Master Thesis

A Framework for Robust Forensic Image Identification

Public Version

Maurits Diephuis

August 18, 2010

Dr.	Andreas Wombacher	Twente University	(Supervisor)
Prof.	Sviatoslav Voloshynovskiy	Université de Genève	(Supervisor)
Dr.	Maurice van Keulen	Twente University	(Co-Supervisor)
ir.	Fokko Beekhof	Université de Genève	(Co-Supervisor)

UNIVERSITÉ DE GENÈVE
FACULTÉ DES SCIENCES
Département d'informatique

University of Twente
Enschede - The Netherlands

Preface

This is the end. But until that time, I have this allocated place to myself to ponder on how it all came to be and on who helped me along the way. I came to Geneva to get my degree and to climb in the Alps every weekend. Pending who I talked to, I would change the priority of these two things. That, and I needed a place to live. Later would I realize that the first real bed I would sleep in, after two weeks of camping in the living room of **Fokko Beekhof**, was a hospital bed. A sunny window and a lot of drugs insured that I had a whale of a time, although it also meant that the climbing season was over. **Prof Sviatoslav Voloshynovskiy** summarized it all by mentioning that the whole episode would probably be of great benefit to my academic work.

Prof. Sviatoslav Voloshynovskiy I actually met a year earlier when I had come to Geneva to finish an elective project, and yes, to climb. After seeing icy pictures of such an adventure by his PHD student **Fokko Beekhof** and me, he instructed me, that I may not engage in dangerous activities with his research staff. This is how I got to know the Stochastic Information Processing group in Geneva. They provided me with a desk and the missing link to my elective project, which incidentally I was doing for **Andreas Wombacher**. A new senior lecturer at Twente University, he quickly gathered a lot of interesting problems and students together, and I was very lucky to be one of them. **Andreas Wombacher** provided me with more then one interesting problem, a desk, guidance and lots of enthusiasm for nearly my entire master. With him behind the wheel I would eventually return to Switzerland and to the SIP group when I was scouting around for a master assignment. At the SIP group for a

second time, I was quizzed by Prof. Sviatoslav Voloshynovskiy who fired of questions from a pattern recognition book. Imagine my relieve when I recognized the book [19]. Written by **Ferdi van der Heijden** from the Signals and Systems group who was also the lecturer and tutor behind the Minor Imaging. It was he who introduced me to the world of imaging and imaging measurement systems and the more abstract way of modeling thinking about him. His problems gave me a steepness night or two, and in return I found something I truly liked.

This leaves me with the endless list. The list of people who helped me to get there, or where just always there when needed. In complete random order: [1] Sam en Stieneke Diephuis, Annemijn, Fokko Beekhof, Sviatoslav Voloshynovskiy, Oleksiy Koval, Andreas Wombacher, Ferdi van der Heijden, Andre du Croix, Vroni Wiesgickl, het Schaap, Sebas, Laksmi, Farel and Sedarta Graber, Taras Holotyak, Eva Wentink, Sander Bockting, Jadd Khoury, Ciska Kurpershoek, Benoit Deville, Jurgen Braams, Farzad Farhadzadeh, Joep Kierkels, Edward Akerboom, Stephan Kruisman, Mohammad Soleymani, Ander Erburu, Suzanne Rau, Anneke van Abbema, Erica van der Wiel Bert Groenman, Brigitte Bogaards, Bart van der Wal and Viola Mashoed.

[1] Actually you are sorted by the procedure detailed in 5.4.1

Abstract

This work has researched the possibilities and limitations of using physical micro-structures for identification. Using the physical structure itself from an object negates the need to add a special marker for identification. Buchanan *et. al.* [10] found that paper documents, packaging and plastic cards contain microscopic surface structures that are unique for the sample. The naturally occurring randomness forms both a unique and currently unclonable identification token.

The work addresses two main research questions. The first part of this work is devoted to an imaging algorithm that always extracts the exact same patch of micro-structure from an acquired sample, independent of how the acquisition was done. The second part of this work deals with the identification system as a whole and explores the fundamental bounds of an identification system based on micro-structures.

Image Synchronisation

Image synchronisation is achieved by stipulating that some fixed part of the originating sample, such as a letter or logo must be in the field of view. This fixed template, is used both in the enrollment and verification stage to ascertain the exact region from which the micro-structure is extracted. Any occurring geometrical transformation needs to be corrected. An algorithm has been developed. It is based on an edge map and invariant feature points in combination with Hough pose-space clustering and RANSAC for robust approximation. This

algorithm can deal with heavily corrupted datasets that generate false matches.

Identification

The second part explores the suitability of a micro-structure as a unique and random identification token. The primary assumption is that micro-structures are independently and identically distributed ($i.i.d$) random samples. This enables us to completely model the system using Shannon's communication model. The primary test that was deployed to ascertain the suitability of micro-structures and the quality of synchronisation is the empirical determination of the *intra* and *inter* class distance probability density functions. The results for a number of test datasets are very promising and validate acquiring a realistically large set.

The single most damaging factor to micro-structure quality is blur. In al cases the blurring acts as a low pass filter, which removes detail from the micro-structures and induces local correlations.

Future Work

Future work will primarily focus extracting fingerprints from micro-structures that achieve dimension reduction, a certain invariance to geometrical transformations and that significantly speed up database queries. Random projections and Magnitude Sorting are showing positive indicators that they can drastically limit the query search space.

Contents

L'alpiniste est un homme qui conduit son corps là où, un jour, ses yeux ont regardé. Et qui revient.
– Gaston Rébuffat (1921 - 1985)

Chapter 1

Introduction

1.1 Identification and Forensics

Imagine the following scenario: A shipment of medicine of dubious origin is intercepted by customs. Counterfeited drugs are a relative new but steadily increasing problem. The US Food and Drug Administration loosely estimates that about 15 percent of all sold medicines are fake. In parts of Asia and Africa these estimates go up to 50 percent [12]. Customs now stands the task to determine the origin of the intercepted drugs. The officer therefore takes a novel approach. He takes his mobile phone, and with the aid of a small magnifying lens, he photographs one of the medicine boxes up close. This picture of the box its surface structure is sent via MMS to a database server. The server compares the received picture against a database with pictures from medicine boxes originating from the authentic manufacturer. As the micro-structure from the package is unique, much like a human fingerprint, the system quickly identifies whether the intercepted medicine box is authentic or counterfeited.

The essence of the problem is the so-called *identification* problem. We aim to identify a binary sample, possibly corrupted by noise, using a database of original samples. The key assumption is that the surface microstructure that is photographed and binarized has *unique forensic features*, and is *unclonable* [10]. Further more, it should be possible to extract the correct region from

the presented image in order to compare structures from the same place on the object. See Figure 1.1 for an overview.

Traditional anti-forgery methods are based on some form of watermarking, meaning that the product is altered by a proprietary process that is hard to replicate. Examples are special dyes and inks or holograms. Obviously there are a lot of scenarios in which product alteration is as expensive as it is undesirable; luxury goods for example. The beauty of the microstructure method is that it requires no special process, no product alternation or manufacturing change to apply the anti-forgery method, as it is the product itself that is being used for identification. Best of all, methods to manufacture physical objects that have identical micro-structures currently do not exist [10].

Microstructure matching can be applied to any application whose samples have surface structures that are sufficiently unique and that can be automatically localized and photographed. Metal spare parts for aircraft, the plastic and paper of identification credentials, credit cards, the list of micro-structure sources and thus forensic applications is endless.

Our approach is insidious as it plays on the strength of the forgery rather than its weakness. It uses present features, such as a logo, to correctly extract the fingerprint region. The better the forgery, the better the extracted fingerprint with the micro-structure becomes, and thus the reliability of the resulting reject becomes high. Sloppy forgeries might make it very difficult to ascertain the fingerprint region in the image. But then again, failure to ascertain a fingerprint is an automatic reject.

1.2 Scenario

The usage scenario and corresponding architecture can be seen in Figure 1.1. This section will highlight a number of key elements of the system that will be researched in this work.

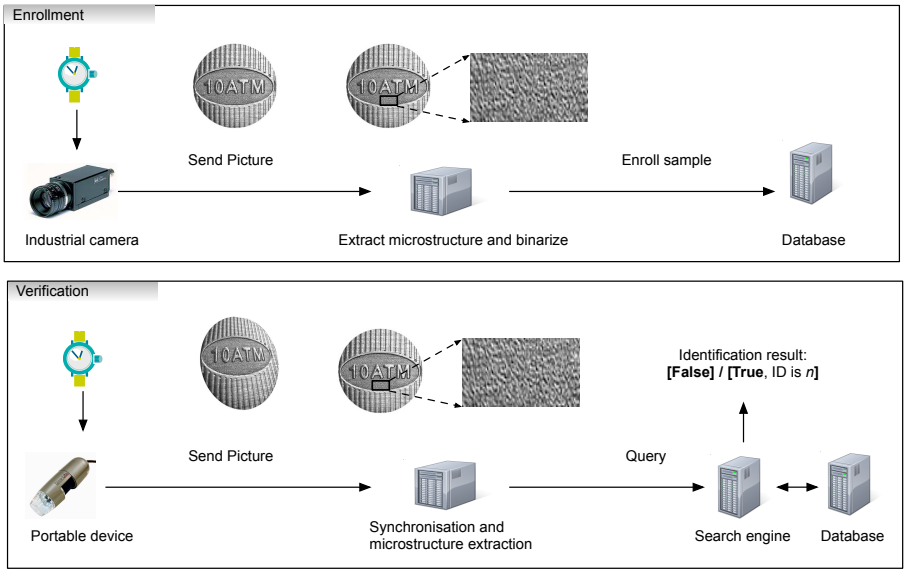


Figure 1.1 – Schematic figure of enrollment and identification architecture.

1.2.1 System Architecture

The basic architecture, as seen in Figure 1.1, consists of two stages, *enrollment* and *verification*. In the enrollment stage, samples are enrolled into the database. Products are photographed using a high quality industrial camera. A digital fingerprint, such as a hash, is extracted from all the photos and is stored. In the verification stage, a product sample is photographed with a cheap imaging device. Software then extracts a region of interest from the image and post processes it into a binary query. This query is run against the database for identification. Obviously two major factors come into play:

- The gap in imaging between acquisition and identification. Bridging this gap in image quality will be denoted as the *synchronization* problem. Part 1 of this work is devoted to it.
- The fast identification of image fingerprints. This process involves all things related to the probability of false identification, the number of distinguishable fingerprints, security and privacy. It is the focus of part 2 of this thesis.

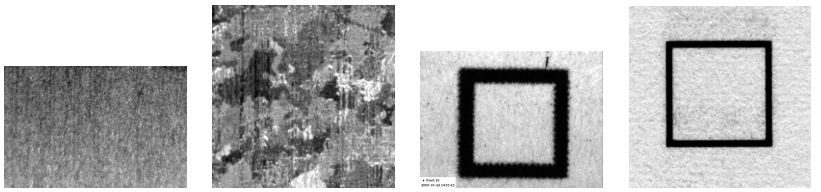
1.2.2 Acquisition and Synchronization

Synchronization software will bridge the imaging capture difference between enrollment in the factory and identification in the field. It basically provides two mechanisms.

- Ascertain the correct region of interest to extract the microstructure from. This is done via a part of the sample that does not change, such as a logo.
- Post process the acquired microstructure to eliminate differences in lighting, contrast and resolution.

1.2.3 Micro-structures

Traditionally, anti-forgery methods for physical objects are based on some form of watermarking. Examples are the application of invisible dyes and inks, holo-



(a) (b) (c) (d)

Figure 1.2 – Examples of optically acquired micro-structures. Figures 1.2a and 1.2b are aluminum samples, 1.2c and 1.2d are paper samples.

grams and the watermarks found on money bills [15, 54]. The disadvantage of such (proprietary) techniques is that they need to be applied specially during fabrication of the protected product and that specialized equipment is needed to read out the watermark. Using an object's micro-structure negates this entire process. Furthermore, there is currently no known process to fabricate identical micro-structures [10].

The usage of natural occurring randomness in physical objects to ascertain a token or identifier is not new. Buchanan *et. al.* [10] found that paper documents, packaging and plastic cards contain microscopic surface structures that are unique for the sample. They use so-called *laser speckle* [16] to capture surface structures. Using cross-correlation as the matching metric, they report accurate identification of their samples which include paper that is soaked in water and plastic credit cards. Smith *et al.* [51] designed and built an optical device to acquire micro-structure images from paper. They identified individual samples by comparing them against a list of original samples from a database also using cross-correlation. Although successful, this approach is not feasible for large databases as it becomes computational infeasible to compare all samples with this metric. The patent from Kariakin [1] therefore proposes to calculate a hash from all the samples.

1.2.4 Digital Fingerprinting

Digital fingerprinting refers to all methodologies to project objects in some high-dimensional space onto a lower one in such a way that the objects can still be identified or compared. The most common example of course is the hash function. The most immediate advantage is the reduced memory and storage footprint. This in terms makes it possible to exhaustively search large lists and leads to greatly enhanced security and privacy.

This work will look at digital fingerprinting algorithms with an extra requirement, namely algorithms that next to dimensionality reduction also retain a certain invariance to geometric distortions. This immediately rules out all cryptographic hashes.

1.2.5 Identification

Identification in this framework denotes a form of nearest-neighbor query in a very high dimensional search space. The nearest neighbor algorithm that in this case amounts to maximum likelihood (ML) decoding, is visualized in Figure 1.3.

A binarized microstructure image is essentially a noisy signal sample. Naturally there is an intrinsic relation between the noisiness of the sample and the sample properties (entropy) versus the amount of samples that can be successfully identified and retrieved. This relation translates to the following database parameters:

- Database scalability
- Retrieval complexity and speed
- Retrieval accuracy

Database scalability refers to the number of samples the database can hold. Although in theory this number may be only bounded by the amount of computing resources one can allocate, there is a catch. First, the larger the number of samples, the more complex retrieval becomes. The framework contains binary signals that must be matched, and in principle, this means that for a single

query all entries must be compared linearly. Application of meta data (Section 3.3.2) and smart coding techniques (Section 5.4.1) for which the index file can be kept in memory will speed up the process, but the lack of tree-like structures in this type of database is a limitation to retrieval speed.

Secondly, the larger the amount of samples, the greater the probability for collisions i.e. the fact that the retrieval engine will return more than a single result but rather a set of which the entries are all equally likely to match the query sample. This can in some part be amended by introducing smart decoding techniques and matching metrics (Chapter 5) but the principle remains. It is of course trivial to see why in forensic applications, be it fingerprints, biometrics or micro-structures it is very undesirable to have false accepts.

As stated in the introduction we wish the database to be able to work with the biggest possible query deformations. The framework should be able to deal with query images that have been acquired with a small portable micro-scope under varying circumstances. This process is modeled with Shannon's noisy channel model [50, 14, 33] seen in Figure 1.5. For the specific application of the micro-structures this framework thus contains an imaging component that aims to restore the distorted image query ($\boldsymbol{y}$ in Figure 1.5) to the best possible estimate of the sample original (Chapter 2).

These informal database parameters can be modelled formally, as introduced in Section 1.3.1. These formal parameters will be the universal benchmark throughout this work.

1.3 Research

1.3.1 Theoretical Framework

A binarized microstructure image is essentially a (noisy) signal. Its components or pixels X, with alphabet $\mathcal{X}$ can be seen as realizations ('draws') from a distribution $p(\boldsymbol{x}) = Pr[X = \boldsymbol{x}]$. Identification of a noisy query $\boldsymbol{y}$ against a database holding M samples can be modeled as maximum likelihood decoding.

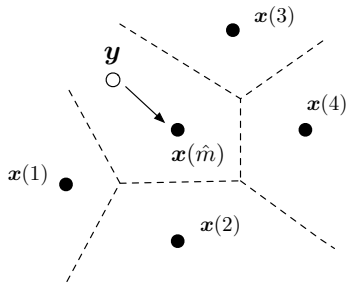


Figure 1.3 – Visualization of maximum likelihood (ML) identification in an n dimensional search space.

The estimated identification index $\hat{m}$ is ascertained as follows:

$$\hat{m} = \underset{1 \leq m \leq M}{\arg\max}\, p(\boldsymbol{x}(m)|\boldsymbol{y}) \tag{1.1}$$

Schematically this model can be seen in Figure 1.5, it is known as Shannon's communication model. Information theory [14, 33], most notable the work of Shannon [50], provides the mathematical tools to model this system. The primary assumptions, as seen schematically in Figure 1.4, is that micro-structures are formed from random independently and identically distributed samples. Shannon's work can be used to analyze this type of system. Amongst other things it can prove what the number of samples is, that a database with such samples can hold without collisions, what amount of noise the system can withstand, and how big the retained binarized fingerprint should be. An overview of the database parameters together with their mathematical descriptors can be seen in Figure 1.6. These elements of the information theory together with Shannon's channel model will therefore be used to determine the theoretical bounds of the database system. It is the focus of chapter chapter:fundamental and 4, and they will consistently be used in this work to test and validate system performance.

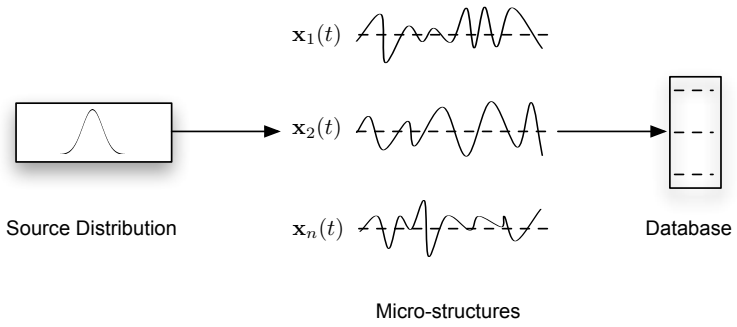


Figure 1.4 – Schematic of the basic assumption that micro-structures are independently and identically distributed (*i.i.d.*) variables drawn from some source distribution.

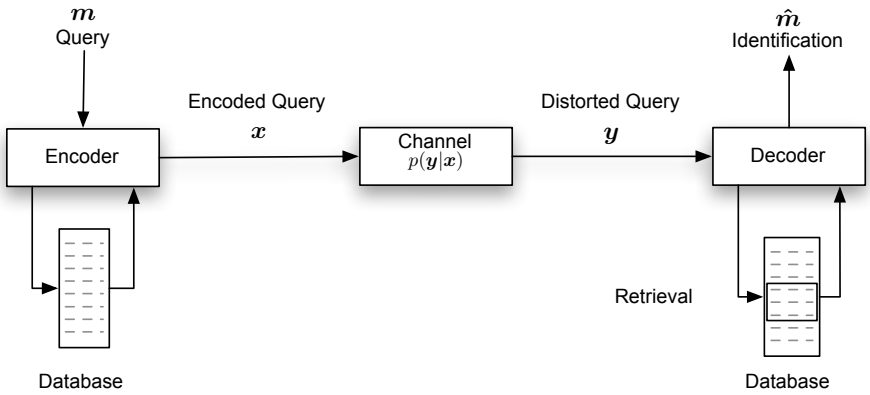


Figure 1.5 – Schematic of the noisy database retrieval framework modelled with Shannon's communication model.

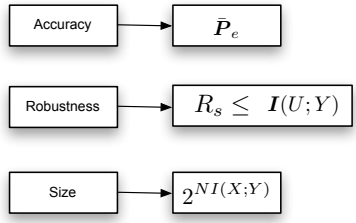


Figure 1.6 – Fundamental parameters of the identification framework together with their mathematical definitions.

1.3.2 Research scope and questions

In the scope of this thesis, the noisy samples are less than perfect acquired images. Part of this work is thus devoted to processing the distorted image samples that framework is presented with. The aim is to correct any kind of geometrical and lighting distortion that might occur when the query image is acquired. Formally, this work aims to maximize the empirical mutual information between the original sample and the distorted ones. In practice, this means that the framework aims to extract the best possible micro-structure from the correct image region and post-process it. The first research question therefore becomes:

- *How can the geometrical distortions and imaging artifacts be removed between an acquired image and a sparse template?*

The second part of this work deals with the identification system as a whole and explores the fundamental bounds of such a framework given the real data at hand. The two main concerns are *database scalability* and *identification accuracy*. The main research question that thus will be answered for every dataset, imaging device and processing algorithm therefore is:

- *What is the maximum number of unique distinguishable samples the database will hold?*

1.3.3 Research validation

Research validation of the two primary research questions will broadly follow the following methologies. To validate the imaging algorithm's ability to correct and restore geometric transformations we will look at the dataset statistics before and after restoration, specifically at the *intra-class* distance between different observations originating from an identical sample. Distortions will occur naturally as most datasets were acquired manually by a person who operates a small imaging device without any tripod or other fixture.

Validation of the second research question uses two interlinked steps. In all (biometrical) pattern identification problems the absolute key issue is the *inter-* and *intra-* class variance. These two properties determine if patterns can be identified without error [18, 19]. Identification is only possible if the variability of multiple instances of a single object is less than the variability between different objects. If this property is satisfied the intra-class distance can than be used to model the maximum number of distinguishable objects.

1.4 Thesis Overview

Chapter 3 introduces the reader to all the information theory that is needed to understand Shannon's noisy channel theorem. It then proves the fundamental limit in terms of sample quality and the number of distinguishable samples in (any) database system. Chapter 2 deals with all tried, tested and enrolled imaging techniques that the framework utilizes to correct the distorted image samples it receives as query. Decoding and retrieval strategies are covered in Chapter 4 and 5. And finally, framework tests and conclusions can be found in Chapter 6.

1.5 Contribution

The main contributions of this work are:

- Conception and implementation of the feature-based synchronisation algorithm, as described in Chapter 2, Section 2.4.
- Implementation and testing of the framework that ascertains the validity of using micro-structures for identification in large databases.
- Deployment, implementation and testing of random projections on actual micro-structures..

1.6 Notation

The following notation based on [14] and [58] is used throughout this work:

x	Scalar variable, realization of random variable X
$\boldsymbol{x}$	Vector variable
x^N or $\boldsymbol{x}$	Equivalent to $\{x[1]....x[N]\}$
X	Scalar random variable
$\mathcal{X}$	Alphabet of discrete random variable X
$P(X)$	Probability density function of X
$p(x)$, $p_X(x)$ or $Pr[X = x]$	Probability mass function
X $p(x)$	Discrete random variable X is distributed as $p(x)$
$\mathcal{N}(\mu, \sigma^2)$	Gaussian probability density function
$\mathcal{G}$	Gaussian kernel
$\mathcal{L}$	Laplacian kernel

Table 1.1 – Notation.

