presented at:

1. *SWGTTREAD* Biannual meeting in VA, October 2008.
2. *Foster and Freeman* on July 2, 2009 in Evesham, UK.
3. *Discovery Channel, Canada* made a video of our research in Fall 2009 and it was aired on Canadian television.
4. *NIJ Grantees Meeting* at IAI in Louisville, KY, July 2009.
5. *FBI, Quantico, VA* in February 2010.
6. *NIJ Conference* in Arlington, VA in June 2010.
7. *Foster and Freeman* on July 6, 2010 in Evesham, UK.

## Chapter 3

### References

The references in the Program Narrative are given below. The references are in the order in which they are referred to in the text.



# Bibliography

- [1] Bodziak, W.: Footwear Impression Evidence Detection, Recovery and Examination, second ed. CRC Press (2000)
- [2] Alexander, A., Bouridane, A., Crookes, D.: Automatic classification and recognition of shoeprints. In: Proc. Seventh International Conference Image Processing and Its Applications. Volume 2. (1999) 638–641
- [3] Girod, A.: Computerized classification of the shoeprints of burglar’s shoes. Forensic Science International **1** (1982) 59–65
- [4] Bouridane, A., Alexander, A., Nibouche, M., Crookes, D.: Application of fractals to the detection and classification of shoeprints. In: Proceedings International Conference Image Processing. Volume 1. (2000) 474–477
- [5] Geradts, Z., Keijzer, J.: The image-database REBEZO for shoeprints with developments on automatic classification of shoe outsole designs. Forensic Science International **82** (1996) 21–31
- [6] Mikkonen, S., Suominen, V., Heinonen, P.: Use of footwear impressions in crime scene investigations assisted by computerised footwear collection system. Forensic Science International **82** (1996) 67–79
- [7] Sawyer, N.: SHOE-FIT: A computerised shoe print database. In: Proc. European Convention on Security and Detection. (1995)
- [8] Ashley, W.: What shoe was that? the use of computerised image database to assist in identification. Forensic Science Int. **82** (1996) 7–20
- [9] Foster, Freeman: Solemate. <http://fosterfreeman.com> (2010)
- [10] Bouridane, A.: Imaging for Forensics and Security: From Theory to Practice. First edn. Springer (2009)
- [11] Stone, R.S.: Footwear examinations: Mathematical probabilities of theoretical individual characteristics. Journal of Forensic Identification **56** (2006) 577–599
- [12] Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE Trans. on Pattern Analysis and Machine Intel. **27** (2005) 1615–1630

*BIBLIOGRAPHY*

*BIBLIOGRAPHY*

- [13] Pavlou, M., Allinson, N.M.: Automatic extraction and classification of footwear patterns. In: *Lecture Notes in Computer Science, Proc. Intelligent Data Engineering and Automated Learning*. (2006) 721–728
- [14] deChazal, P., Flynn, J., Reilly, R.B.: Automated processing of shoeprint images based on the Fourier transform for use in forensic science. *IEEE Trans. Pattern Anal. Mach. Intell* **27** (2005) 341–350
- [15] Zhang, L., Allinson, N.: Automatic shoeprint retrieval system for use in forensic investigations. In: *UK Workshop On Computational Intelligence*. (2005)
- [16] Sun, W., Taniar, D., Torabi, T.: Image mining: A case for clustering shoe prints. *International Journal of Information Technology and Web Engineering* **3** (2008) 70–84
- [17] AlGarni, G., Hamiane, M.: A novel technique for automatic shoeprint image retrieval. *Forensic Science International* **181** (2008) 10–14
- [18] Xiao, R., Shi, P.: Computerized matching of shoeprints based on sole pattern. *Lecture Notes In Computer Science; Proceedings of the 2nd international workshop on Computational Forensics* **5158** (2008) 96–104
- [19] Jingl, M.Q., Ho, W.J., Chen, L.H.: A novel method for shoeprints recognition and classification. *International Conference on Machine Learning and Cybernetics* **5** (2009) 2846–2851
- [20] Nibouche, O., Bouridane, A., Gueham, M., Laadjel, M.: Rotation invariant matching of partial shoeprints. *International Machine Vision and Image Processing Conference* (2009) 94–98
- [21] Dardi, F., Cervelli, F., Carrato, S.: A texture based shoe retrieval system for shoe marks of real crime scenes. *Proc. International Conference on Image Analysis and Processing* **5716** (2009) 384–393
- [22] Cervelli, F., Dardi, F., Carrato, S.: Comparison of footwear retrieval systems for synthetic and real shoe marks. In: *Proc. ISA'09 6th Intl. Symp. Image and Signal Processing and Analysis, Salzburg, Austria*. (2009) 684–689
- [23] Dardi, F., Cervelli, F., Carrato, S.: A combined approach for footwear retrieval of crime scene shoe marks. *Proc. ICDP-09, Third International Conference on Imaging for Crime Detection and Prevention, London, UK* (2009) Paper No. P09
- [24] Gueham, M., Bouridane, A., Crookes, D.: Automatic classification of partial shoeprints using advanced correlation filters for use in forensic science. *International Conference on Pattern Recognition* (2008) 1–4
- [25] Wang, R., Hong, W., Yang, N.: The research on footprint recognition method based on wavelet and fuzzy neural network. *International Conference on Hybrid Intelligent Systems* (2009) 428–432