Part I

Imaging

If your pictures aren't good enough, you aren't close enough.
– Ernő Friedmann (1913 - 1954)

Chapter 2

Image Synchronisation

2.1 Introduction

Image synchronisation is the process of finding a transformation between a distorted and a target image. This chapter will therefor answer the first research question:

> *How can the geometrical distortions and imaging artifacts be removed between an acquired image and a sparse template?*

The template that is used to ascertain the geometrical distortions is part of the image that is presumed to be fixed. This can be a shape such as the part of a logo, or a predominant texture. The imaging artifacts are all differences that occur when a human operator acquires an image under varying circumstances.

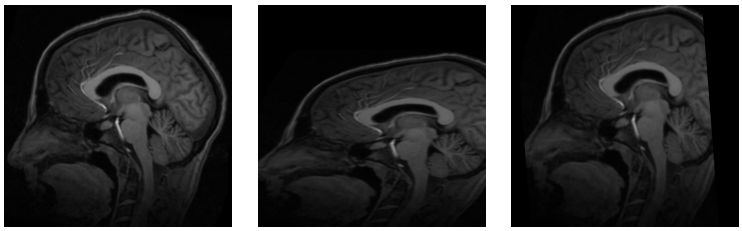
Most common synchronisation techniques [63] fall into the following categories:

- The family of gradient based methods
- Methods based on invariant image features
- Methods that operate in the Fourier domain

Gradient driven methods are commonly referred to in (medical) literature as image registration. It involves the process of finding some transformation that aligns a distorted or sparse image against a target image. It is widely used for medical images. Applications include synthesizing images originating from different modalities such as PET and MRI or compensating for patient movement while an image is acquired. Gradient based methods originated from Paul Viola in [57] and [56]. The primary assumption is that if two images are aligned perfectly, the mutual information between the images is maximized. To find the actual transformation this family of algorithms all use numerical optimization methods that search for global extrema of a function by manipulating a given the set of parameters from the geometrical transformation in terms of the error metric. It common practice, in medical applications, to have an operator who inputs corresponding control points in both the distorted and the target image. The primary drawback of these methods is that next to very being computational expensive, there is no guarantee that the obtained extrema in parameter-space is global. In other words, the found mapping doesn't isn't always the best mapping from the distorted to the target image. An example of a gradient based registration in an MRI image can be seen in Figure 2.1. This example uses a common optimization method called *simplex* [43].

Invariant feature based methods seek some kind of image feature that is invariant to geometrical and lighting distortions. There are predominantly based on image derivatives that are calculated in scale-space image decompositions [61, 31, 27]. They can be local extrema, corners, edges or curves. Invariant feature based methods are the focus of Section 2.3.

Many registration methods operate in the Fourier domain as this domain exhibits a number of properties under translations and scaling that make it extremely useful [36]. Specifically, the magnitude spectrum is translation independent, but scale and translation properties are carried into the Fourier domain. See Figure 2.2 for an example.



(a) Original. **(b)** Distorted **(c)** Registered

Figure 2.1 – Basic non rigid image registration on a MRI scan using a gradient method. This specific function uses the *simplex* algorithm [43]. These figures can be generated with `demo_affine_registration.m`.

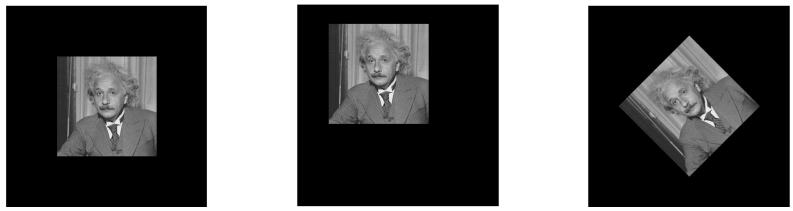
2.2 Acquisition

Currently four different imaging devices are tested for their suitability within our identification framework. Two of them, are handheld devices. The two others are industrial camera's that are usually found in industrial applications such as wear detection and product inspection. The most important factors for micro-structure acquisition are:

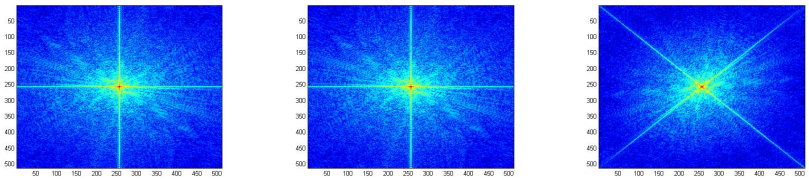
- The field of view (FOV)
- The magnification factor
- The Depth of Field (DOF)
- The lighting

Field of view and magnification

The field of view, depth of field and the magnification are known to be conflicting requirements. Obviously, the larger the magnification, the better the micro-structures can be acquired. However, our framework needs the synchronization template to also be in the field of view, preferable in the center. This



(a) Original. **(b)** Translated **(c)** Rotated



(d) Original. **(e)** Translated **(f)** Rotated

Figure 2.2 – Basic example how various affine transformations in the real domain are reflected in the fourier domain. These figures can be generated with `demo_fourierdomain.m`.

means that significant magnification is only possible if the template is small, but more important, of very high (printing) quality. The more template imperfections the microscope detects, the more these imperfections influence the imaging algorithms further in the pipeline. In all, but the case of the Aluminum dataset, we have chosen to aim for maximal magnification in order to capture the microstructure at its best.

lighting

Experiments where done with four basic types of lighting:

- Type A LED ring light
- Diffused ring lights

- Direct lighting
- Infrared lighting
- Ultra Violet lighting

The choice of lighting is driven by a number of requirements:

- The lighting must bring out sufficient micro-structure detail
- The lighting must be invariant to minor changes in the overall setup
- If possible, it must mimmic the lighting of the hand held imaging devices

Figure 2.3 shows how different lighting influences the acquisition. It is immediately apparent that *direct lighting*, as seen in Figure 2.3 brings out the most detail. The shallow directional light brings out all small textures and ridges in the sample. However, the major drawback is that the acquired images become highly specific for the used imaging and lighting device. In one of the tests with the Dataset 14 dataset we acquired a sample set with a directional light mounted on a small purpose-built tripod. The results in terms of the *intra* and *inter* class distance where excellent and exceeded the results from the set that was acquired with a diffused ring light. However when the operator disassembled and reassembled the lighting setup, the resulting difference was enough to produce a set that could no longer be matched error-less against the ring light acquired set. This invalidates the use of direct lighting.

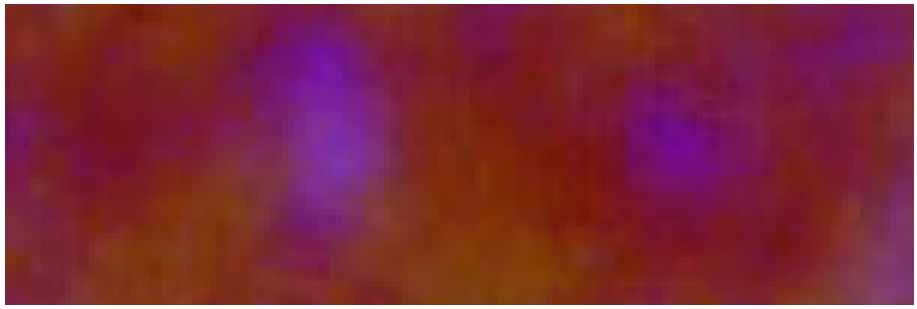
Dataset 14 was acquired with both a diffused ring and a normal Type A LED ring light. Figure 2.3 shows two different acquisitions that differ considerable. The diffused ring light proved to give the most stable results, but considerably blurs the image. As will be shown in Chapter 4 this blurring damages micro-structures.

Camera A is the handheld device that was used to acquire most of the datasets. It uses five un diffused led lights in a semi circle. It provides reasonable results, but still blurs more than Camera B in combination with the Type A ring light.

Figure 2.3 shows an extracted and synchronized patch of micro-structure from an the identical sample from Dataset 14 seen in Figure 2.3 .

(a) Camera B, broad diffused LED ring.

(b) Camera B, Type A LED ring.



(c) Camera A

Figure 2.3 – Examples of different lighting conditions and the resulting micro-structures. These samples are the extracted micro-structures from the images.

2.3 Feature based Registration

2.3.1 On Scale-space and Affine features

This Section will cover the general theory of scale-space representation of images. The scale-space representation allows for image operations to be made scale invariant. This is a necessary step in many applications to be able to deal with the variations in size of real world objects, and the varying distance between them and the capturing device.

Imagine a picture of a tree. The tree-feature is only meaningful at a certain scale, namely the one in which you can see a good proportion of the tree. The leaves of the tree only become distinct features the moment of zooms in on the picture or takes the picture close up. Although not completely true, this a useful example to understand how one can have different image features at different scales.

A single image in scale-space is represented as a set of images that are successively smoothed by a kernel. The kernel is usually a Gaussian and the smoothing parameter σ is often referred to as the *scale parameter*. The witz being that image features at scale σ that are smaller than $\sqrt(\sigma)$ have been smoothed away. See Figure 2.7 for an example.

The scale-space representation was first proposed by Witkin, [61]. Major contributions to scale-space theory and image structures were made by Koenderink [27] and Lindeberg [31].

2.3.2 Gaussian scale-space

For a 2D image $f(x, y)$ the Gaussian scale-space is defined as follows:

$$\mathcal{L}(x, y; \sigma) = (\mathcal{G}_\sigma * f)(x, y) \tag{2.1}$$

$$\mathcal{G}_\sigma(x, y) = \frac{1}{2\pi\sigma} exp^{\frac{-(x^2+y^2)}{2\sigma}} \tag{2.2}$$

Here $*$ denotes convolution at image point (x, y) at scale parameter σ. Note that scale 0, or $\sigma = 0$ results in a devision by 0. Informally, scale 0 denotes the original image in most literature.

Of vital importance is the fact that the used kernel for smoothing the image may not introduce new image artifacts when generating coarser scale images from finer scales. Extensive research has been done by [61, 31, 27, 30, 53] to prove the fact that the Gaussian kernel is the optimal kernel for constructing scale-space representations from discrete digital images. The Gaussian kernel has a number of other useful properties, namely separability, linearity, causality and it is semi-group like. We will briefly address these properties

The separability entails that a multi dimensional Gaussian kernel can be derived from the product of one dimensional Gaussian kernels, i.e.

$$\mathcal{G}(x, y) = \mathcal{G}(x)\mathcal{G}(y) \tag{2.3}$$

This means that a 2D image can be smoothed by two one dimensional Gaussians: one for each dimension. This is beneficial as one dimensional Gaussian filers can be implemented very efficiently.

The causality property stipulates that no additional image structures of any kind may be introduced when deriving the coarser scales. The coarser scale must only be a simplified representation of the original, or finer scaled images.

The Gaussian kernels exhibits a semi group like commutative property. Successive smoothing of an image $\boldsymbol{I}(\boldsymbol{x})$ with n kernels yields the same result as a single smoothing operation with a single kernel whose size equals the sum of the same n kernels. Formally:

$$\mathcal{G}(\sigma_1) * ... * \mathcal{G}(\sigma_n) * \boldsymbol{I}(\boldsymbol{x}) = \mathcal{G}(\sigma_1 + ... + \sigma_n) * \boldsymbol{I}(\boldsymbol{x}) \tag{2.4}$$

Building the scale-space set can be accelerated by sampling coarser images by their corresponding scale factor. This results in a pyramid like structures as seen in Figure 2.4b. This approach is for example used by Lowe's SIFT algorithm [32]. The downside to this approach is the fact that locating points through scale-space becomes slightly more complicated and the fact that the sampling may introduce aliasing.

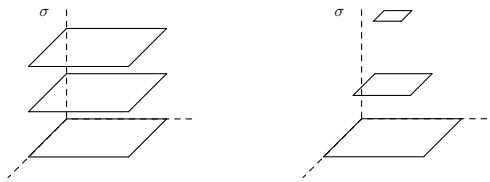


Figure 2.4 – Scale-space pyramids.

2.3.3 Scale-space derivatives

Image derivatives play a very important role in many image feature detectors such as Harris [20] and SIFT [32] points to name just a few. To this end we will take a look at the second order Taylor expansion of an image derivative. Assuming the image is differentiable and ignoring image boundary issues it is defined as:

$$\boldsymbol{I}(\boldsymbol{x} + \Delta\boldsymbol{x}) \approx \boldsymbol{I}(\boldsymbol{x}) + \Delta\boldsymbol{x}^T \nabla \boldsymbol{I}(\boldsymbol{x}) + \Delta\boldsymbol{x}^T \mathcal{H}(\boldsymbol{x}) \Delta\boldsymbol{x} \tag{2.5}$$

Where Δ is the image gradient, and $\mathcal{H}$ is the Hessian matrix. The image gradient is also known as the *second moment matrix* and is utilized extensively for feature detectors. The Hessian matrix is a relative newcomer to the image feature field. It is for example used in recognizing the so called *characteristic scale*. Both features will be addresses in detail in Section 2.3.4.

2.3.4 The Second moment and Hessian matrix

The second moment matrix describes the gradient distribution in a region of interest (ROI) around some image point. The gradient is approximated by convolving the image with a Gaussian function. Furthermore all gradients are smoothed by using some kind of window function, usually also a Gaussian kernel. The eigenvalues of the matrix denote the principal curvatures of that point. The stronger the curvatures the more the signal (or image) changes in orthonormal

directions. See Figure 2.7 and 2.8 for an example. This is indicative for a corner or an edge. Its most famous application is probably in the Harris [20] corner detector (Appendix A.1). Formally:

$$M(\boldsymbol{x}, \sigma_I, \sigma_D) = \sigma_D^2 \mathcal{G}(\sigma_I) * \begin{bmatrix} \mathcal{L}_x^2(\boldsymbol{x}, \sigma_D) & \mathcal{L}_x\mathcal{L}_y(\boldsymbol{x}, \sigma_D) \\ \mathcal{L}_x\mathcal{L}_y(\boldsymbol{x}, \sigma_D) & \mathcal{L}_y^2(\boldsymbol{x}, \sigma_D) \end{bmatrix} \tag{2.6}$$

The Hessian matrix is build from the second order image derivative approximation in the following way:

$$\mathcal{H}(\boldsymbol{x}, \sigma_D) = \sigma_D^2 \begin{bmatrix} \mathcal{L}_x^2(\boldsymbol{x}, \sigma_D) & \mathcal{L}_x\mathcal{L}_y(\boldsymbol{x}, \sigma_D) \\ \mathcal{L}_x\mathcal{L}_y(\boldsymbol{x}, \sigma_D) & \mathcal{L}_y^2(\boldsymbol{x}, \sigma_D) \end{bmatrix} \tag{2.7}$$

It denotes the changes in the normal vector of the image isosurface. Various components of this matrix, such as the trace are used in feature detection by [45, 47, 27]. The trace specifically, is the Laplacian Filter. It detect regions where the image changes rapidly and as such is often used in edge detection algorithms. Often the image is smoothed first by convolving it with a Gaussian function to reduce noise sensitivity. This removes high frequency components prior to differentiating the image, These two filters can be combined with what is commonly referred to as the *Laplacian of Gaussian* (LoG) filer. Formally for an image $\boldsymbol{I}$ the Laplacian is:

$$\nabla \boldsymbol{I} \quad = \quad trace(\mathcal{H}) = \mathcal{L}_x^2(\boldsymbol{x}, \sigma_D) + \mathcal{L}_y^2(\boldsymbol{x}, \sigma_D) \tag{2.8}$$

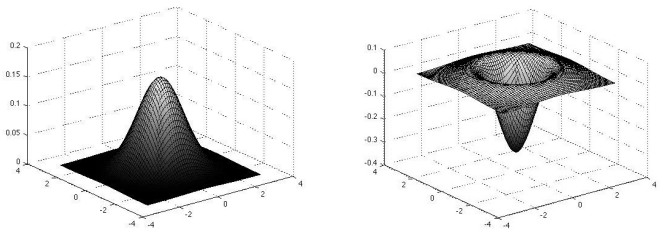
The LoG filter then becomes:

$$LoG(x, y, \sigma) \quad = \quad \nabla \mathcal{G}(x, y, \sigma) = \frac{x^2 + y^2 - 2\sigma^2}{2\pi\sigma^6} exp(-\frac{x^2 + y^2}{2\sigma^2}) \tag{2.9}$$

The LoG filter is attractive as it can be approximated efficiently by the so called Difference-of-Gaussian (DoG) filter:

$$LoG(x, y, \sigma) \approx \frac{1}{\sigma^2(k-1)} (\mathcal{G}(x, y, \sigma k) - \mathcal{G}(x, y, \sigma)) \tag{2.10}$$

The DoG filter is applied through out scale-scale by numerous feature detectors, most notably by SURF[2] en SIFT [32] points.



(a) (b)

Figure 2.5 – Standard 2 dimensional Gaussian and Laplacian kernel. See `demo_show_kernel.m`.

2.3.5 Features points

Two popular and extensively used features that are based on the second moment Hessian matrix and the Laplacian of Gaussian pyramid are *Harris* points [20] and *SIFT* features [32, 8]. Both algorithms are covered in detail in Appendix A.1 and A.2. Although more complex and computational more intensive, this framework uses SIFT features in favor of Harris points. They have proven to be more invariant to lighting and acquisition conditions as SIFT features fall within the edge maps instead of on edges as the Harris points do. The fact that SIFT points seek minima that are surrounded by multiple ridges (edges) makes them more stable than points that cling to edges alone.

2.4 Feature based Registration Algorithm

A schematic overview of the deployed algorithm can be seen in Figure 2.8. It has four major components, namely:

- Edge detection
- SIFT feature detection
- SIFT feature matching

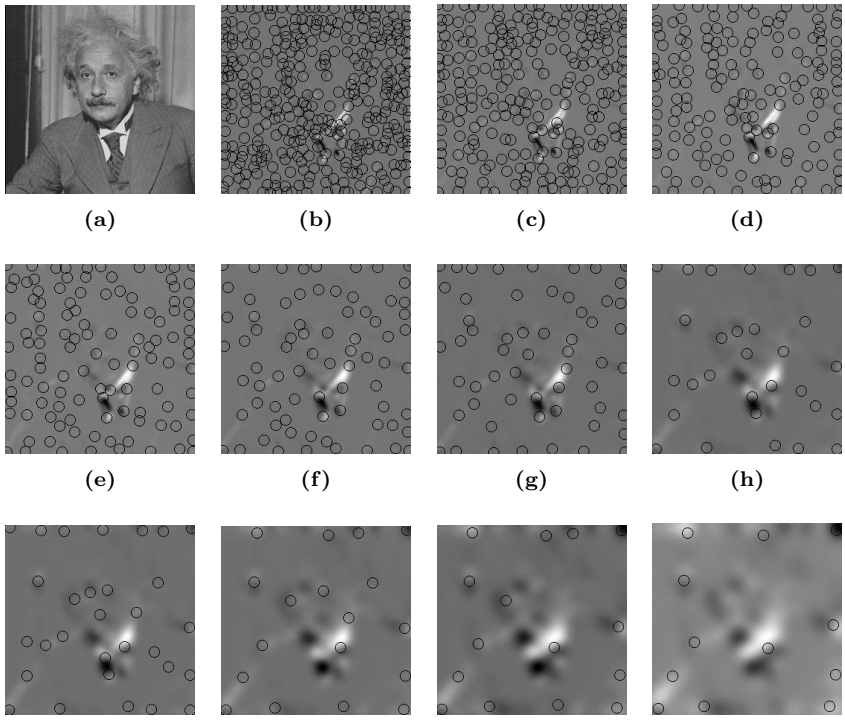


Figure 2.6 – Harris features and Scale-space. Images generated with `demo_char_scale1.m`.

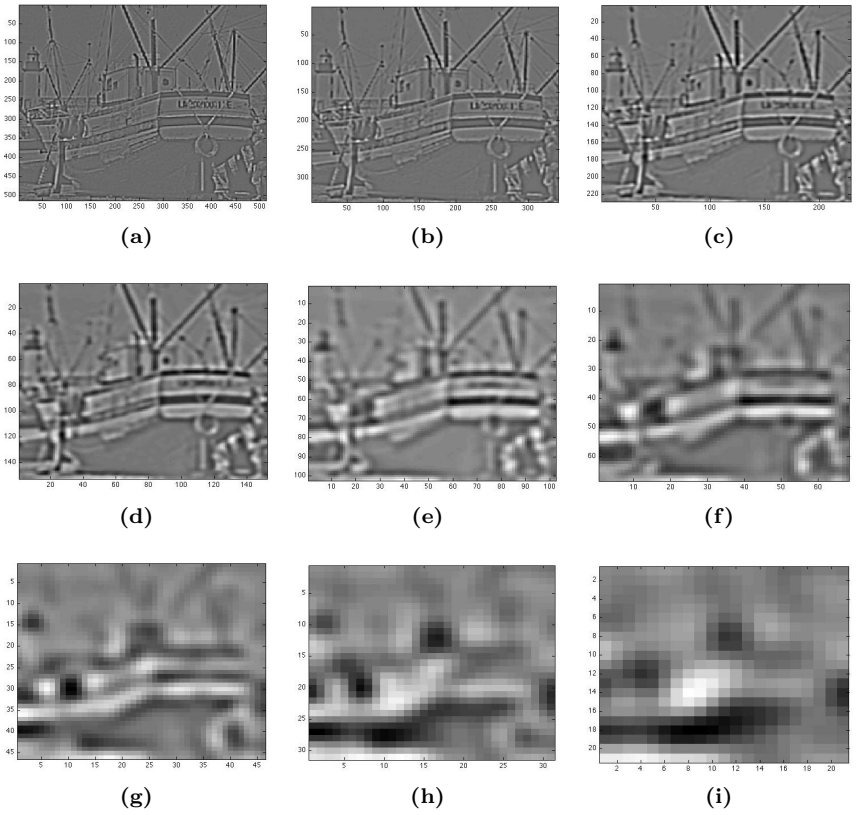
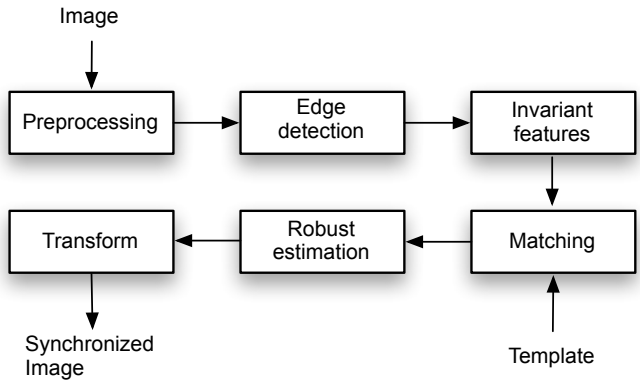


Figure 2.7 – Scale-space pyramid where the LoG operator has been approximated by DoG. This pyramid is utilized by the SIFT algorithm. See `demo_scale_space1.m` to generate this particular pyramid.