*BIBLIOGRAPHY*

*BIBLIOGRAPHY*

- [26] Patil, P.M., Kulkarni, J.V.: Rotation and intensity invariant shoeprint matching using gabor transform with application to forensic science. *Pattern Recognition* **42** (2009) 1308–1317
- [27] Mikkonen, S., Astikainen, T.: Database classification system for shoe sole patterns - identification of partial footwear impression found at a scene of crime. *Journal of Forensic Science* **39** (1994) 1227–1236
- [28] Huynh, C., de Chazal, P., McErlean, D., Reilly, R., Hannigan, T., Fleud, L.: Automatic classification of shoeprints for use in forensic science based on the Fourier transform. In: *Proc. 2003 International Conference Image Processing*. Volume 3. (2003) 569–572
- [29] Ghouti, L., Bouridane, A., Crookes, D.: Classification of shoeprint images using directional filter banks. *International Conference on Visual Information Engineering* (2006) 167–173
- [30] Su, H., Crookes, D., Bouridane, A.: Thresholding of noisy shoeprint images based on pixel context. *Pattern Recognition Letters* **28** (2007) 301–307
- [31] Crookes, D., Bouridane, A., Su, H., Gueham, M.: Following the footsteps of others: Techniques for automatic shoeprint classification. *Second NASA/ESA Conference on Adaptive Hardware and Systems* (2007) 67–74
- [32] Aitken, C., Taroni, F.: *Statistics and the Evaluation of Evidence for Forensic Scientists*. Wiley (2004)
- [33] I. Evett, J., Lambert, Buckleton, J.: A Bayesian approach to interpreting footwear marks in forensic casework. *Science and Justice* **38** (1998) 241–247
- [34] Biedermann, A., Taroni, F.: Inadequacies of posterior probabilities for the assessment of scientific evidence. *Law, Probability and Risk* **4** (2005) 89–114
- [35] Evett, I.: Towards a uniform framework for reporting opinions in forensic science casework. *Science and Justice* **38** (1998) 198–202
- [36] Otsu, N.: A threshold selection method from gray level histogram. *IEEE Transaction on Systems, Man and Cybernetics* **9** (1979) 62–66
- [37] Ramakrishnan, V., Srihari, S.N.: Extraction of shoeprint patterns from impression evidence using conditional random fields. In: *Proceedings of International Conference on Pattern Recognition, Tampa, FL, IEEE Computer Society Press* (2008)
- [38] Shetty, S., Srinivasan, H., Beal, M., Srihari, S.: Segmentation and labeling of documents using conditional random fields. *Document Recognition and Retrieval XIV SPIE Vol 6500* (2007) 65000U1–9
- [39] Canny, J.: A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **8** (1986) 679–698