(a)

Figure 2.8 – Schematic of the invariant feature based synchronisation algorithm.

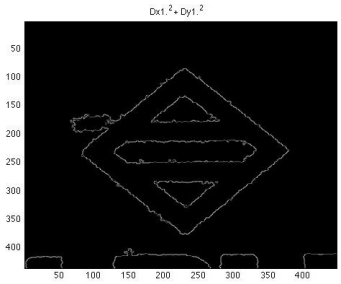
- RANSAC to discard erroneous matches.
- The Direct Linear Transform

2.4.1 Edge detection

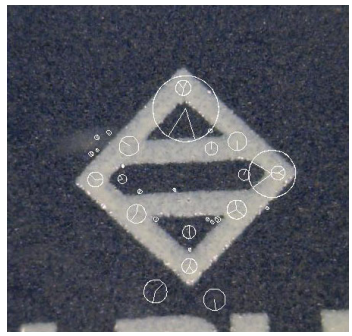
Edge detection is done with Laplacian of Gaussian (LoG) Filter, mentioned earlier in Section 2.3.4 and Equations 2.8 and 2.9.

2.4.2 SIFT feature detection

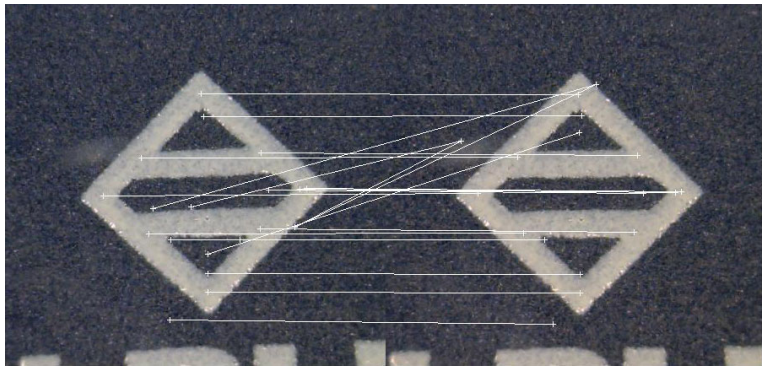
SIFT features were conceived by Lowe and Brown [32, 8]. A SIFT point is made up of four parameters and a data vector. The four parameters are the x and y coordinate, the *scale* and the *orientation*. The data or *descriptor* vector is an 128-bit value that is made up from eight histograms from the region of interest (ROI) around a SIFT point. One can view the descriptor vector as an advanced template taken from the SIFT point region. Matching individual SIFT points is done on the basis of the so called *descriptor vector*.



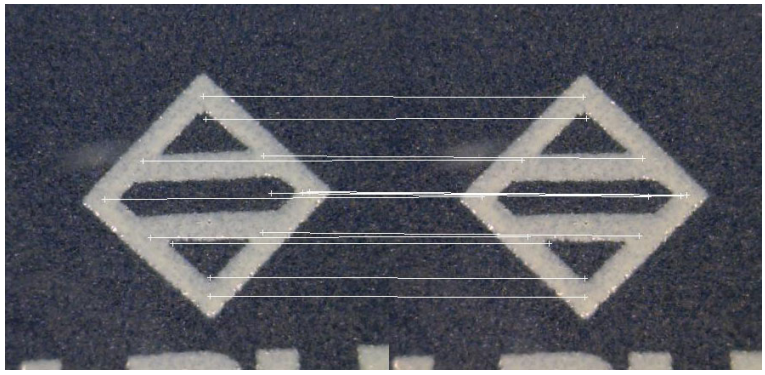
(a) Edge map



(b) Found features



(c) Initial matching



(d) Robust matches

Figure 2.9 – Trial run of the synchronisation algorithm of Section 2.4 and the front plates of the 10ATM dataset. Note how in sub figure 2.9b the features all within the edge regions instead of on the edges themselves.

A more extensive study on the matching process of descriptor vectors and the database issues this raises can be found in Appendix B.3. Essentially, the problem is equal to the *nearest neighbor search*.

The implemented method in the framework, as proposed by Lowe [32], is based on an exhaustive nearest neighbor search. All SIFT descriptor vectors from the image are matched against the descriptor vectors from the model. It's two main characteristics are:

- The distance metric is the angle between the two normalized descriptor vectors i.e. $\cos\theta = a \cdot b/|a||b|$.
- A match is only accepted if the best match is at least 60 percent better then second candidate.

2.4.3 Inferring the projective transformation

There are a number of methods available to find the predominant projective transformation that occurs between SIFT point-correspondences. One could ascertain the homography between point correspondences using the Direct Linear Transform (DLT) [21] algorithm and use that as a basis for clustering. The downside is that this method does not harness the full SIFT point potential as it only uses the point coordinates, discarding *scale* and *orientation* information. Alternatively one could employ the Random Sample Consensus (RANSAC) algorithm. RANSAC is a robust estimation technique. Its 'robustness' lies in the fact that it can fit a model to a dataset that is polluted with erroneous samples that follow some unknown error distribution. In this application we would wish to infer the homography (the model) between two sets of matching SIFT points. One of the drawbacks of using RANSAC is that the algorithm performs badly when the number of inliers drops below 50 percent. An alternative that does harness the full SIFT point potential is so called *Hough pose-space* clustering. Simply put, this is a form of four dimensional histogramming.

Hough pose-space clustering is attractive as it can deal with heavily corrupted datasets. In our tests is was capable of dealing with datasets that had up to 70 procent of erroneous matches. Despite the good results, the are some

drawbacks to pose-space clustering. The pose-space is huge and requires a lot of memory. Pending the bucket size, the 4D accumulator requires a bigger address space then a 32 bit platform can provide.

A good solution is to set up Hough pose-pase clustering with broad bins and to use the algorithm to clear the dataset of the worst noise. RANSAC can then be used to discard the remaining outliers and determine the optimal affine transformation. This approach gives stable results.

Hough Pose-Space

Each correspondence match between images results in four parameters per SIFT point: *x, y, scale* and *orientation*. A match thus gives eight parameters in total. To detect both clusters belonging to different objects and to reject outliers, a four dimensional dataset is created. This dataset is commonly referred to as the *Hough pose-space*. See Appendix B.5 for the precise definition.

Clustering is done using a Hough-like 4D accumulator array. Each pose-pase pair not only votes for its own 4D bucket but also for the buckets in the neighborhood. The bucket neighborhood is sphere like.

The next step is to to detect local extrema. All points voting for (neighboring) buckets where local extrema occur are clustered into groups. Points falling outside these buckets are classified as outliers, as are points from buckets that have a limited number of votes.

RANSAC

A short description of the RANSAC algorithm will be given below. For full details see [21]. Various implementations can be found in [28].

1. Select a random sample s from a dataset S. The model m is initialized with subset s.

2. Determine what subset S_i of points from S is within a certain distance d from the model. S_i is called the consensus set and defines the inliers of S.

3. If the number of inliers, S_i, is greater the threshold T, the model m is re-estimated using the inliers and the algorithm terminates.

4. If the number of inliers, S_i is less the the threshold T, the algorithm selects a new subset and goes back to step 2.

5. If the maximum number of sample & model steps N has been reached, the largest subset S_i (that thus contains the largest number of inliers) is selected. The model is re-estimated using that model.

This overview leaves some fundamental choices unanswered. The choice of the model m, the threshold T for the number of required inliers, the metric to use to determine the distance d between de model m and the subset S_i and the sample size N. The model used is the affine transformation. The distance threshold d is chosen so that α is the chance that a point (or match in this case) is an inlier. It can be determined empirically, but following [21] it is assumed that the error follows a Gaussian distribution with $(\mu, \sigma) = (0, \sigma)$. Now the square of the distance d between a point and the model is the sum of squared Gaussian variables, which in itself is a χ^2_m distribution. The degree of freedom m is two in the case of the affine model. So given t^2 from the χ^2_2 cumulative distribution the choice of deciding what distance comprises an out- or inlier becomes:

$$\begin{cases} d^2 < t^2 & \text{inlier} \\ d^2 \geq t^2 & \text{outlier} \end{cases} \tag{2.11}$$

From [21] we use $5.99\sigma^2$ for t^2 and 0.95 for α. The final question is the choice of N, the number of samples to be drawn from set S. Generally it is chosen such that at least one of the subsets S_i drawn from S doesn't contain any outliers at all with a probability p. This probability is usually set to $p = 0.99$. So given the fact that w is the probability that a point is an inlier, we define $\epsilon = 1 - w$ as the change that this point is an outlier. The number of selections N of size s becomes:

$$N = \frac{log(1-p)}{log(1-(1-\epsilon)^s)} \tag{2.12}$$

Direct Linear Transform

Given a set of i 2D point correspondences $\{\boldsymbol{x}_i \leftrightarrow \boldsymbol{x}'_i\}$, these correspondences are used to determine the 2D homography matrix H such that $\boldsymbol{x}_i = H\boldsymbol{x}_i$ using the Direct Linear Transform (DLT) [21]. For details about DLT see Appendix B.6.

2.5 Fourier based Registration

All Fourier based registration methods leverage the fact that affine transformations in the real domain manifest themselves in certain predictable ways in the Fourier domain. Given an distorted image $\boldsymbol{d}$ that is related to a template image $\boldsymbol{t}$ by some affine matrix A. Then every pixel in $d(x_d, y_d)$ is mapped by A to a pixel $t(x_t, y_t)$. Following [36] this can be formally expressed as:

$$\boldsymbol{d} = A\boldsymbol{t} \tag{2.13}$$

$$\begin{pmatrix} x_d \\ y_d \\ 1 \end{pmatrix} = \begin{pmatrix} s\cos\theta & -s\sin\theta & \Delta x \\ -s\sin\theta & s\cos\theta & \Delta y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_t \\ y_t \\ 1 \end{pmatrix} \tag{2.14}$$

Which can be refractured into:

$$t(x, y) = d(\Delta x + s \cdot (x\cos\theta - y\sin\theta), \Delta y + s \cdot (x\sin\theta + y\cos\theta)) \tag{2.15}$$

The affine parameters one wishes to infer are the translation given by Δy and Δx, the scaling s and the rotation θ. [40] .

Following the *shift theorem* that states that the Fourier magnitude spectrum is translation invariant as a shift of function $f(x)$ by α merely multiplies the Fourier transform by $e^{-i\alpha\mu}$:

$$f(x) \xrightarrow{\mathcal{F}} \mathcal{F}\{u\} \tag{2.16}$$

$$f(x - \alpha) \xrightarrow{\mathcal{F}} \mathcal{F}\{u\}e^{-i\alpha\mu} \tag{2.17}$$

For the two dimensional case the magnitude spectrum is defined as the product of the two dimensional Fourier transform $\mathcal{F}\{u, v\}$ times its complex conjugate

$\mathcal{F}^*\{u, v\}$. Given that:

$$f(x - \alpha, y - \beta) \xrightarrow{\mathcal{F}} \mathcal{F}\{u, v\}e^{-i(\alpha\mu+\beta v)} \tag{2.18}$$

The power spectrum $\mathcal{F}\{u, v\}\mathcal{F}^*\{u, v\}$ for $f(x - \alpha, y - \beta)$ becomes:

$$e^{-i(\alpha\mu+\beta v)}\mathcal{F}\{u, v\}e^{-i(\alpha\mu+\beta v)}e^{i(\alpha\mu+\beta v)}\mathcal{F}\{u, v\} \tag{2.19}$$

Which, as the $e^{-i(\alpha\mu+\beta v)} * e^{i(\alpha\mu+\beta v)} = 1$, equals the unshifted, zero phase, spectrum:

$$\mathcal{F}\{u, v\}\mathcal{F}^*\{u, v\} \tag{2.20}$$

The translation is, however, reflected in the phase of the Fourier transform, as can be seen in Equation 2.18.

The *similarity theorem* states that the Fourier transform of a scaled version of a function $f(x)$ by a parameter α: $f(\alpha x)$ results in the shrinking or expanding of the Fourier transform of that function $f(x)$ by the reciprocal of that factor α. Formally:

$$f(\alpha x, \beta y) \xrightarrow{\mathcal{F}} \frac{1}{|\alpha\beta|}\mathcal{F}\{\frac{\mu}{\alpha}, \frac{v}{\beta}\} \tag{2.21}$$

In the case of our affine matrix, the scaling factor s is equal for x and y direction, so the resulting factor becomes s^2.

Finally, the *rotation theorem* states that any rotation of a two dimensional signal is fully reflected in the resulting magnitude spectrum. It rotates through the same angle. Rotating function $f(x, y)$ by angle θ becomes:

$$\begin{aligned} &f(x\cos\theta + y\sin\theta, -x\sin\theta + y\cos\theta) \quad \xrightarrow{\mathcal{F}} \\ &\mathcal{F}\{\mu\cos\theta + v\sin\theta, -\mu\sin\theta + v\cos\theta\} \end{aligned} \tag{2.22}$$

2.6 Post processing and Comparison Metrics

2.6.1 Post processing

The framework uses the following post processing steps pending the specific database used.

- Global mean removal
- High pass filtering
- Histogram equalization

High pass filtering seems a bit of an unlikely step as common knowledge usually dictates that the most informative content of images resides in the lower frequency bands. In the case of the R label dataset high pass filtering was deployed as observations from unequal labels gave unusual high cross correlation values as can be seen in Figure 4.7a. This means that all observations carry a component with a large magnitude that influences the resulting cross correlation value, and suppresses any minor similarity that might occur when observations originate from equal labels.

2.6.2 Comparison Metrics

There are a number of basic metrics in usage for comparing micro-structures:

- Euclidian distance
- Cross-Correlation
- Term weighted cross correlation

Term weighted cross correlation is defined as cross-correlation with including the last term of the Euclidian distance:

$$\rho_{\boldsymbol{xy}} = \boldsymbol{y}\boldsymbol{y}^T - 2\boldsymbol{y}^T\boldsymbol{y} + \boldsymbol{x}^T\boldsymbol{x} \tag{2.23}$$

$$\rho'_{\boldsymbol{xy}} = \rho_{\boldsymbol{xy}} - \frac{1}{2}\boldsymbol{x}^T\boldsymbol{x} \tag{2.24}$$

The motivation behind this metric follows Section 2.6.1. Its intent is to prevent high magnitude components that are prevalent throughout the entire dataset to unruly increase the cross correlation value between observations from unequal labels. Termweighting gave greater inter and intra class separation for Dataset 3 as can be seen in Figure 4.8a.

2.7 Validation

The level of precision that is required for synchronisation within this framework is high; it easily exceeds popular stitching software. The latter, to our knowledge, can not deal with sparse images, nor does it achieve accuracy beyond a visually appealing image. For that reason, printing synchronized images in this work, will not prove much, as even badly synchronized images in our standards will look pixel perfect with standard printing quality.

This validation section will focus on the applicability and robustness of the two synchronisation algorithms.

The final verdict will be based on a number of mathematical tools for validation. These tools are the focus of Chapter 3 and Chapter 4.

2.7.1 Feature based synchronisation

In general the feature based based algorithm will fail in the following scenarios:

- The attained edge map is not stable between different observations.
- The template does not offer enough features to ascertain an affine matrix.
- The template is symmetrical and the resulting features become ambigu in their matching.

Unstable edge map

The edge map is a small but vital step for the feature based algorithm. Edge features are also a bit notorious for their sensitive character. Especially when (numerical) image derivatives are used to determine the edges. Of course there exists a big variety of algorithms to ascertain edges that contain morphological steps and a priori shape knowledge. This does make it possible to extract and trace template shapes in most cases. However, all such steps make the algorithm increasingly specific for the image dataset and template used, and worse, specific for the condition in which the image was enrolled. For example, the sort of lighting.

Sparse features

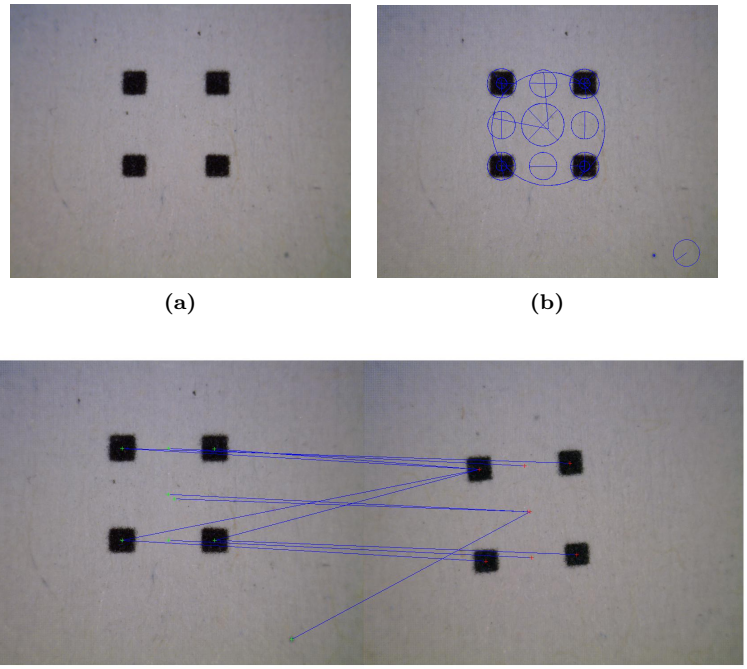
Pending the specific geometrical transformation that needs to be corrected between two images, a number of feature points needs top to be present. For an affine transformation as determined by the DLT algorithm one needs at least five. As no feature will be perfect and small deviations are possible, one would like much more stable features over the required five such that Hough pose space clustering can be used to rid the set of outliers and RANSAC can be used to find a general solution amongst the inliers.

The majority of our datasets were acquired using Camera A microscope. This microscope suffers from significant lens distortions on the towards the edges of its field of view. This creates distortions that are non linear. Therefore only features that are in the center of the field of view of the microscope are useable when deploying a transformation model that assumes an affine type deformation.

Symmetrical features

Feature matching is based, in most cases, on the direct neighborhood of a feature. These regions are binned in to histograms and compared using the euclidian distance (standard SIFT point method) or compared exhaustively using cross correlation (most Harris point implementations). In any case, the images in this framework are the edge maps. The surrounding micro-structure is unique for the sample and can there for not be used to ascertain a match. This means that the region descriptors of the features are quiet sparse as they only draw information from the surrounding edges. This is a problem the moment the dataset has a template that is symmetrical. Features will then have regions that are similar while their position is not. This problem occurs in the Dataset 4 dataset and can be seen in Figure 2.10.

This situation can probably be amended by switching to graph based matching, as proposed in Section 2.8.



(a) **(b)**

(c)

Figure 2.10 – Examples of failed image synchronisation for **Dataset 4** using invariant features due to symmetrical and thus ambiguous features.

2.7.2 Closing Thoughts

The feature based algorithm needs a good edge map, This requirement requires morphological operations to a certain extend. Morphological operations are all thresholds deep down and they should be avoided in any robust versatile framework. With the datasets that are currently enrolled, edge detection has proved to be a sensitive operation, and therefore more research such be done.

On the positive side, the feature based algorithm can work with big templates, such as entire logo's and whole brand names.

2.8 Future Work

Phase Congruency

The feature based algorithm (Section 2.4) relies heavily on a correct edge-map from the template features. Traditional gradient based edge detection methods such as Canny [11] or Sobel [52] are sensitive to illumination and blurring.

More recently, Kovesi [29, 28] advocated the use of *phase congruency* to detect corners and edges. Key idea is that all image features such as edges and corners give rise to points where the Fourier components of that image are maximally in phase. The biggest single advantage is that these edge features are invariant to changes in image lighting and contrast. Phase congruency is a dimensionless quantity, and as such it allows for more universal thresholds to classify edges. The biggest disadvantage is that phase congruency is more complex and vastly more computational intensive to calculate than the traditional gradient based edge detection methods.

Sub-pixel sampling

Sub-pixel sampling is advocated and used by many researchers [9]. The common approach is to take all luminance values from a certain region of interest around a feature point. These values are then used to analytically determine a second order quadratic surface. With 9 points this amounts to an elliptic paraboloid. From this function the extrema can easily be calculated and the

extrema's position becomes the new sub-pixel accurate position of the initial feature point.

Sub pixel sampling was implemented and tested on the Dataset 11. Due to the vary nature of the fitting, this method of sup pixel sampling is very sensitive to any pixel value variation in the region around the initial feature. Even although Dataset 11 was acquired with an industrial camera under very stable conditions, sub pixels sampling gave wildly off results to due small variations in luminance values. It is therefore not used in any framework in this work.

Graph based matching

Features are currently matched based on the image edges in their direct surroundings. This poses some restrictions on the choice of feature, as its surrounding must be distinct enough to support an initial matching between features in two images. The alternative is to match features with a graph based approach. The features in both images can viewed as a fully connected graph between which similarity measures can be calculated. Difficulties arise quickly as graph points in one image may have undergone an affine transformation. Further more, there might be feature points missing all together. The problem of matching two relational graphs can be transformed into the equivalent problem of finding a maximal sub-clique in a derived association graph. This problem is NP-complete [44, 41, 13]. Even so, there exists a large family of heuristics and algorithms that are able to find good approximations. These methods have originated from the astronomy world to match stars within a view against the entire map, both for star recognition as for space ship orientation [34, 62]. Currenty, this approach is deemed the most promising and experiments are in progress.

Image quality assessment

As will be shown in the final validation in Chapter 4 one of the single most damaging things that can occur while acquiring a micro-structure is image blur. or mall focussing of the acquiring device. High frequency components are vital for successful identification and blurring also increases the mutual information

between observations from different samples. Detecting blurred images can be done, in principle, by looking at the frequencies in Fourier domain. The more blurred, the more high frequency components will disappear, although in reality detecting this is somewhat more complicated. Similar work is done in the Iris recognition domain [18], although commercial systems seem to take a rapid succession of images.

Print quality

As the designated micro-structure region must be extracted precisely, the sample quality must be very high. Specifically, the template may not be ambiguous. All, but the Aluminum dataset (5.9d) suffered from templates that where not consistently made. Besides the limitation of the used imaging device, and the printing resolution, template variations hurt performance the most. It causes ambiguity when matching against a template, i.e, there were multiple optimal fits between the target image and the template. Attempts to categorize the type of deformations such that the algorithm can estimate with what situation it is dealing did not lead to anything. As shown in Chapter 4, Section 4.2 and Figure 4.7a the results for this dataset are the worst. The R dataset is therefor also the subject of all explorations into invariant fingerprints in Chapter 5.

The second noteworthy factor is the used printer driver. When printing the template of the Dataset 4 or the SIP database, it quickly became apparent that results vary wildly between Adobe drivers and the ones provided by Microsoft. The latter printing squares in different shapes and forms.

Symmetric transfer error

Peak positions are naturally also suspect to measurement errors, meaning that for perfect matched image correspondences $\{\boldsymbol{x}_i \leftrightarrow \boldsymbol{x}'_i\}$ there is no single homography H that maps one set to the other. In the case of an overdetermined solution one can resolve to iterative methods to minimize some cost function that expresses this error. In the case where errors occur in both images, one

can use the symmetric transfor error [21]:

$$\sum_i d(\boldsymbol{x}_i, H^{-1}\boldsymbol{x}'_i)^2 + d(\boldsymbol{x}'_i, H\boldsymbol{x}_i)^2 \tag{2.25}$$

Tests show that the DLT algorithm from five and nine points leaves some residual rotation. Implementing an iterative schema using *Levenberg-Marquardt* as outlined by [21] using the symmetric transform error as cost function did not improve our results.

Part II

Identification

I just wondered how things were put together.
– Claude Shannon (1916- 2001)

Chapter 3

Fundamental Aspects of Noisy Databases

The identification architecture identifies noisy samples using a database with original noise-free samples. Obviously there is a fundamental relation between the size of the samples, the noise, the number of samples in the database and the probability of error, i.e. the chance of false accepts and rejects. In this chapter we will formally explore this relation using information theory [33, 14] and Shannon's noisy channel model.

3.1 Introduction to Information Theory

Information theory started by answering two fundamental questions in computer science. What is the theoretical achievable data compression, and what is the ultimate achievable data transmission over a communication channel? [1] Specifically information theory tells us how one can achieve near-perfect communication over a noisy channel. In this context, the *noisy channel* is the conceptual name for any input-output system that suffers from some kind of error e.g.

[1] The answers are the *entropy* H and the *channel capacity* C.

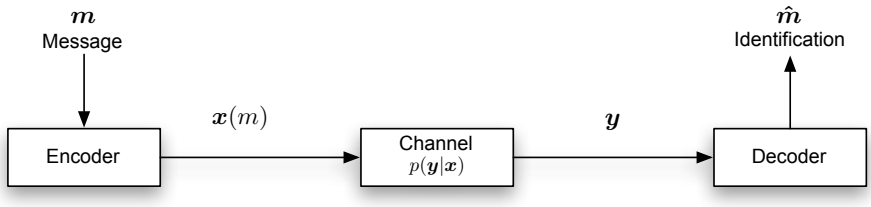
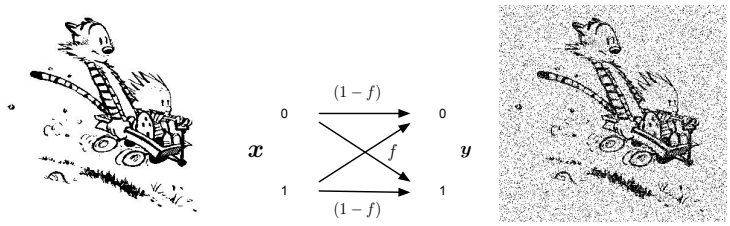


Figure 3.1 – The memory less Channel model.



(a) Original **(b)** Channel **(c)** Distorted

Figure 3.2 – Noisy channel model with $p_b = 0.1$. The figures can be generated with `demo_nc1.m`.

- The communication from earth with Spirit and Opportunity, the two rovers currently on planet Mars.
- The hard or optical drive in your pc, that is prone to reading and writing errors.
- Mutations that occur when viruses or any other cell replicates.

A schematic overview of the channel concept can be seen in Figure 3.1. In our context the channel represents the probabilistic likelihood that an entry in the database matches a query.

Figure 3.2 shows an example, adapted from [33], of a noisy channel that is commonly referred to as the *binary symmetric channel*. This channel receives

binary input sequences $\boldsymbol{x}$ and outputs $\boldsymbol{y}$, but flips the individual input bits with probability p_b. Using coding theory we can encode the message m with error detecting and correcting codes in such a way that we can decode the received message without error even though the channel flips the odd bit. Information theory deals with the fundamental limits of such systems in terms of the noise and the number of bits that need be transmitted for flawless communication. We will explore the binary channel example by introducing a trivial code that simply replicates the input three times before sending. The decoder simply counts the number of zeros and ones in a message and decodes the message to the most occurring symbol. That means that decoding only goes wrong in two distinct cases. When all three message bits are flipped or, three instances where two out of three bits are flipped. The probability that this occurs is:

$$p_e = 3p_b^2(1 - p_b + p_b^3 \tag{3.1}$$

In our example with $p_b = 0.1$ the employed code lowers the probability of error from 0.1 to 0.028, an improvement of factor 3.57. The bad news is that the lower probability of error comes at a price of having to send three times as many bits through the channel. The *rate* of communication has gone done from 1 to 1/3. The relation between the rate R of the binary channel and the probability of a bit flip p_b can be seen in Figure 3.3a. Obviously, there is a fundamental relation between rate R and the probability error p_e. It was widely believed that the boundary between achievable and unachievable (p_e, R) pairs was a curve starting from the origin. For one, that meant that in order to achieve a negligible p_e that the system would suffer from an extremely low rate R.

However, in 1948, Claude Shannon published the seminal paper "A Mathematical Theory of Communication" [50] in which he proved that for every channel it is possible to communicate with arbitrary small error p_e and that the curve between achievable and unachievable (p_e, R) pairs starts at a point $R = C$ and not the origin, where C is the *channel capacity*. This epic result is known as the *noisy-channel coding theorem*.

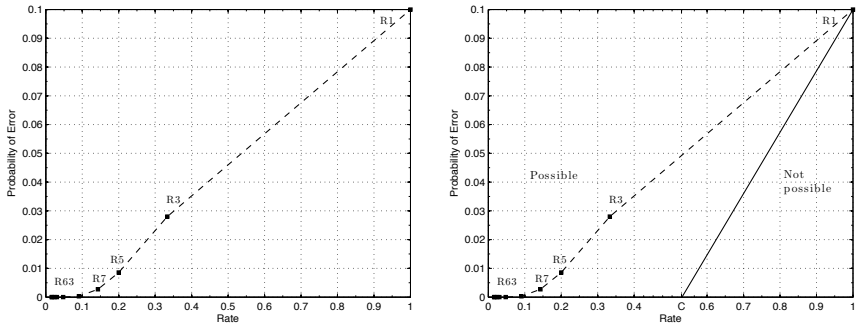


Figure 3.3 – The probability of error versus the rate for the replicating code $R_i, i = \{63, 7, 5, 3\}$ and the binary symmetric channel with $p_b = 0.1$. This figure can be generated with `demo_nc2.m`.

To continue our example with the binary channel with $p_b = 0.1$. The channel capacity C for this particular channel can be ascertained via:

$$\begin{aligned} C(p_b) &= 1 - H_2(p_b) \\ &= 1 - \left[p_b \log_2 \frac{1}{p_b} + (1 - p_b) \log_2 \frac{1}{1 - p_b} \right] \\ &= 0.53 \end{aligned} \tag{3.2}$$

The resulting curve for the binary channel can be seen in Figure 3.3b.

3.2 Concepts and Building blocks

This section briefly describes all the mathematical and information theoretic concepts that are used later in this chapter.

3.2.1 Entropy and Mutual Information

For a discrete stochastic variable X, with alphabet $\mathcal{X}$ and probability mass function $p(x) = Pr\{X = x\}$, the entropy is defined as:

$$\begin{aligned} H(X) &= -E_X\left[\log p(x)\right] \\ &= -\sum_{x\in\mathcal{X}} p(x)\log_2 p(x) \end{aligned} \tag{3.3}$$

Entropy is a measure of uncertainty of a stochastic variable. It is therefore also a measure of the average amount of bits needed to describe a stochastic variable.

Extending the case to two stochastic variables X, Y we are interested in their degree common uncertainty i.e. their joint probability density function $p(x, y)$. This notion is formalized via their joint entropy $H(X, Y)$:

$$\begin{aligned} H(X,Y) &= -E_{XY}\left[\log p(x,y)\right] \\ &= -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log p(x,y) \end{aligned} \tag{3.4}$$

Joint entropy is a measure of randomness in a system of two stochastic variables Y and X. In the same spirit one can define the *conditional entropy* $H(X|Y)$ which is a measure for the uncertainty that remains about x once y is known. Formally:

$$\begin{aligned} H(Y|X) &= \sum_{x\in\mathcal{X}} p(x)H(Y|X=x) \\ &= -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log(p(x|y) \end{aligned} \tag{3.5}$$

Following the *chain rule*, the joint entropy $H(X, Y)$ can be expressed in terms of the conditional entropy for variables X and Y:

$$\begin{aligned} H(X,Y) &= H(X) + H(Y|X) \\ &= H(Y) + H(X|Y) \end{aligned} \tag{3.6}$$

If X and Y are independent, knowing X will not learn one anything about Y. As a consequence, the joint entropy then becomes:

$$H(X,Y) = H(X) + H(Y) \tag{3.7}$$

Informally this means that in order to describe $p(x, y)$ from the independent X and Y, one needs to describe both $p(x)$ and $p(y)$ individually as they share nothing. This leads to the concept of *mutual information*, the measure of the amount of information that one stochastic variable shares with another stochastic variable. Given two random variables X and Y with a joint probability density function $p(x, y)$ and marginal probability mass functions $p(x)$, $p(y)$, the mutual information $I(X;Y)$ is defined as:

$$I(X;Y) = H(X) + H(Y) - H(X,Y) \tag{3.8}$$

If one refractors Equation 3.8 with Equation 3.3 and 3.4 the definition for mutual information becomes:

$$I(X;Y) = \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{3.9}$$

In the channel context (Figure 3.1), the mutual information tells one how much information the channel output Y conveys about channel input X.

Example

Consider the example from Figure 3.2 and Section 3.1 again in which we send a cartoon through a channel without the usage of any encoder. This binary channel then receives symbols from the input alphabet $\mathcal{X}$ that is comprised solely of zeros and ones. These zeros and ones are the black and white pixels that make up the picture. The empirical probability mass function of this particular picture can be found be simply counting and normalizing the pixel values:

$$p(x) = \begin{cases} Pr[X = 0] = 0.097 & \text{black pixel} \\ Pr[X = 1] = 0.9029 & \text{white pixel} \end{cases} \tag{3.10}$$

Note that $X = 0$ denotes a black value. Furthermore it should be noted that as we only have one particular picture to work with, it is impossible to model the true source distribution whose realizations are the different cartoons.

Using mutual information one can now calculate how much information is conveyed by the binary channel that has a probability of a bit flip of $p_b = 0.1$.

$$\begin{aligned} I(X;Y) &= H(X) + H(Y) - H(X,Y) \\ &= H(Y) - H(Y|X) \end{aligned} \tag{3.11}$$

Where $H(Y|X)$ is the weighted sum over all x for $H(Y|x)$. Here it can be calculated via:

$$P[Y=1] = p_b \cdot Pr[X=0] + (1-p_b) \cdot Pr[X=1] \tag{3.12}$$

$H(Y)$ can than be calculated using the *binary entropy function*

$$\begin{aligned} H_2(p) &= H(p, 1-p) \\ &= p \log \frac{1}{p} + (1-p) \log \frac{1}{(1-p)} \end{aligned} \tag{3.13}$$

The mutual information than becomes:

$$\begin{aligned} I(X;Y) &= H(X) + H(Y) - H(X,Y) \\ &= H(Y) - H(Y|X) \\ &= H_2(0.82) - H_2(0.1) \\ &= 0.68 - 0.47 = 0.21 \end{aligned} \tag{3.14}$$

Considering that the entropy of the source equals about 0.47 bits, the channel loses nearly half of the present input information.

See `m_h2.m` and `demo_channelprop.m` to run this example.

The maximal and minimal value of $I(X;Y)$ for certain X and Y form fundamental boundaries. The maximum is the entropy of X, which is the data transmission limit. Following our example intuitively this means the following: The minimum is reached when the channel exhibits no error. Then the amount of bits needed to transmit the picture is the entropy of the source distribution from which this picture is drawn. Alternatively, if the channel induces distortions such as in our example the mutual information shows the information that has been retained after transmission.

3.2.2 The Asymptotic Equipartition Property

The *weak law of large numbers* (WLLN) states that given a set of independent, identically distributed (i.i.d.) random variables X_i, the average value over a set of n members will be close to the expected value $E[X]$ for sufficiently large n. Formally:

$$A_\epsilon^{WLLN(N)}(X) = \left\{ x^N \in \mathcal{X} : \left| \frac{1}{N} \sum_{i=1}^{N} x_i - E[X] \right| < \epsilon \right\} \tag{3.15}$$

This notation can be generalized for some function ϕ of X:

$$A_\epsilon^{\phi(N)}(X) = \left\{ x^N \in \mathcal{X} : \left| \frac{1}{N} \sum_{i=1}^{N} \phi(x_i) - E[\phi(X)] \right| < \epsilon \right\} \tag{3.16}$$

$A_\epsilon^{\phi(N)}(X)$ is known as the *typical set*. As we are mostly interested in the (empirical) entropy of bit sequences, the function $\phi(X)$ becomes:

$$\phi(X) = -\log_2 p_X(x) \tag{3.17}$$

The definition of the *weakly typical set* $A_\epsilon(X)$ now becomes:

$$A_\epsilon(X) = \left\{ x^N \in \mathcal{X} : \left| -\frac{1}{N} \log_2 p_X(x^N) - H(X) \right| < \epsilon \right\} \tag{3.18}$$

So, the weak typical set contains those sequences whose empirical entropy approaches their (true) theoretical entropy within some error ϵ. This notation thus partitions the space of possible sequences in to two.

Similarly, the Asymptotic Equipartition Property (AEP) states that given a sequence of random variables $X_1, X_2, ..., X_N$ all i.i.d. from some distribution $p(x)$:

$$\lim_{N\to\infty} \frac{1}{N} \log \frac{1}{p(X_1, X_2, ..., X_N)} \to H(X) \tag{3.19}$$

This property, again, allows one to divide sequences into two sets. The sets of sequences whose empirically determined entropy is close to their (theoretically) true entropy, and all other sequences. Note that the number of *possible*

sequences can be vastly larger, than the number of sequences that were drawn from distribution $p(x)$ and that adhere to the AEP. We will explore this property further.

Given the AEP and some admissible error ϵ, one can refractor the definition of the AEP:

$$\begin{aligned}
\lim_{N\to\infty} \frac{1}{N} \log \frac{1}{p(X_1, X_2, ..., X_N)} &\to H(X) \\
-\epsilon \leq \left[-\frac{1}{n} \sum_{x_i} \log P_{\mathcal{X}}(x_i) - H(X) \right] &\leq \epsilon \\
-\epsilon + H(X) \leq \left[-\frac{1}{n} \sum_{x_i} \log P_{\mathcal{X}}(x_i) \right] &\leq \epsilon + H(X) \\
-N(\epsilon + H(X)) \leq \left[\sum_{x_i} \log P_{\mathcal{X}}(x_i) \right] &\leq -N(-\epsilon + H(X)) \qquad (3.20)
\end{aligned}$$

Which, keeping in mind that for i.i.d draws holds that $p(X^N) = \prod_{i=1}^{N} p_{\mathcal{X}}(x_i)$ and $\log ab = \log a + \log b$ refractors to:

$$\begin{aligned}
-N(\epsilon + H(X)) \leq \left[\log p(X_N) \right] &\leq -N(-\epsilon + H(X)) \\
2^{-N(\epsilon + H(X))} \leq p(X_N) &\leq 2^{-N(-\epsilon + H(X))} \qquad (3.21)
\end{aligned}$$

The set of sequences $(X_1, X_2, ..., X_N) \in \mathcal{X}^N$ i.i.d. from some $p(x)$ that adheres to derived Equation 3.21 is known as the *weakly typical set* $A_\epsilon^{(N)}$:

$$A_\epsilon^{(N)} = \{2^{-N(\epsilon + H(X))} \leq p(x^N) \leq 2^{-N(-\epsilon + H(X))}\} \qquad (3.22)$$

In general the typical set thus contains a small subset of possible sequences, but the sequences that are contained have an empirical entropy that is nearly equal to the true entropy. This has consequences for the number of bits that is required to describe all sequences which of course, is important for data compression.

For a sequence of length N made up of i.i.d. random variables $X_1, X_2, .., X_N$ drawn from pdf $p(x)$ all sequences $\mathcal{X}^N$ can be divided into the ones belonging to the typical set $A_\epsilon^{(N)}$ and the rest. The typical set contains a maximum of

$2^{-N(-\epsilon+H(X))}$ sequences. They can thus be described by an index number of at most $N(H+\epsilon)+2$ bits. Two bits are added because $N(H+\epsilon)$ isn't necessary an integer and a single bit to indicate the fact that the sequence belongs to the typical set. All other sequences can than be described with $N \log |\mathcal{X}| + 2$ bits.

3.2.3 The Noisy-Channel Coding Theorem

Given a memoryless channel, such as depicted in Figure 3.1, the channel capacity C is defined as:

$$C = \max_{p_X} I(X;Y) \tag{3.23}$$

Where p_X is the so called *optimal input distribution*. Memoryless means that the channel output $\boldsymbol{y}$ only depends on the current input $\boldsymbol{x}$. Section 3.1 and Figure 3.3 show an informal example of the channel capacity. The relation between the mutual information $I(X,Y)$ and the input distribution p_X for the binary channel can be seen in Figure 3.4. Following Shannon's theorem, the channel capacity measures the rate with which messages can be communicated over a channel with arbitrary small error.

An (N,K) code that is transmitted over a channel is a list with of $S = 2^K$ possible codewords each with a length of N bits. Formally the list takes on the following shape:

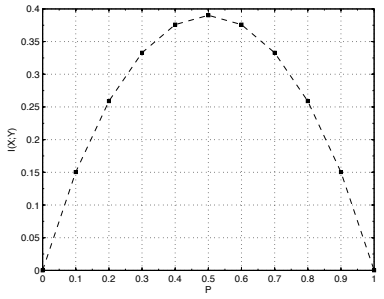
$$\{\boldsymbol{x}(1), \boldsymbol{x}(1), ..., \boldsymbol{x}(2^K)\} \in \mathcal{A}_{\mathcal{X}}^N \tag{3.24}$$

where each codeword $\boldsymbol{x}$ has length N. The codewords are of course but a subset of all possible sequences. The set that is comprised of the codewords is denoted with $\mathcal{A}_{\mathcal{X}}^N$ and is known as the *typical set* (Section 3.2.2). It plays an important role in coding and channel theory.

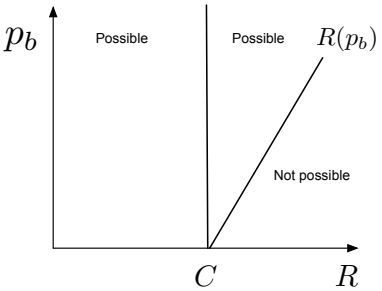
For messages that are encoded with an (N,K) code and transmitted over some noisy channel, the *rate* for an alphabet with a cardinality of 2, $|\mathcal{X}| = 2$ is defined as follows:

$$R = \frac{K}{N} \tag{3.25}$$

This finally brings us the Shannon's noisy-channel coding theorem:



(a) Input alphabet probability p_1 plotted the against mutual information $I(X;Y)$ for a binary channel with $p_b = 0.15$. It can be generated with `demo_max_mi.m`.



(b) Schematic overview of Shannon's noisy-channel theorem. Figure adapted from [33].

Figure 3.4 –

Theorem 1. For every discrete memoryless channel with channel capacity C is holds that:

- For any $\epsilon > 0$ and $R < C$, there exists a code with length N and a *rate* R and a decoding algorithm such that the average probability of a block error is smaller than ϵ.

- Given an acceptable probability of error p_e, a rate up to $R(p_e)$ is achievable:

$$R(p_e) = \frac{C}{1 - H_2(p_e)} \tag{3.26}$$

- Rates greater than $R(p_e)$ are not achievable for any p_e.

See Figure 3.4b for a schematic overview.

3.2.4 Jointly Typical Sequences

For a sequence of length N made up of i.i.d random variables $X_1, X_2, .., X_N$ from pdf $p(x)$ and a sequence of the same length N from $Y_1, Y_2, .., Y_N$ from pdf $p(y)$, *jointly typical* is defined as follows. Realization $\boldsymbol{x}$ is *weakly typical* of $p(x)$ when:

$$\left| \frac{1}{N} \log \frac{1}{p(\boldsymbol{x})} - H(X) \right| < \epsilon \tag{3.27}$$

In the same fashion, $\boldsymbol{x}$, $\boldsymbol{y}$ are jointly typical when:

$$\begin{aligned} \left| \frac{1}{N} \sum_{i}^{N} \log \frac{1}{p(\boldsymbol{x}_i)} - H(X) \right| < \epsilon \\ \wedge \\ \left| \frac{1}{N} \sum_{i}^{N} \log \frac{1}{p(\boldsymbol{y}_i)} - H(Y) \right| < \epsilon \\ \wedge \\ \left| \frac{1}{N} \sum_{i}^{N} \log \frac{1}{p(\boldsymbol{x}_i, \boldsymbol{y}_i)} - H(X,Y) \right| < \epsilon \end{aligned} \tag{3.28}$$

This means that the set $A_\epsilon^{(N)}$ of jointly typical sequences $(\boldsymbol{x}^N, \boldsymbol{y}^N)$ is the set of N sequences whose empirical entropies equal their true entropies with in some error bound ϵ:

$$A_\epsilon^{(N)} = \left| \frac{1}{N} \log \frac{1}{p(\boldsymbol{x}^N, \boldsymbol{y}^N)} - H(X,Y) \right| < \epsilon \tag{3.29}$$

The number of items, or weak typical sequences, in the weak typical set $A_\epsilon^{(n)}$ is then:

$$|A_\epsilon^{(N)}| \leq 2^{N(H(X,Y)+\epsilon)} \tag{3.30}$$

This has fundamental consequences for the number of sequences that can be generated. If one looks at sequences from $\boldsymbol{x} \sim X^N$ and $\boldsymbol{y} \sim Y^N$, and fixes the output sequence $\boldsymbol{y}$ there will be $2^{NH(X|Y)}$ conditionally typical input sequences.