*BIBLIOGRAPHY*

*BIBLIOGRAPHY*

- [40] Gonzalez, R.C., Woods, R.E., Eddins, S.L.: Digital Image Processing Using MATLAB 1<sup>st</sup> ed. Prentice Hall (2003)
- [41] Rui, Y., Huang, S., Chang, S.: Image retrieval: Current techniques, promising directions, and open issues. *Journal of Visual Communication and Image Representation* **10** (1999) 39–62
- [42] Srihari, S.N., Huang, C., Srinivasan, H.: On the discriminability of the handwriting of twins. *Journal of Forensic Sciences* **53** (2008) 430–446
- [43] Lowe, D.: Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* **60** (2004) 91–110
- [44] Hough, P.: Machine analysis of bubble chamber pictures. *International Conference on High Energy Accelerators and Instrumentation*, CERN (1959)
- [45] Srihari, S.N., Govindaraju, V.: Analysis of textual images using the Hough transform. *Machine Vision and Applications* **2** (1989) 141–153
- [46] Goulermas, J., Liatsis, P.: Incorporating gradient estimations in a circle-finding probabilistic hough transform. *Pattern Analysis and Applications* **26** (1999) 239–250
- [47] Wu, W.Y., Wang, M.J.J.: Elliptical object detection by using its geometric properties. *Pattern Recognition* **26** (1993) 1499–1509
- [48] McLaughlin, R.: Randomized Hough transform: better ellipse detection. *IEEE TENCON-Digital Signal Processing Applications* **1** (1996) 409–414
- [49] Haralick, R.M., Shapiro, L.G.: *Computer and Robot Vision*. Addison Wesley (1992)
- [50] Bunke, H., Irniger, C., Neuhaus, M.: Graph matching: Challenges and potential solutions. *International Conference on Image Analysis and Processing* **3617** (2008) 1–10
- [51] Sanfeliu, A., Fu, K.S.: A distance measure between attributed relational graphs for pattern recognition. *IEEE transactions on systems, man, and cybernetics* **13** (1983) 353–362
- [52] Bunke, H., Messmer, B.T.: Efficient attributed graph matching and its application to image analysis. In: *Proceedings of the 8th International Conference on Image Analysis and Processing*. (1995) 45–55
- [53] Rubner, Y., Tomasi, C., Guibas, L.J.: The earth mover’s distance as a metric for image retrieval. *International Journal of Computer Vision* **40** (2000) 99–121
- [54] Hillier, F.S., Liebermann, G.J.: *Introduction to Mathematical Programming*. Second edition edn. McGraw-Hill (1995)
- [55] Kim, D.H., Yun, I.D., Lee, S.U.: Attributed relational graph matching algorithm based on nested assignment structure. *Pattern Recognition* **43** (2010) 914–928

*BIBLIOGRAPHY*

*BIBLIOGRAPHY*

- [56] Pelillo, M., Siddiqi, K., Zucker, S.: Many-to-many matching of attributed trees using association graphs and game dynamics. In: Proceedings of 4th international Workshop on Visual Form. (2001) 583–593
- [57] Smith, E., Szidarovszky, F., Karnavas, W., Bahill, A.: Sensitivity analysis, a powerful system validation technique. *The Open Cybernetics and Systemics Journal* **2** (2008) 39–56
- [58] Aldenderfer, M., Blashfield, R.: *Cluster Analysis*. SAGE (1984)
- [59] Tang, Y., Srihari, S.N., Kasiviswanthan, H.: Similarity and clustering of footwear prints. In: *IEEE Symposium on Foundations and Practice of Data Mining (GrC 2010)*, IEEE Computer Society Press (2010)
- [60] Nixon, M., Aguado, A.: *Pattern Extraction and Image Processing*. Elsevier Science (2002)
- [61] NAS: *Strengthening the Forensic Sciences: A Path Forward*. National Academy of Sciences press (2009)