The probability that 'some other' input sequence $\boldsymbol{x}$ is jointly typical with output sequence $\boldsymbol{y}$:

$$\frac{2^{NH(X|Y)}}{2^{NH(X)}} \Rightarrow \tag{3.31}$$

$$2^{N(H(X|Y)-H(X)} \Rightarrow \tag{3.32}$$

$$2^{-N(-H(X|Y)+H(X)} \Rightarrow \tag{3.33}$$

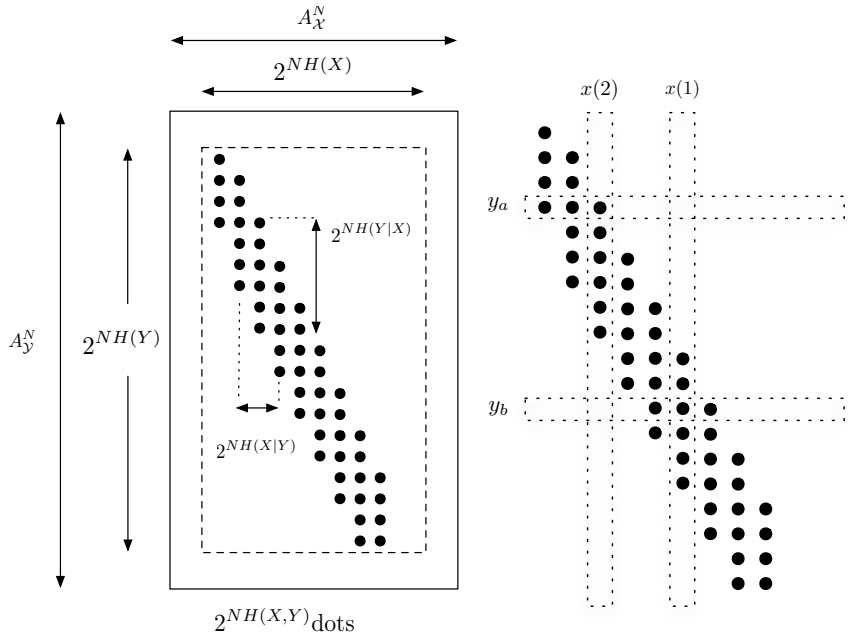
$$2^{-N(I(X;Y))} \tag{3.34}$$

Which means that one can choose about $2^{N(I(X;Y))}$ codewords $\boldsymbol{x}$ from X before hitting one that is equally likely to have caused $\boldsymbol{y}$. There are thus about $2^{N(I(X;Y))}$ distinguishable signals $\boldsymbol{x} \sim X$. A schematic overview of the jointly typical set for $p(\boldsymbol{x}, \boldsymbol{y})$ can be seen in Figure 3.5b.

3.2.5 Random and Typical Set Decoding

A typical set decoder will decode channel output $\boldsymbol{y}$ as $\hat{m}$ if, and only if, $\boldsymbol{y}$ and $\boldsymbol{x}(\hat{m})$ are jointly typical. An overview can be seen in Figure 3.6. The algorithm is then as follows:

- Generate a random code book from a distribution $p(X)$ with $M = 2^{NR}$ codewords of code (N, K).
- This code book is shared by both transmitter and receiver.
- For a message m, one chooses a corresponding codeword $\boldsymbol{x}(m)$ with index m, which is transmitted.
- The receiver gets the channel output $\boldsymbol{y}$ and now looks through its code book to find the sequence $\boldsymbol{x}(\hat{m})$ that is jointly typical with the received $\boldsymbol{y}$. Secondly, there may not exist any other sequence with which $\boldsymbol{y}$ is also jointly typical. If both requirements hold, the decoder outputs index $\hat{m}$ of the jointly typical codeword.



(a) The jointly typical set.

(b) The jointly typical decoding. Channel output y_a correctly decodes to $x(2)$.

Figure 3.5 – The jointly typical set and jointly typical decoding. Figure adapted from [33].

3.3 Metadata

This Section explores the concepts and limits of the usages of so called *side information* in Shannon's communication framework. This approach is conceptually based on Slepian-Wolf encoding. Side information can be seen as a form of metadata. As shown in Chapter 3, Section 3.2.5, the maximum amount of distinguishable samples for a framework that uses typical set, or random set decoding, is about $2^{I(X;Y)}$ [14, 33]. This is also the maximum number of samples any database can hold while retaining a near zero probability of error when

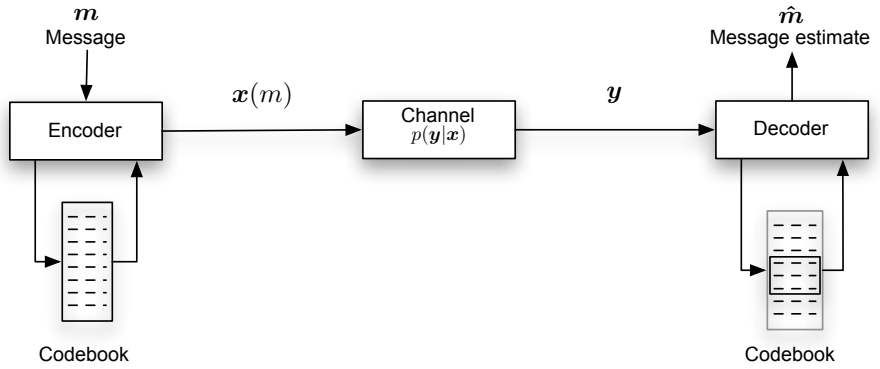


Figure 3.6 – Random codes

identifying query samples. e.g. the probability of a *collision*.

The maximum number of distinguishable samples can only be increased by taking one of the following steps:

- Increase the number of samples regardlessly and accept the resulting identifications errors or collisions. Obviously, forensic applications should exhibit a near zero probability of error, rendering such an approach infeasible.

- Process samples in such a way that their quality becomes good enough to be modeled as discrete memory less source so that they can be enrolled in the current list-decoding framework. This approach is the focus of Chapter 2.

- Aid the system by adding reliable meta-data to the query that is not distorted by channel noise.

This section will show the fundamental concepts and limitations of using metadata in (noisy) databases. Further more, we will show how meta data can be used to increase database capacity and database identification speed.

3.3.1 Channel Identification Limitations

Given the Discrete Memory less Channel (DMC) channel model (Figure 3.7) and a source that produces independantly identically distributed (i.i.d.) random variables, the notation of *identification capacity* C_{id} is defined as follows:

$$C_{id} = \frac{1}{N} I(\boldsymbol{X}; \boldsymbol{Y}) \tag{3.35}$$

Where $I(X;Y)$ is the mutual information between channel input and output. One should note the important difference between the identification capacity C_{id} and the *channel capacity* C. The latter is defined as:

$$C = \max_{p_X} I(\boldsymbol{X}; \boldsymbol{Y}) \tag{3.36}$$

The channel capacity can thus be maximized by tuning the input distribution p_X. This is of course a luxury one doesn't have in the identification and retrieval framework. One has to make do with the fixed dataset it is offered, be it pictures, fingerprints or something else. Furthermore, the true distribution of the dataset may not be known.

The expected amount of sequences M that may occur according to the Asymptotic Equipartition Property (Section 3.2.2) is:

$$M = 2^{H(\boldsymbol{X})} \tag{3.37}$$

Per definition:

$$I(\boldsymbol{X}; \boldsymbol{Y}) \leq H(\boldsymbol{X}) \tag{3.38}$$

Where equality only occurs when the channel is noiseless. In terms of the identification capability C_{id} this means that the latter is upper bounded by $H(X)$:

$$C_{id} = \frac{1}{N} I(\boldsymbol{X}; \boldsymbol{Y}) \tag{3.39}$$

Refractoring for I:

$$C_{id} = \frac{1}{N} (H(\boldsymbol{X}) - H(\boldsymbol{X}|\boldsymbol{Y}) \tag{3.40}$$

$$C_{id} = \frac{1}{N} (H(\boldsymbol{X}) - H(\boldsymbol{X}|\boldsymbol{Y}) \leq \frac{1}{N} H(\boldsymbol{X}) \tag{3.41}$$

Thus in the case of the DMC, i.i.d. samples and $2^{NH(X)}$ sequences, there will be $2^{NH(X|Y)}$ sequences that can not be distinguished. This means that the maximum amount of samples that can be distinguished is $2^{I(X;Y)}$ which in turn equals: $2^{NC_{id}}$. The total amount of information (in bits) that need to be added for successful decoding of sequences of length N for a noisy channel with identification capacity C_{id} therefore is:

$$\text{Bits of meta-data} = \log\left(\frac{2^N}{2^{NC_{id}}}\right) \tag{3.42}$$

3.3.2 Concepts and Limitations of Meta-data

This Section will explore the usage of the meta-data within in the Discrete Memory less Channel (DMC) framework. Schematically, the approach can be seen in Figure 3.7. Meta-data, or side-information does two things:

- Enlarges the amount of samples that can be identified in the database without collisions.
- Meta-data queries are very fast.

The basic idea is to separate the codebook into bins. These bins do not have to be similar in shape, but the maximum length of a bin should always be such that there are no collisions within the bin when decoding the noisy query. The bins are formed dynamically, that is, they are formed pending the specific type and amount of meta-data that is added to the query. The bin size is upper bounded by the channel identification capacity C_{id}. Consequently, this also provided us with the minimum amount of bits (Equation 3.42) of meta-data that need to be added to the query to prevent collisions.

Apart from increasing the number of distinguishable samples, meta-data queries are fast. Normal decoding more or less requires that one compares the query samples against all samples in the database. This is an *NP-hard* problem. And even in cases where hash tables are utilized, the entire table will not fit into memory in a lot of applications, such as fingerprint databases. Meta-data searches exclude large parts of the database leaving less to be compared based on decoding.

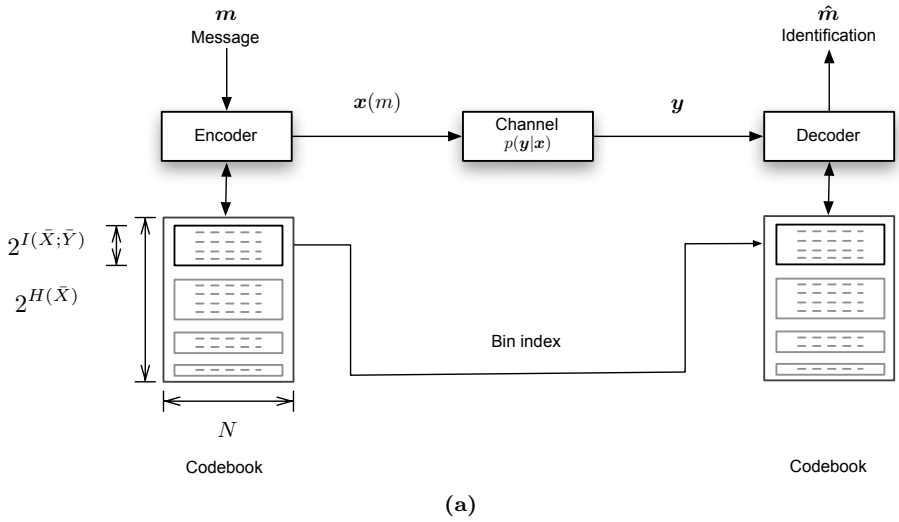


Figure 3.7 – Side information and Shannon's communication model.

All micro-structures are random. But some are clearly more random than others.
– Me (1978 - present)

Chapter 4

Empirical Identification Limits

This chapter will use all mathematical tool and models from Chapter 3 to empirically answer the second research question:

- *What is the maximum number of unique distinguishable samples the database will hold.*

This chapter will empirically determine the upper bound for the datasets that are used in combination with the specific synchronisation algorithms deployed. Not all micro-structures are as rich in information, and in some datasets the micro-structures are more alike between samples.

Bad synchronisation also kills the upper bound, as this chapter will prove. The first research question:

- *How can the geometrical distortions and imaging artifacts be restored between an acquired image and a sparse template.*

is therefore also answered by default.

4.1 Validation method

Validation of the synchronisation methods and the microstructure potential as an identifiable token consists of four stages:

- Channel distortion caused by the acquiring image device
- The entropy of the micro structures
- The mutual information between observations from equal samples that have been synchronized
- The intra and inter class distance distributions

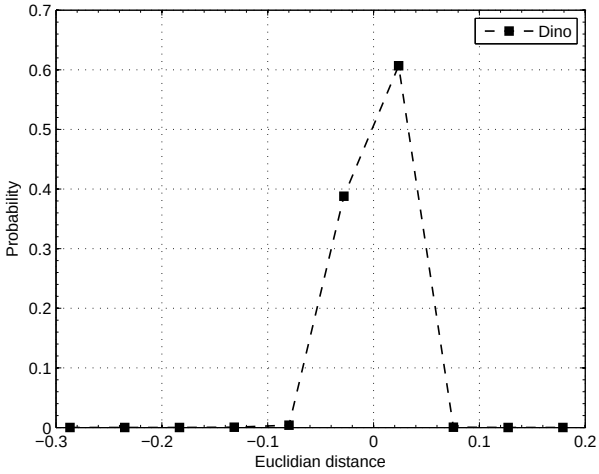
4.1.1 Scenarios

Using the above validation method, the following scenario's are tested:

- Enrollment and Identification device and lighting are identical. All devices and lighting setups are tested for which datasets are available.
- Mismatch in enrollment and identification device. These test the most portable device, Camera A, against the two industrial camera's. These are Camera B and Camera D.
- Mismatch in lighting. This test if the differences between a diffused ring light and a more direct lighting setup can be bridged.

4.1.2 Device channel distortion

Even if one discards all user induced distortions while acquiring an image with a hand-held image device, there will still be distortions that are caused by the device itself. This is a systematic error that gives insight into the limitations of the framework.



(a) Channel distortions caused by the Camera A microscope.

Figure 4.1 – Channel distortions.

4.1.3 Entropy

The empirical bit entropy is calculated following Chapter 3, Section 3.2.1, Equation 3.3. Generally speaking, one would like a candidate micro-structure to have pixels whose values are uniformly distributed. If all possible values are equally present in a structure it conveys the most information. As proved in Chapter 3, the bit entropy is one of the mathematical tools to determine the upper bound on the number of identifiable micro-structure samples in a database.

For n synchronized micro-structures that originate from an equal sample m, the algorithm executes the following two steps:

- Ascertain the empirical probability density function $p(x)$ with the Parzen density estimation algorithm. For details see Appendix B.1.
- Entropy is determined via $H(X) = -\sum_{x \in X} p(x) \log_2 p(x)$

4.1.4 Mutual Information

Mutual information provides a solid stochastic model to determine what the 'overlap' in information is between two samples. It is the measure of the amount of information that one stochastic variable contains about another stochastic variable. The samples, or the micro-structures, are realizations from these stochastic variables.

Mutual information is covered in depth in Chapter 3, Section 3.2.1, Equations 3.8 to 3.9. The empirical mutual information from observations originating from equal and unequal samples is determined as follows:

- Determine the joint probability density function $p(\boldsymbol{x}, \boldsymbol{y})$ using the Parzen density estimation algorithm. For details see Appendix B.1. Determine $H(X, Y)$ via Equation 3.4.
- Estimate the marginal probability density functions $p(\boldsymbol{x})$ and $p(\boldsymbol{y})$. Determine the empirical bit entropy $H(X)$ and $H(Y)$ following Section 4.1.3 and Equation 3.3.
- Calculate the empirical mutual information via Equation 3.8, i.e.

$$I(X;Y) = H(X) + H(Y) - H(X,Y)$$

The ultimate goal of this framework is to ensure that the empirical mutual information of the samples is close to their true mutual information, after synchronisation. This difference is the ultimate benchmark for the software quality. Secondly, one hopes that the micro-structures behave as *i.i.d* samples and that they have a large *inter* class distance.

Ideally one achieves to following result. If the samples are true *i.i.d* realizations and observations from different samples are independent, than, for X and Y independent, $H(X|Y)$ will equal $H(X)$.

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) \\ I(X;Y) &= H(X) - H(X) = 0 \end{aligned} \tag{4.1}$$

(4.2)

The empirical probability density graph for the mutual information between all observations originating from different samples should thus ideally be a Dirac spike with a zero average.

For observations originating from an identical sample, the ideal situation is as follows:

$$I(X;X) \quad = \quad H(X) \tag{4.3}$$

In our case we are dealing with X and some X' which is observation from an identical sample. Ideally the synchronisation framework ensures that these two are identical. The better the framework works, the closer $I(X;X')$ comes to the entropy of X, $H(X)$.

The empirical probability density graph for the mutual information between all observations originating from identical samples should thus ideally be a Dirac spike with an average equal to the empirical entropy of the source.

Empirical Determination

The mutual information is determined empirically using the Parzen Estimator. See Appendix B.1. The empirical estimation is tricky business. Results can vary significantly pending the parameters of the Parzen estimator.

4.1.5 Intra en inter class distance distribution

Queries are resolved against the database using a certain metric. Identification of a noisy query $\boldsymbol{y}$ against a database holding M samples can be modeled as maximum likelihood decoding. The estimated identification index $\hat{m}$ is ascertained as follows:

$$\hat{m} = \underset{1 \leq m \leq M}{\arg\max}\, p(\boldsymbol{x}(m)|\boldsymbol{y}) \tag{4.4}$$

Practically speaking, this means that entries in the database should occupy non overlapping regions in the comparison metric-space, as is depicted in Figure 1.3. For successful identification there may be no indices m that give a greater

likelihood than the true index. In the classic two case bayesian classification with classes m_1 and m_i this can be expressed by:

$$p(\boldsymbol{x}|\boldsymbol{y}(m_1)) > p(\boldsymbol{x}(m_i)|\boldsymbol{y}) \tag{4.5}$$

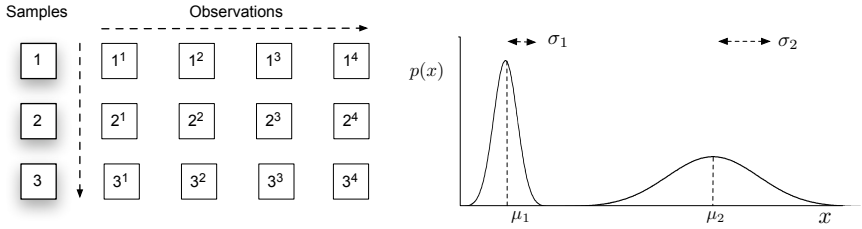
Where m_i are all indices for all database entries that do not match with the query. The resulting probability densities for the two hypothesis $p(\boldsymbol{x}(m_1)|\boldsymbol{y})$ and $p(\boldsymbol{x}(m_i)|\boldsymbol{y})$ may thus not overlap. The same goes for any combination with $\boldsymbol{y}$. Should these probabilities overlap, the system designer will be left with a choice between *false accepts* or a certain probability of error p_b. The framework than transforms into a *retrieval* system, rather than an *identification* system.

To validate the used identification metric the distance between true matches and the noisy query, or inter class distance, may not exceed the distance between the query and any other database entries. This is validated as follows. Given a set of m unique samples and a n observations per unique sample, the validation algorithm does the following:

- Determine all differences between observations originating from equal samples. For n observations for m samples this gives $(m \cdot \frac{n}{2})$ combinations.
- Determine all differences for all observations originated from unequal samples. This gives $n \cdot (\frac{m}{2})$ combinations.
- The probability density function for both sets is empirically determined.

See Figure 4.2a for an overview. Ideally this results in the situation as depicted in Figure 4.2b. Differences from observations from an identical sample are all small and have a small variance. The more narrow the left peak, the better the synchronisation algorithm can transform a presented observation into the form in which the original observation is stored in the database.

It is also important to note that for true gaussian *i.i.d* samples the euclidian distance serves as the optimal minimum distance decoder in this framework. Also, for true *i.i.d* gaussian samples there is no difference between calculating the cross correlation of the euclidian distance as proved in Appendix B, Section B.2.



(a) Overview of testing procedure.

(b)

Figure 4.2 – Schematic of the validation framework. Figure 4.2b shows optimal results. The graphs represent the differences between observations originating from the same sample and the all differences from observations originating from different samples. The more the *intra* class distance pdf (left graph here) resembles a spike, the better the synchronisation is.

4.2 Results

4.2.1 Identical enrollment and identification device

Dataset 4

Dataset 4 was acquired with both Camera D and Camera A. The mutual information before and after synchronisation shows a clear improvement for both camera's. Camera A (Figure 4.3d) does show a superior result in comparison with Camera D (Figure 4.3b).

Camera A Dataset 4 set achieves good separation of the inter and intra class distances. This proves that error-less identification is possible with this dataset and camera. This setup is currently in use as a commercial demonstration.

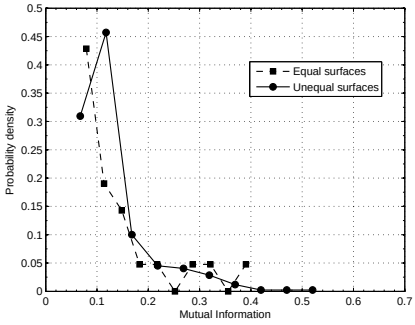
The intra and inter class distances for Dataset 4 exhibit a very long tail is the most notable feature of the intra class distance probability density function. In general, a heavy tailed intra class distance pdf is evidence for two things: blurring or bad synchronisation. Blurring a micro-structure destroys high detail as it is effectively a low pass filter. The remaining low frequency components

	Dataset 11	Dataset 4	Dataset 4	Dataset 3	Dataset 1
Observation	Entropy	Entropy	Entropy	Entropy	Entropy
a	6.2	4.80	3.86	**6.12**	**7.41**
b	6.2	**4.90**	**4.11**	5.87	7.38
c	6.2	4.71	3.90	5.74	7.33
d	6.2	4.82	3.95	5.82	7.33
e	6.2	4.85	4.10	6.08	7.34
f	6.0	4.74	3.87	5.95	7.38
g	**6.3**	4.84	3.92	5.76	7.36
h	**6.3**			6.05	7.38
i	6.1			5.90	7.39
j	6.1			5.78	7.34

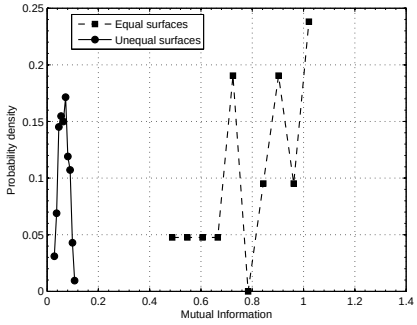
Table 4.1 – Empirical bit entropy. Note that for 8 bit gray scale images, the maximal attainable value is 8 bit.

will always have a higher cross correlation, or mutual information, with all other micro-structures from any class. Visual inspection shows that synchronisation is not flawless. The algorithm does suffer a bit from the fact that the Dataset 4 were printed with a regular office printer which gives shape irregularities. Secondly, Camera D was used with out any special lighting and images were therefore acquired with ambient office light. This will most certainly have also contributed to some blurring.

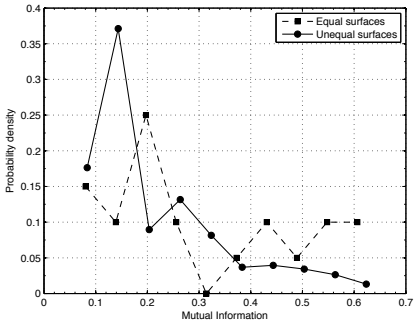
About 5 percent of the dataset resides in the tail. This dataset has 48 samples with 3 observations each. The number of combinations for the inter class distances is therefore $48 \cdot \binom{3}{2}$ which is 144. That means that there are 7 combinations between equal labels that have very low cross correlations. These 7 combinations might not seem much, but it does mean that error less identification with this particular setup is not possible at this moment. Further more, the long tail and small gap between the intra and inter class distances suggest synchronisation and lighting problems.



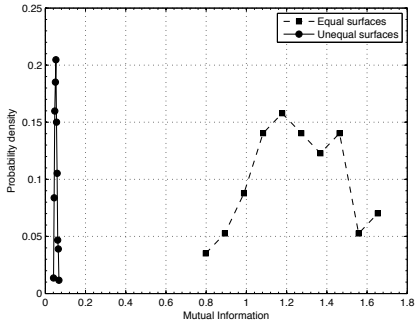
(a) Dataset 4 before synchronisation, acquired with Camera D.



(b) After synchronisation.



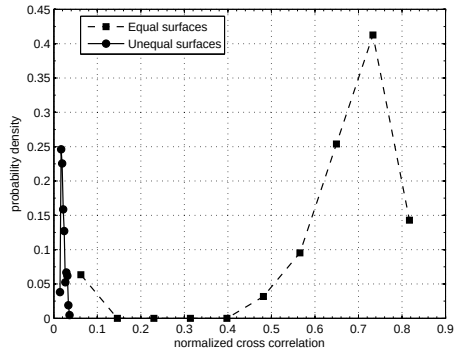
(c) Dataset 4 before synchronisation acquired with Camera A.



(d) After synchronisation.

Figure 4.3 – The probability density functions of the empirical mutual information for observations originating from equal and unequal samples before, and after synchronisation for **Dataset 4**.

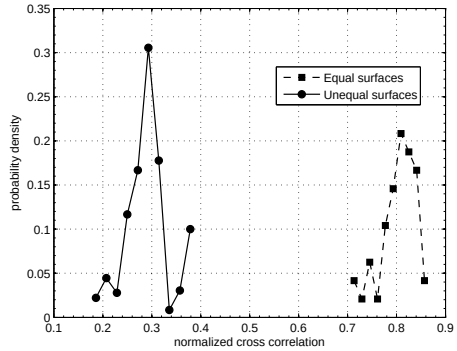
Dataset 4	
Device	Imaging-source
Quantity	48*3
Micro structure	58800 bytes
Metric	Cross Correlation
Post	global mean subtraction high pass filtering



(a)

Table 4.2 – Inter and intra class distance for **Dataset 4** acquired with Camera D.

Dataset 4	
Device	Camera A
Quantity	38*3
MS	133331 bytes
Metric	Cross Correlation
Post	global mean subtraction



(a)

Table 4.3 – Inter and intra class distance for **Dataset 4** acquired with Camera A.

Dataset 11

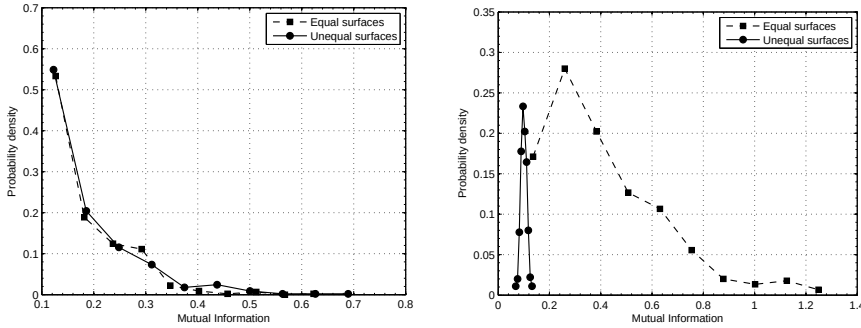
The aluminium dataset was acquired with an industrial camera. The inter and intra class distances were determined using the Euclidean distance as comparison metric. The results are excellent and can be seen in Figure 4.5a. The framework uses both Euclidean distance and the maximum 2D cross correlation value as a metric. The latter is naturally much more resilient against small geometric distortions. In the Dataset 11 case, the leftmost intra class distance pdf is almost a spike. This is statistical evidence for near perfect synchronisation. There is also a huge gap between the intra and inter class distance pdf. This supports the claim that the aluminium micro-structures differ significantly between samples. Flawless identification based on this dataset is therefore possible. This particular setup and framework have been implemented in a commercial demonstration.

The intra class mutual information pdf (Figure 4.4b) provides additional evidence. It is a spike close to zero, indicating that observations from different samples are statistically independent. This can not be said about the inter class pdf. It is nowhere near the source entropy, which is 6 bits (Table 4.1). As outlined in Section 4.1.4, determining the empirical mutual information is a error prone process. Given the fact that this dataset and synchronisation framework work excellent in a real life demonstration, an ill determined empirical mutual information is the most likely cause for this anomaly.

Dataset 3

Dataset 3 was by far the most problematic and most researched dataset. Its labels, including the template, were ill manufactured which resulted in a large variety of shapes and sizes of the template symbol that was used for synchronisation. There is also evidence that R micro-structures to not behave as $i.i.d$ samples, but are in fact, locally correlated.

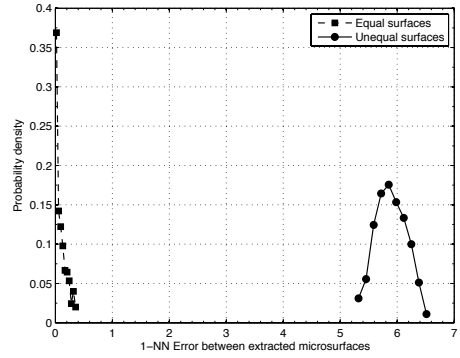
Figure 4.7a shows the intra and inter class distance pfd's using normalized cross correlation as comparison metric. They show that about 0.04 procent of the inter class distances overlap with 0.07 procent of the intra class distances.



(a) Before synchronisation. **(b)** After synchronisation.

Figure 4.4 – The probability density functions of the empirical mutual information for observations originating from equal and unequal samples. Figure 4.4a shows the situation for the **Dataset 11** dataset before synchronisation, Figure 4.4b shows the situation after.

Dataset 11	
Device	Industrial Camera
Quantity	100
MS	1188 bytes
Metric	Euclidian dist
Post	global mean subtraction



(a)

Figure 4.5 – Inter and intra class distance for **Dataset 11**.

Out of the $\binom{181}{2} = 147153$ possible inter class distance combinations, this means that there are about 588 label combinations between a query label and the enrolled database that give unusual high cross correlations. The reverse applies for all the possible intra class distance combinations. There are $182 * \binom{3}{2} = 1092$ possible intra class combinations of which approximately 77 equal label combinations give unexpectedly low cross correlation values.

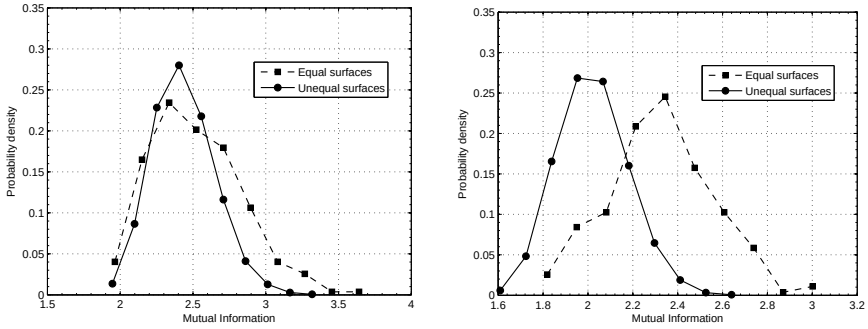
Due to the fact that the template shape in this dataset exhibits quite some variation in shape and size one can see visually that synchronisation is far from perfect. This certainly contributes to the high variance of the intra class distance pdf. As detailed in Section 2.6.2 the cross correlation by its very nature is influenced by large-valued components. To investigate if the R dataset labels all shared a signal component that is large in magnitude, three experiments where done:

- Usage of term-weighted cross correlation, as outlined in Section 2.6.2.
- Usage of high pass filtering
- Feature extraction of local variances of each micro-structure

Figure 4.8a shows the results using term-weighted cross correlation. It shows a significant improvement over the result that was attained using normal cross-correlation. This is weak evidence that these labels are indeed not $i.i.d$ and share a significant common signal component. For an identification system the results are still not good enough, as the tails still overlap.

Figure 4.9a shows the results of high pass filtering in combination with the term-weighted cross correlation. Strictly speaking, this setup was already incorporating a form of high pass filtering by removing the global mean from the micro-structres prior to comparison. The results are again an improvement, but still exhibit long tails. As predicted, this identification setup showed a significant amount of false accepts/rejects and false identifications.

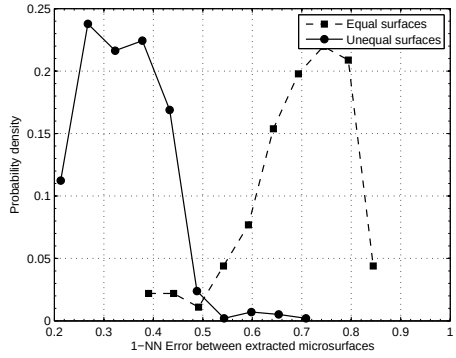
The last experiments focussed on using local variances of the micro-structures as a robust feature, possibly in combination with the graph based approach proposed in Section 2.8. The local variances as sole feature proved not to be viable for identification purposes. The graph based approach is still ongoing.



(a) Before synchronisation. **(b)** After synchronisation.

Figure 4.6 – The probability density functions of the empirical mutual information for observations originating from equal and unequal samples. Figure 4.6a shows the situation for the **Dataset 3** dataset before synchronisation, Figure 4.6b shows the situation after.

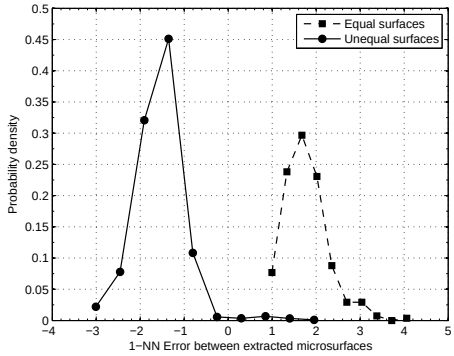
Dataset 3	
Device	Camera A
Quantity	546
MS	1024 bytes
Metric	Cross Correlation
Post	global mean subtraction



(a)

Figure 4.7 – Inter and intra class distance for **Dataset 3**.

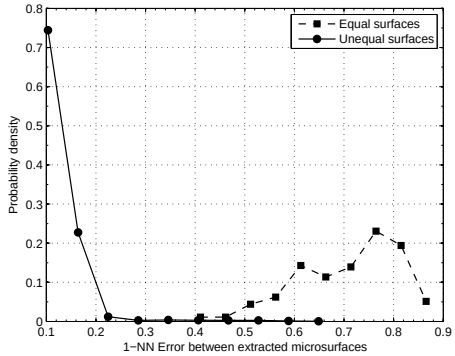
Dataset 3	
Device	Camera A
Quantity	546
MS	1024 bytes
Metric	Term weighted cc
Post	global mean subtraction



(a)

Figure 4.8 – Inter and intra class distance for **Dataset 3**.

Dataset 3	
Device	Camera A
Quantity	546
MS	1024 bytes
Metric	Term weighted cc
Post	global mean subtraction
	High pass filtering



(a)

Figure 4.9 – Inter and intra class distance for **Dataset 3**.

Dataset 1

Dataset 1 is limited in size, comprising of only 12 samples with 3 observations each. The preliminary results, however, are very promising. Undoubtedly helped by the good printing quality of the template, the synchronisation is good. A point that is further shown by the small variance of the intra class distance pdf in Figure 4.11a. Separation between the intra and inter class distance pdf's is also among the best.

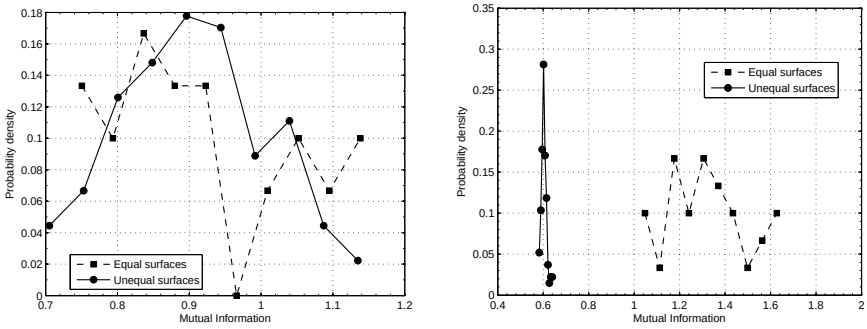
This dataset and framework were used in a commercial demonstration. Preliminary results are very hopeful and a trial with a very large dataset should definitely be done.

Dataset 14

Dataset 14 was acquired five times. The first trial was with a limited set of 45 samples and Camera A microscope. The second dataset was comprised of 288 samples. The big dataset was acquired with three very different imaging devices: Camera A, Camera C, the industrial Camera B with a diffused ring light and with Camera B with the small Type A ring light. Visually, these acquisitions are very different. Details of the micro-structures for these acquisitions can be seen in Figure 4.19.

Preliminary tests with a limited set and Camera A showed promising results, as can be seen in Figure 4.14a. This lead to the acquisition of a much larger dataset. The results for the large set and Camera A can be seen in Figure 4.15a. The mean of the inter class distance pdf has become smaller, which is a positive development, however, the tail has grown a bit. A possibility for this is the fact that Camera A blurs its acquisitions slightly. In both cases the intra class distance pdf has a small variance and is spike like. This is evidence for good synchronisation.

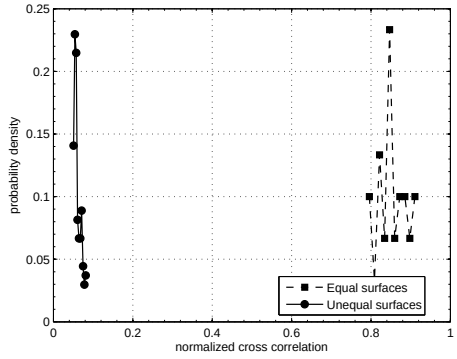
A small test was undertaken with the limited set and the handheld Camera C device. Camera C detects UV-light. The results can be seen in Figure 4.13a. Camera C lacks significant detail to bring out micro-structure detail. The initial results do not look promising as both inter and intra class distance pdf's overlap



(a) Before synchronisation. **(b)** After synchronisation.

Figure 4.10 – The probability density functions of the empirical mutual information for observations originating from equal and unequal samples. Figure 4.10a shows the situation for **Dataset 1** before synchronisation, Figure 4.10b shows the situation after.

Dataset 1	
Device	Camera A
Quantity	12 * 3
MS	46631 bytes
Metric	Cross Correlation
Post	global mean subtraction



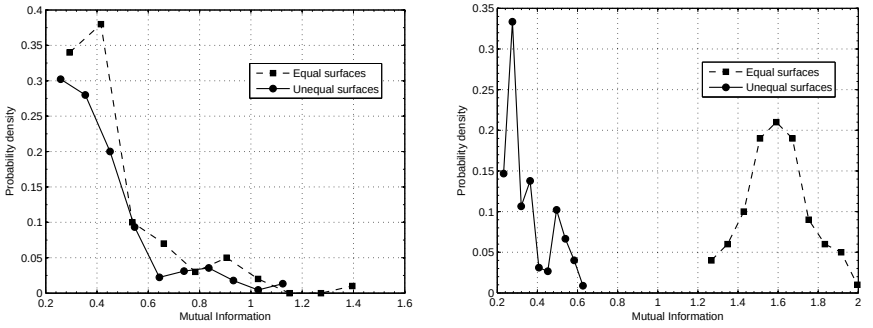
(a)

Figure 4.11 – Inter and intra class distance for **Dataset 1**.

and have big variances. Trials with Camera C were therefore stopped.

Results for the setup with Camera B and a big diffused ring light can be seen in Figure 4.16a. It is notable that the intra class distance pdf is nearly identical to the one attained with the Type A LED light (Figure 4.17a). This indicates good synchronisation by the algorithm. Most notable, however is the long tailed, big variance, inter class distance pdf. The big diffused ring light blurs the acquisition result. Blurring is a form of low pass filtering and thus increases the mutual information between observations of unequal labels. Identification with a small probability of error is therefore no longer possible.

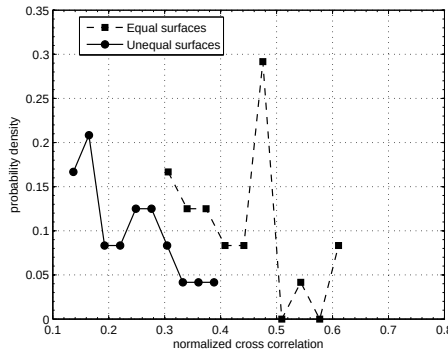
The setup with Camera B and the Type A LED provided the best results (Figure 4.17a). The separation of the inter and intra class distances obtained in this setup approaches the attainable limit. However, in the case of industrial deployment, the Camera B lens' very small depth of field will have implications for the tolerable vibrations the conveyor belt with product samples may induce. In the current setup, where samples are placed manually under Camera B, error less identification is possible with dataset. The framework and this dataset are currently being enrolled into a commercial demo.



(a) Before synchronisation. **(b)** After synchronisation.

Figure 4.12 – The probability density functions of the empirical mutual information for observations originating from equal and unequal samples. Figure 4.12a shows the situation for **Dataset 14** before synchronisation, Figure 4.12b shows the situation after.

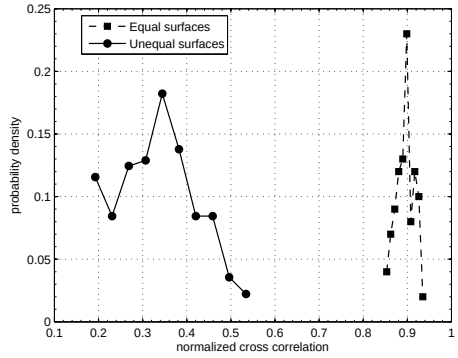
Dataset 14	
Device	Camera C
Quantity	45
MS	200 bytes
Metric	cross correlation
Post	local mean subtraction



(a)

Figure 4.13 – Inter and intra class distance for the big **Dataset 14** acquired with the handheld Camera C with UV light.

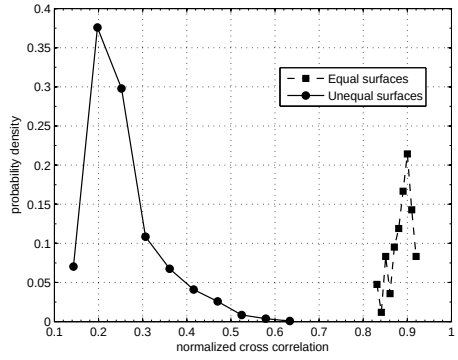
Dataset 14	
Device	Camera A
Quantity	45
MS	4000 bytes
Metric	cross correlation
Post	global mean subtraction



(a)

Figure 4.14 – Inter and intra class distance for the small **Dataset 14**.

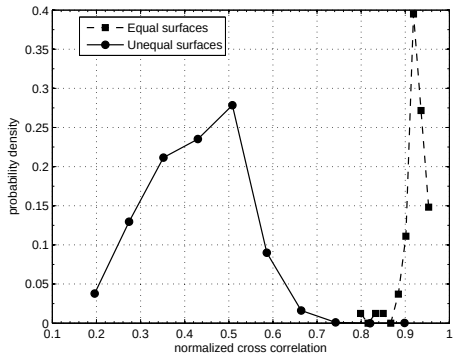
Dataset 14	
Device	Camera A
Quantity	288
MS	4000 bytes
Metric	cross correlation
Post	local mean subtraction



(a)

Figure 4.15 – Inter and intra class distance for the big **Dataset 14** with significant distortions. Samples were rotated over 30 degrees and shifted to the edge of the FOV of the Camera A microscope.

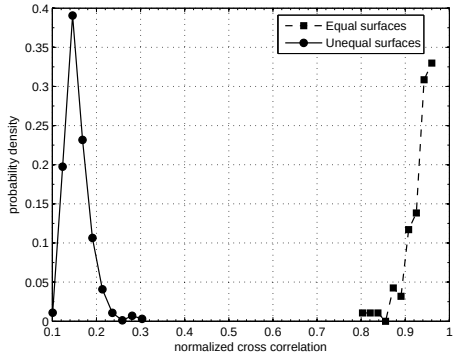
Dataset 14	
Device	Camera B Big 12 cm diffused ring light
Quantity	288
MS	4000 bytes
Metric	cross correlation
Post	local mean subtraction



(a)

Figure 4.16 – Inter and intra class distance for the big **Dataset 14** with significant distortions. Samples were rotated over 30 degrees and shifted to the edge of the FOV of the Camera B microscope. These samples where acquired with a big diffused ring light. The latter causes blurring which hurts performance very significantly.

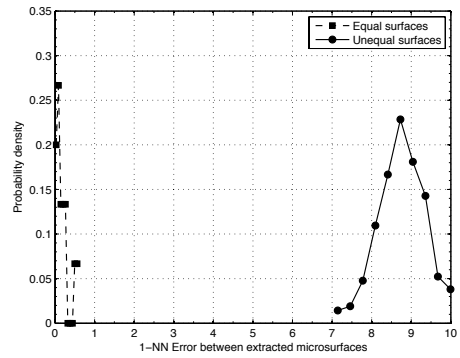
Dataset 14	
Device	Camera B Type A LED ring light
Quantity	288
MS	4000 bytes
Metric	cross correlation
Post	local mean subtraction



(a)

Figure 4.17 – Inter and intra class distance for the big **Dataset 14** with significant distortions. Samples were rotated over 30 degrees and shifted to the edge of the FOV of the Camera B microscope. These samples where acquired with a small LED ring light.

Tree Shape	
Device	Camera A
Quantity	10
MS	400 bytes
Metric	Euclidian distance
Post	global mean subtraction



(a)

Table 4.4 – Inter and intra class distance for the **Tree** dataset.

The Tree shape

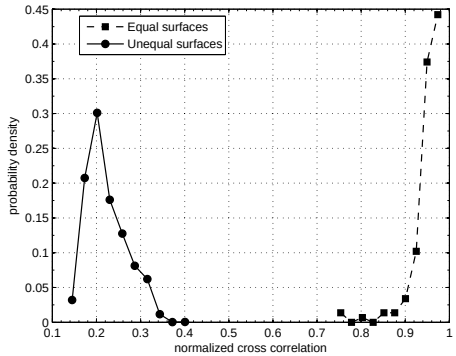
The Tree dataset shows excellent synchronisation results. A point that is further re-enforced by the fact that the comparison metric shown in Figure 4.4a is the Euclidian distance.

There are two primary reasons for caution. First of all, a dataset of 10 samples does not proof anything. Secondly, this dataset was pre-processed using morphological operations which tend to be highly specific for the imaging device and lighting used.

The Dataset 15

Due to the small database size of 14 samples and 3 observations each, the excellent results for Dataset 15 only give a very preliminary indication. Failed synchronisation, the acquisition lighting and image blurring are the most prominent controllable factors that lead to bad results. The results from Figure 4.6a further emphasize the influence of ill focussing. The second set of results was namely obtained by removing 3 ill focussed observations from the database. With this small set, it leads to a significant performance increase. Automatic detection of ill focussed images is the subject of ongoing work, as put forward

Dataset 16	
Device	Camera B, Type A LED ring
Quantity	50 * 3
MS	5250 bytes
Metric	Cross correlation
Pre	Morphological processing
Post	global mean subtraction



(a)

Figure 4.18 – Inter and intra class distance for **Dataset 16**.

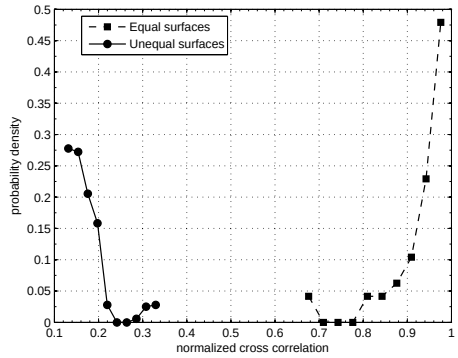
in Section 2.8.

The Dataset 16

The Dataset 16 dataset has 50 samples with 3 observations each. The results for the inter and intra class distance can be seen in Figure 4.18a. The samples are of reasonable quality although the inter class distance pdf shows they are somewhat less distinctive than the Red or Dataset 15 datasets. Synchronisation results are good, as the intra class distance curve is reasonable spike like.

Major drawback of this particular dataset, is the fact that the labels have more clutter and disturbing elements that make extraction of the R-shape more hard. This dataset required quite significant morphological processing prior to synchronisation. Morphological processing is nearly always highly specific for the acquisition conditions under with the pictures where acquired, making such processing undesirable.

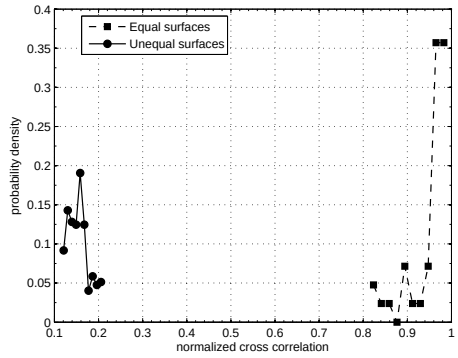
Dataset 15	
Device	Camera B, Type A LED ring
Quantity	14 * 3
MS	17500 bytes
Metric	Cross correlation
Post	global mean subtraction



(a)

Table 4.5 – Inter and intra class distance for **Dataset 15**.

Dataset 15	
Device	Camera B, Type A LED ring
Quantity	14 * 3
MS	17500 bytes
Metric	Cross correlation
Post	global mean subtraction
	discarded ill focussed samples



(a)

Table 4.6 – Inter and intra class distance for **Dataset 15**. Note that this dataset has been rid of all ill focussed acquisitions.

	Data 11	Data 4	Data 4	Data 3	Data 1
$I(X,Y)$	0.4	1.3	0.9	2.4	1.3
n	1188	133331	58800	1024	46631
# distinguishable signals	$2^{0.4n}$	$2^{1.3n}$	$2^{0.9n}$	$2^{2.4n}$	$2^{1.3n}$
Probability of error	$2^{-0.4n}$	$2^{-1.3n}$	$2^{-0.9n}$	$2^{-2.4n}$	$2^{-1.3n}$

Table 4.7 – Expected number of distinguishable signals and probability of error. n denotes the number of bits in the micro-structure.

4.2.2 Closing thoughts

Blur

The single most damaging factor to micro-structure quality is blur. Being defocussing blur, blurring induced by excessive interpolation or by the usage of a diffused lighting source. In al cases the blurring acts as a low pass filter, which removes detail from the micro-structures and induces local correlations.

The current industrial micro-scope has a depth of field (DOV) of 0.04 mm. This means that if this system is deployed above a conveyor belt it may not induce vibrations. Samples that are light, such as paper or carton will have to be positioned and pressed down flat under the camera.

Wear and tear

Although not considered within this work, future research should most definitely start testing with samples that have been in use and exhibit normal wear and tear patterns. They will most certainly destroy part of the informative content and furthermore will likely necessitate a more complex micro-structure extraction procedure.

Sample cardinality

Tables 4.2.1 and 4.1 show that the current datasets are information rich in terms of their bit entropy. Furthermore their mutual information suggests that

is would be possible to have a database that holds well over a million micro-structures whilst achieving a negligible probability of error. Given the very limited size of our databases, the largest one having 96 unique samples, this is only but a slight indication that should only serve as an encouragement to undertake a realistically large acquisition of the well performing datasets.

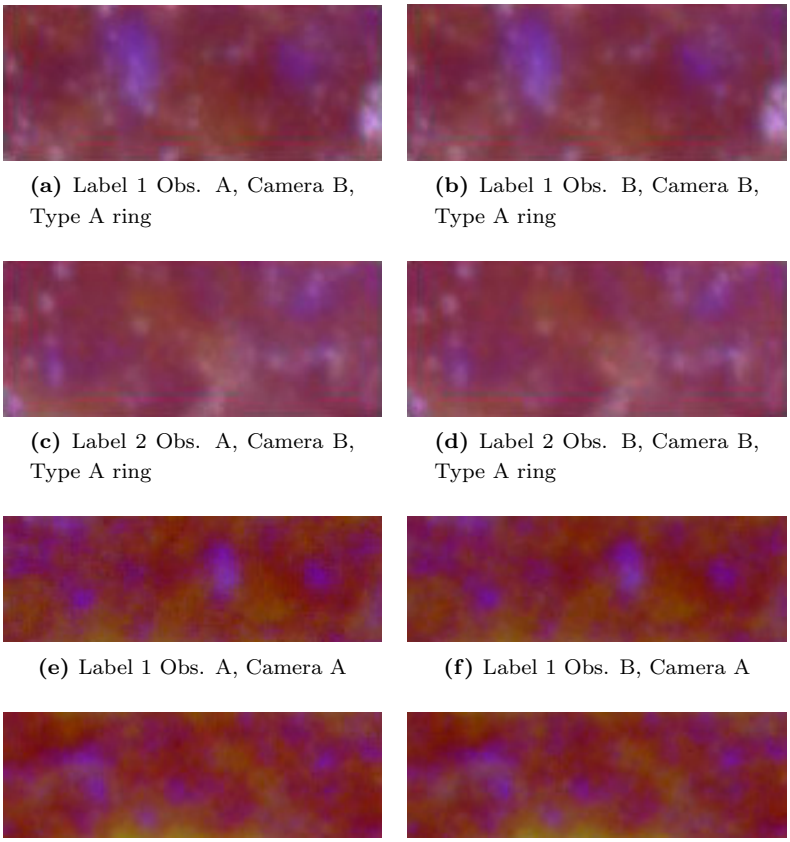
4.2.3 Device mismatch

As stated in the introduction, the ultimate goal of this framework is to have a different imaging device for the enrollment and for identification in the field. The enrollment device ideally is a high speed high quality device that can photograph samples on a running conveyor belt. The device that is used for identification in the field should be portable, simple to use, feature-less and cheap. The gap between the images obtained by the enrollment camera and the identification device can be bridged in two ways. The identification device is custom designed to emulate the enrollment device as close as possible, or imaging software is used. This section presents some preliminary tests for the latter approach. Examples of the different images that are acquired from an identical sample can be seen in Figure 4.19. Our results indicate that the biggest problem is caused by the used lighting.

Camera A versus Camera B

Matching Camera A against Camera B is currently actively researched. Initial results, as seen in Figure 4.2.3, are not that good. Both intra and inter class pdf's are long tailed and overlap. There is still a significant mismatch in the acquired image, as can be seen in Figures 4.19 and 2.3. One that can not simply be bridged by taking the cross correlation value.

Further work will focus on physically adapting both Camera B and Camera A light to create more similar images.



(a) Label 1 Obs. A, Camera B, Type A ring

(b) Label 1 Obs. B, Camera B, Type A ring

(c) Label 2 Obs. A, Camera B, Type A ring

(d) Label 2 Obs. B, Camera B, Type A ring

(e) Label 1 Obs. A, Camera A

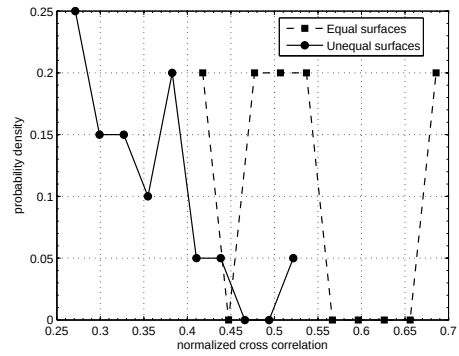
(f) Label 1 Obs. B, Camera A

(g) Label 2 Obs. A, Camera A

(h) Label 2 Obs. B, Camera A

Figure 4.19 – Examples of micro-structures from identical samples and from different samples acquired by Camera B and the Type A LED ring and the Camera A micro-scope.

Dataset 14	
Device	Camera A vs Cam
Quantity	12
MS	4000 bytes
Metric	Cross Correlation
Post	global mean subtraction



(a)

Figure 4.20 – Inter and intra class distance for **Dataset 14** acquired with Camera B for enrollment and Camera A for identification.

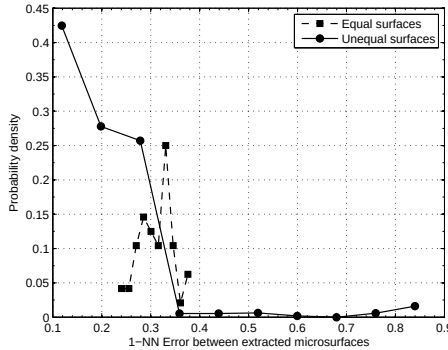
Camera A versus Camera D

Camera A and Camera D take visually similar images even although Camera D used ambient lighting at acquisition and Camera A had a semi LED ring light. The most significant problem faced is the fact that Camera D has about half the magnification Camera A has. Images from Camera D need therefor be to up-sampled by a factor two prior to matching. Initial results show that identification with this framework is currently not possible, but are not entirely pessimistic. The intra class distance pdf is still quiet narrow suggesting good synchronisation. The inter class distance pdf shows that about 70 procent of the dataset has a normalized cross-correlation below 0.2 which is a promising start. The distribution does have a very long tail that ventures almost up to 0.9. About 4 procent of the dataset has very high cross correlation values between observations of unequal samples. Part of the tail is undoubtedly caused by the up-sampling, which blurs the micro-structures slightly.

Camera D also induced some blurring due to the fact that it was used with ambient lighting.

Careful focussing, and adding a ring light to Camera D will most likely

Dataset 4	
Device	Camera A vs Cam
Quantity	144
MS	132600 bytes
Metric	Cross Correlation
Post	global mean subtraction histogram equalization



(a)

Figure 4.21 – Inter and intra class distance for **Dataset 4** acquired with Camera D for enrollment and Camera A for identification.

improve the results. The initial results definitely merit further investment.

I 'm feeling lucky.
– Google.com (1998 - present)

Chapter 5

Future Explorations in Identification

5.1 Introduction

Previous work on identifying micro-structures by [10, 51] utilized cross-correlation to match a newly acquired sample to the sample database. This requires an exhaustive search over all samples and requires the endless calculations of cross-correlations. With M samples of length N a single retrieval would require $\Omega(M)$ lookups, and the cross-correlations would require computations in the order of $\Omega(MN^2)$.

The natural solution is to seek some form of dimensionality reduction. This reduction should adhere to a number of stringent requirements.

- The reduced dataset should achieve the same probability of error as its original counterpart for identification.
- The reduction method should be scalable to potentially millions of samples and should deal with a dataset that rapidly changes over time.
- The method should achieve significant speeds up in both look up and in the number of computations required to ascertain a match.

The scalability requirement ensures the quick dismissal of a number of techniques that are regularly deployed in dimension reduction and classification problems. Principal Component Analysis (PCA) is not only computational intensive, but also discards small details when reconstructing the dataset with less dimensions. Detail that is important for identification. Linear Discriminant Analysis (LDA) and the Fischer Discriminant seek a projection that best separates classes. Other then also being computational intensive, its major issue is that this analysis has to be done again, the moment the dataset changes. Our dataset is prone to regular updates as new samples are enrolled and is far to big for such a measure.

In lieu of the results of Chapter 4 that show less than perfect synchronisation and evidence of labels not being truly *i.i.d*, we specifically seek a form of fingerprinting that also satisfies the following:

- The reduction should retain a certain invariance to geometric distortions.
- The retained feature set should enhance the inter class distance

This directly excludes any form of cryptographic hashing.

As the R dataset exhibits both high levels of cross correlation between unequal labels, and suffers from identification errors as shown in Figure 4.7a it will serve as the testing set to explore what the future possibilities are in robust feature extraction.

5.2 Cross correlation and Coefficients

To examine how components of individual micro-structures from the R dataset contribute to the final cross correlation value the following test was run.

- For each two micro-structures $\boldsymbol{x}$ and $\boldsymbol{y}$ that are compared, take 1 and sort its components by magnitude. Than progressively take the top 32 coefficients while masking of the rest of the microstructure and run the comparison metric. For a microstructure of 1024 bits, this means that the metric is run 10 times.

- The initial metric is the max peak value of the 2D dimensional cross correlation for micro-structures $\boldsymbol{x}$ and $\boldsymbol{y}$ where $\boldsymbol{a}$ has an increasing number of coefficients, i.e

$$\rho_{xy} = \mathcal{F}^{-1}(\mathcal{F}(\boldsymbol{x}).*\mathcal{F}^{*}(\boldsymbol{y}) \tag{5.1}$$

$$metric = \arg\max_{i,j} \rho_{xy} \tag{5.2}$$

$$\tag{5.3}$$

  Note that this manner of comparison is invariant for translation.

- All maximum cross correlation values over all (10) groups of coefficients make up a curve. The maximum of this curve is used as the final version.

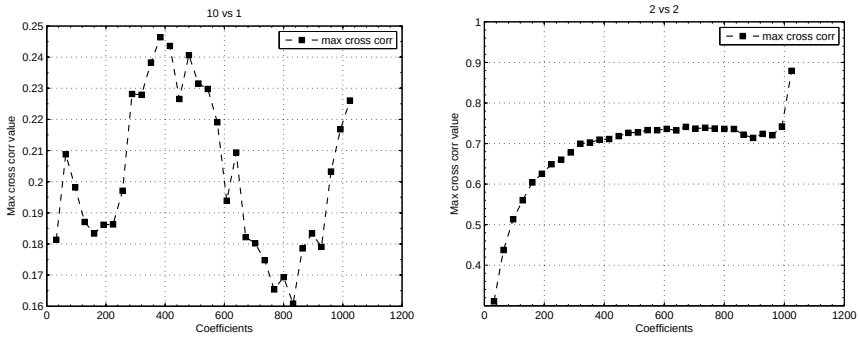
Using the maximum cross correlation value over all coefficients gave a significant better result in the intra en inter class separation as can be seen in Figure 5.2a and 5.2b.

The first step towards a robust feature is based on an observation from the R Dataset on how cross correlation coefficients make up the final result. Two examples of the curves formed by the cross correlation values over all coefficients for equal and unequal labels can be seen in Figure 5.1b and Figure 5.1a. Deploying two simple heuristics that stipulate that:

- Equal labels have monotonically increasing curves
- Equal labels have curves in which the maximum cross correlation value is attained when using all coefficients

Resulted in the best inter and intra class separation for Dataset 3 so far, as can be seen in Figure 5.2c. The tails still overlap, but this overlap constitutes about 0.1 percent of the dataset. There are 77 combinations between unequal labels that give relatively high cross correlation values out of a total of 66000 possible combinations.

Calculating the cross correlation in this way, does of course mean that pending the step in coefficient block size one takes (10 blocks of 32 in this example) the computational load increases with that same factor. Obviously this is not



(a) Label 10 versus label 3. **(b)** label 2A versus label 2B.

Figure 5.1 – Maximum 2D cross correlation value versus the number of largest micro-structure coefficients, in terms of the magnitude, that are retained for the **Dataset 3** dataset.

advisable in any situation, but this type of operation is easily paralyzed as all comparisons are completely independent of each other.

5.3 Random Projections

Random projections were conceived by Fridrich [55] and Lefebvre [37]. The method was applied to an authentication and retrieval framework by Voloshynovskiy and Koval [59]. The concept is simple, elegant and powerful.

Given a datavector $\boldsymbol{y} \in \mathbb{R}^{N \times M}$ and a matrix $\boldsymbol{\Phi} \in \mathbb{R}^{M \times L}$ the random projection is defined as:

$$\tilde{\boldsymbol{y}} = \boldsymbol{\Phi}\boldsymbol{y} \tag{5.4}$$

The resulting vector is then $\tilde{\boldsymbol{y}} \in \mathbb{R}^{L}$. The matrix $\boldsymbol{\Phi}$ is simply known as the *projector*. All its $L \times M$ elements ϕ are random draws from the Gaussian distribution i.e

$$\phi \sim \mathcal{N}(0, \frac{1}{M}) \tag{5.5}$$

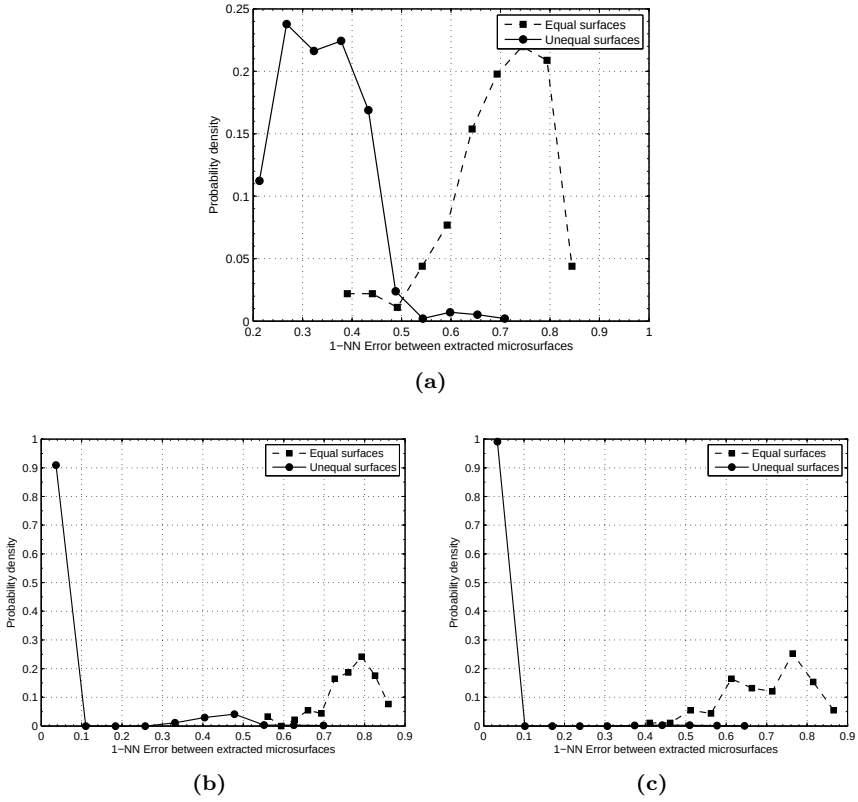
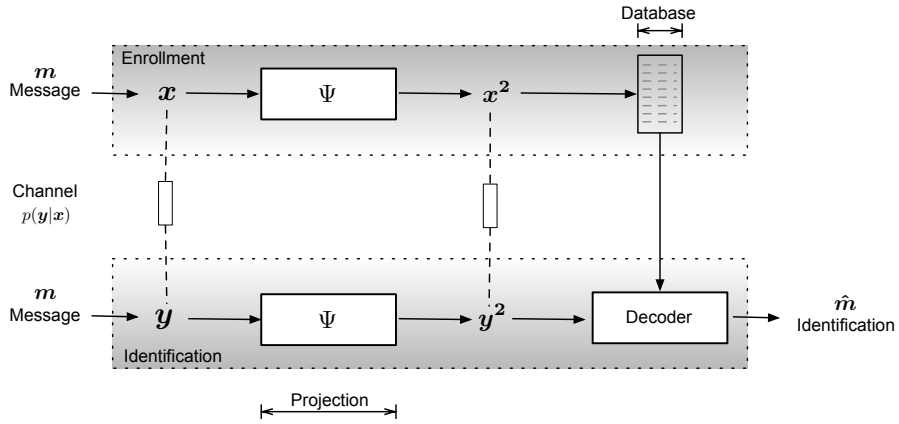


Figure 5.2 – Intra and inter class distance for **Dataset 3** using cross correlation (Figure5.2a), the maximum 2D cross correlation value over all coefficients (Figure 5.2b) and with heuristics (Figure 5.2c).



(a) Random projection framework

Figure 5.3 –

The random projector serves three important purposes:

- Our framework deals with data whose distribution is inherently unknown. After the random projections the distribution is guaranteed to be gaussian. This fact has been used in the work of [] to enhance the performance of traditional classification algorithms.
- The random projection can be used as a form of Monte Carlo inspired, or unsupervised, dimensionallity reduction.
- The loss of information is analytical tractable when using random projections [3, 59]. This is not the case when using techniques a such as LDA or PCA.

5.3.1 Dimension reduction

Random projections can be used for dimensionallity reduction by ensuring that for data vector $\boldsymbol{y} \in \mathbb{R}^{N \times M}$ the projection matrix is $\boldsymbol{\Phi} \in \mathbb{R}^{M \times L}$ with $L < M$.

An example of this can be seen in Figure 5.5 where a microstructure of 58800 bytes has been reduced to a 1024 bytes. The gap between the intra en inter class difference has stayed about the same.

This is expected, following the *data-processing inequality*[14] that states that no data manipulation can improve the inferences that can be made from the data. Formally:

$$X \rightarrow Y \rightarrow Z$$
$$I(X;Y) \geq I(X;Z) \tag{5.6}$$

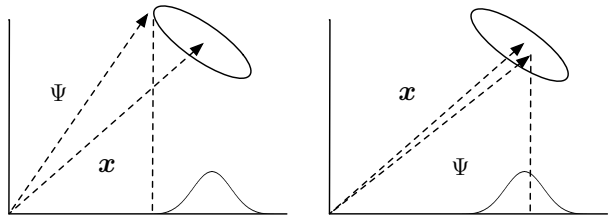
So, there exists no magic processing that turns Y into Z in such a way that the information Y contains about X is increased.

The fact that the separation between the intra and inter class distance has stayed about the same suggests that the extracted micro-structure of 58800 bytes contained more information then is strictly needed to be uniquely identifiable within this particular dataset.

Obviously there is a deep relation between the mutual information and bit entropy for the random samples and the dimension reduction that is achievable with random projections whilst retaining separation between the inter and intra class distances for all samples in the database. For true $i.i.d$ gaussian samples, the expected probability of error is mathematically derivable from the the number of used projections [60].

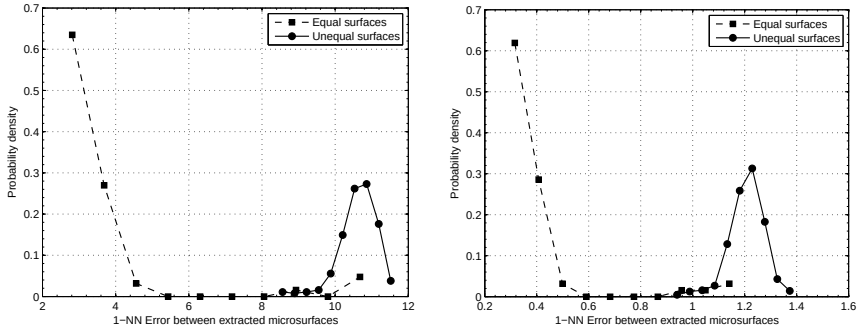
5.3.2 Smoothing of the Projector

Obviously not all random projections are optimal as illustrated in Figure 5.4a. This situation can be improved by blurring the random projection matrix $\mathbf{\Phi}$ with another gaussian filter. The effect is shown in Figure 5.6. Figure 5.6a shows the intra inter class distance probabilities for the Dataset 3 dataset. Here the 1024 micro-structures have been projected using an equal number of projections. The result is slightly worse than without projections, as one can see in Figure 5.2a. Improvement is achieved by smoothing the projection matrix $\mathbf{\Phi}$ with a gaussian filter with $\sigma = 2$ and a region of 5×5. It can be seen in Figure 5.6b.



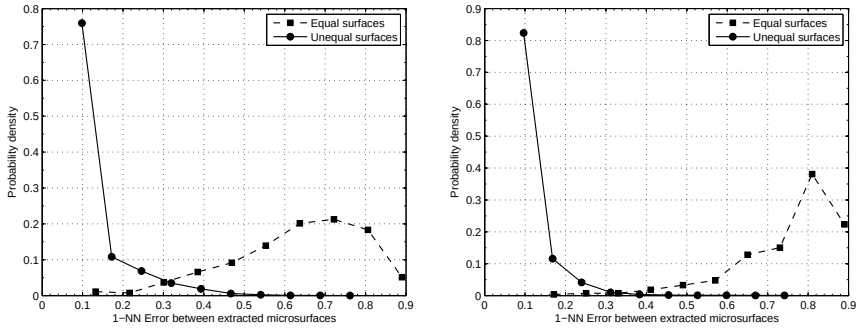
(a)

Figure 5.4 – The effect of different projections $\mathbf{\Phi}$ on datavector $\boldsymbol{x}$.



(a) **(b)**

Figure 5.5 – The inter and intra class distance for **Dataset 4**. The used metric is the Euclidian distance. Figure 5.5a shows the base line situation with microstructures of 58800 bytes. Figure 5.5b shows the same result but with microstructures that have been reduced sixty fold to 1024 bits.



(a) **(b)**

Figure 5.6 – The inter and intra class distance for **Dataset 3**. Figure 5.6a shows the inter and intra class distance after projection of the 1024 bit microstructure onto a 1024 projections. Figure 5.6b shows an equal set up but here the projection matrix has been blurred with gaussian filter with $\sigma = 2$ in a 5×5 neighborhood.

5.4 Reliable Components and Fast Searching

Extraction of reliable components from micro-structures serves two main purposes. If the retained information is sufficient for identification within the dataset it can be used for dimension reduction. Secondly, reliable components can be used to significantly speed up database searches. As we envision that databases will hold a million and more micro-structure samples in the future, this is the most promising application. Especially as currently the only way to resolve an identification query is to exhaustively compare the query against the database.

Two principal methods for reliable component extraction where tested:

- Random Projections and Magnitude Sorting
- Local Variances

5.4.1 Random Projections and Magnitude Sorting

Random projections and magnitude sorting as a method to extract the most reliable components after projection was conceived by [59]. Its application for fast searching stems from [22] who successfully used the method on a database with a million real landscape and nature images and on artificial data. This section will test the method on the available micro-structure databases.

Algorithm

The overview for the reliable component extraction can be seen in Figure 5.7a and resolving a query in Figure 5.7b. The reliable components in this framework are extracted as follows:

- The database or codebook with M microstructures of N length is projected with random projection matrix $\mathbf{\Phi}$ of $N \times L$ size, where $L \geq N$.
- As shown in Figure 5.4a and Section 5.3.2 not all projections are of good quality. Projection results in the $M \times N$ projected codebook are therefore sorted by their absolute value per row.

- From each sorted row in the projected codebook, only the top n values are retained and written back to the new reduced codebook.

- The bit positions of the top n components are stored separately.

Tests

The algorithm was tested on an artificial database with i.i.d samples to serve as a base line test case, and on the Dataset 14 , Dataset 3, Dataset 1, Dataset 11 and Dataset 16 databases. The performed test is almost equal to validation procedure outlined in Section 4.1.5 and Figure 4.2 and consists of determining the inter and intra class probability density functions for the reduced codebook. The single difference in the validation procedure is between the simulated enrollment and verification stage. One set of micro-structures serves as an enrolled set in the database. From this set the positions of the reliable components are determined. When testing against a second set of observations, these are also projected, but the reliable components are extracted using the earlier determined bit positions from the enrolled set. Results can be seen in Figure 5.7 and 5.9.

The base case uses two codebooks of a 512 samples with gaussian *i.i.d* data with a length of a 1024 bits. The second codebook with the second set of observations has a signal-to-noise ration of (SNR) -15 DB. The results before and after magnitude sorting and dimension reduction are shown in Figure 5.8a and 5.8b. The results in terms of separability are more or less the same before and after the dimension reduction to 32 bits. Following the *data-processing inequality* (Equation 5.6) this is the best attainable result. It should come to no surprise that 32 random bits, especially if they are selected with care, are sufficient to identify $2^9 = 512$ samples. Adding more noise to the second codebook will naturally make separation worse, but the purpose of the base gaussian case is only to establish a performance indicator for the other real micro-structure datasets.

All the micro-structure datasets with real data where tested using an extracted micro-structure of a 1024 bits, which is considerably smaller than the

sizes that where previously used in this framework. All extracted micro-structures were then reduced to 32 bits using the random projection and magnitude sorting method. All datasets show a small reduction of performance, most notably the Dataset 11 set (Figures 5.9c and **??**), although most manage to maintain separability.

Fast searching

The Random Projection and Magnitude sorting algorithm can be used to accelerate searching in a large scale database as follows. See Figure 5.7b.

- Reliable components are extracted from a noisy query using the Random Projection and Magnitude sorting algorithm.
- Using these reliable components the database is reduced in size by selecting only the entries that match to the reliable query bits.
- The reduced database is searched exhaustively.

To prove that the extracted reliable components are indeed an advantage and do not amount to a plain binary tree search in which the reliable components are simply taken first, tests were also run without the random projections while sorting was based on plain pixel magnitude values. These results can be seen in Figure 5.11a and show a significant drop in performance.

5.4.2 Local Variances

Tests was conducted using local variance peak values as reliable components. An example can be seen in Figure 5.10. The procedure is as follows:

- Local variances are determined for the entire acquired image. This is done to prevent border artifacts.
- The local variances are extracted from the same region as the micro-structure would in a conventional case.

- Local extrema are detected from the local variances. Their positions and value form the new reliable components.

Tests where done on a number of datasets. The results for Dataset 16 can be seen in Figure 5.11b. It shows clearly that the reliable components from local (peak) variances are significantly outperformed by the components ascertained with Random Projections and Magnitude Sorting.

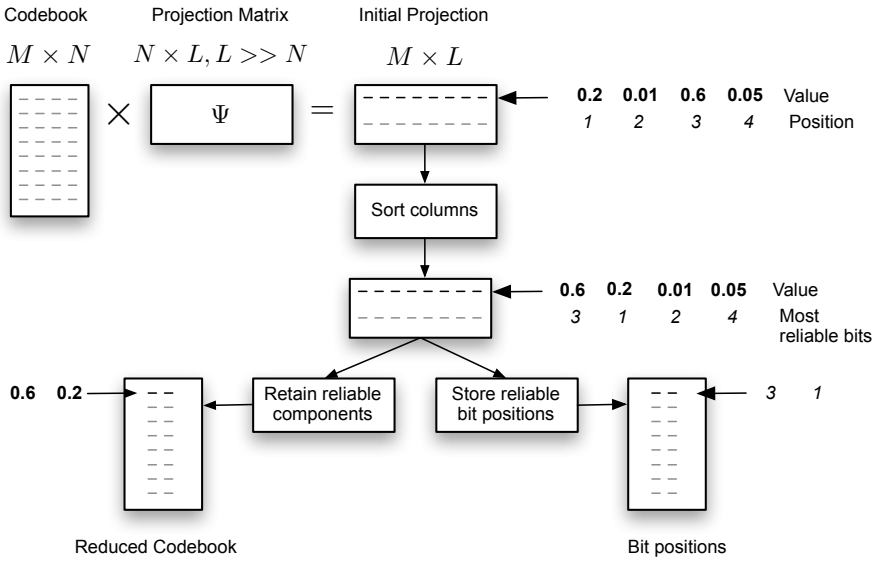
5.5 Circular micro-structure extraction

In the ideal case, rotation can be recovered up to a tenth of a degree. To amend this, micro-structure is usually taken from the center of the image. The other approach is to extract a region in such a way that the resulting structure is rotation invariant. Following the philosophy behind the Trace Transform algorithm [42] [7] [6] and most IRIS recognition systems [17] the micro-structure was extracted from a circular region. An overview of the method can be seen in Figure 5.12. In general the algorithm performs the following steps:

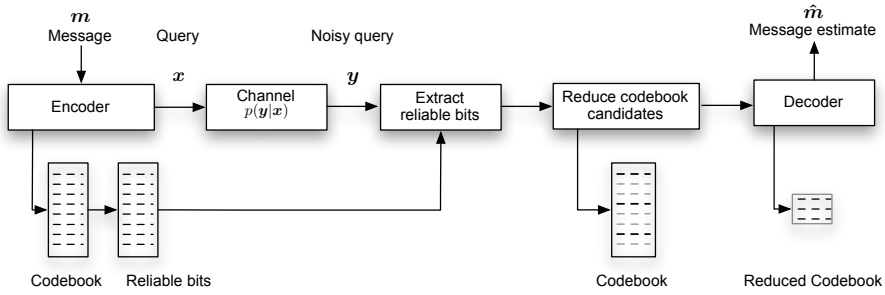
- Select the center of the image, the length of the tracing line ρ and the angle θ range.
- For every increment in θ, trace all the pixel values along ρ. The mapping requires interpolation when mapping all (x, y) coordinates on the line to (ρ, θ) coordinates.

In this fashion micro-structure values in the (x, y) coordinate plane are mapped to a rectangular (ρ, θ) plane. Any rotation deformation now becomes a translation. Using normalized cross correlation as a comparison metric, which is translation invariant, the net achieved effect is that the micro-structure comparison has become rotation invariant.

This algorithm has one important implementation choice. Pixel values originate from a discrete grid, which doesn't map one-to-one to the (ρ, θ) plane. If ones converts every (x, y) coordinate to a (ρ, θ) pair, not all (ρ, θ) values will be filled. One can also choose to define a (ρ, θ) space with a certain resolution and

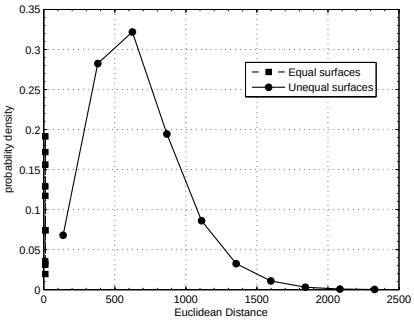


(a) Schematic figure of the random projections and magnitude sorting procedure to ascertain the most significant bits in a codebook, or micro-structure.

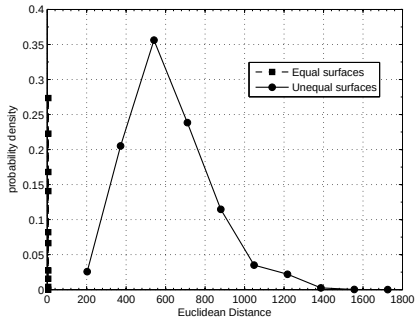


(b) Using reliable bits query bits to reduce the number of candidates prior to decoding.

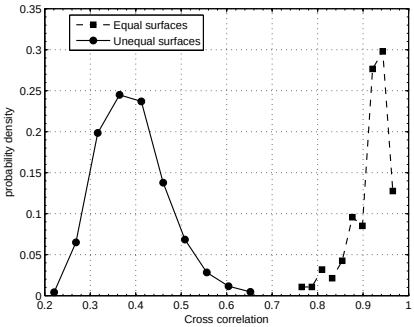
Figure 5.7 – Random projections and Magnitude sorting schematics.



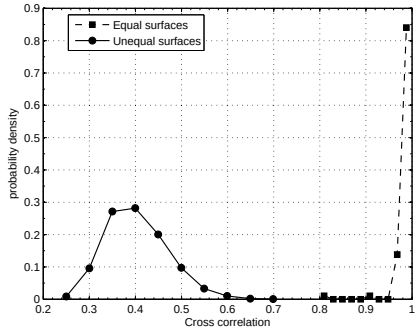
(a) Base case, 1024 bits gaussian $i.i.d$ sample.



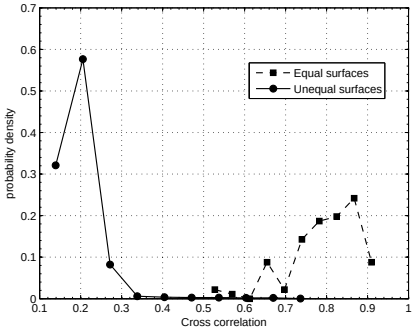
(b) Base case, reduced to 32 bits.



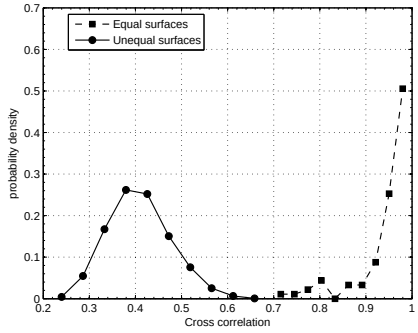
(c) Base, 1024 bits Dataset 14 (**??**).



(d) Dataset 14, reduced to 32 bits.

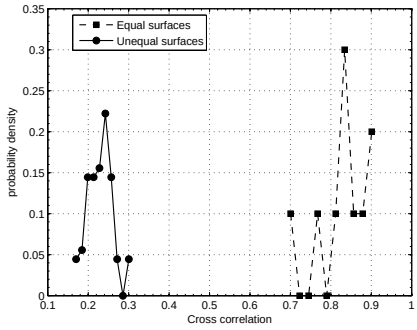


(e) Base, 1024 bits Dataset 3.

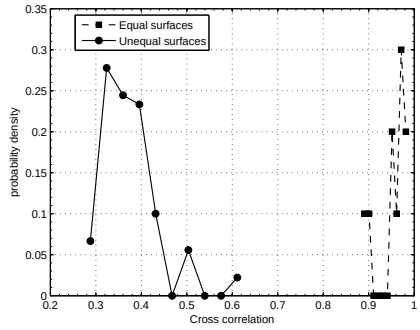


(f) Dataset 3, reduced to 32 bits.

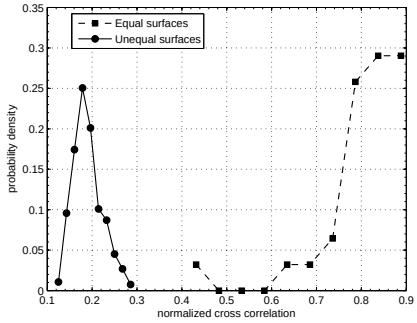
Figure 5.8 – Random Projections and Magnitude Sorting test results.



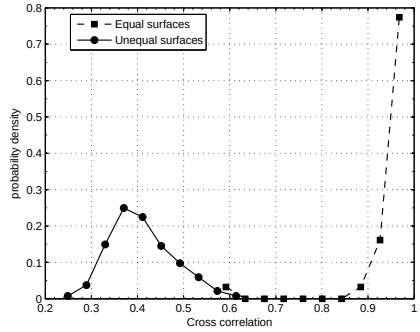
(a) Base, 1024 bits Dataset 1.



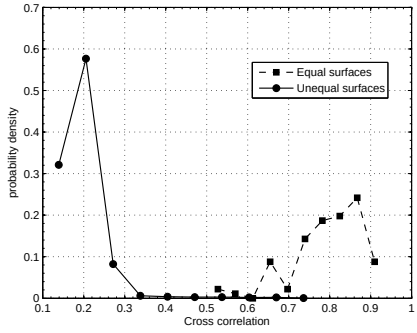
(b) Dataset 1, reduced to 32 bits.



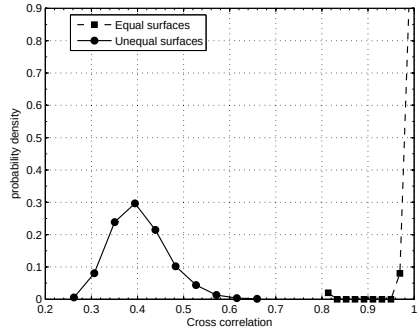
(c) Base, 1024 bits Dataset 11.



(d) Dataset 11, reduced to 32 bits.

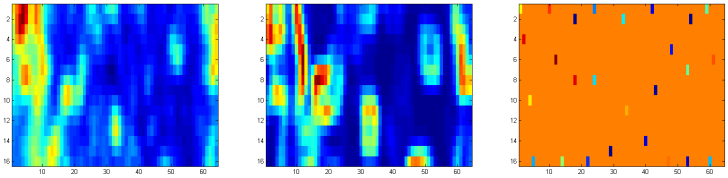


(e) Base, 1024 bits Dataset 16.



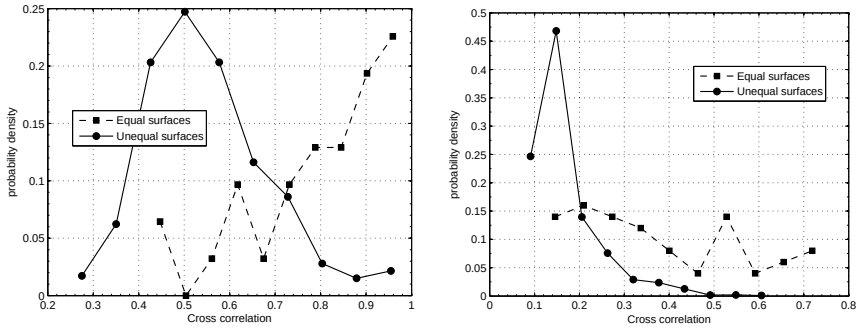
(f) Dataset 16, reduced to 32 bits.

Figure 5.9 – Random Projections and Magnitude Sorting test results.



(a) Pixel values **(b)** Local variance **(c)** Reliable components

Figure 5.10 – Example results for a single micro-structure from **Dataset 16** and using local variances as reliable components.



(a) Results for Magnitude Sorting and the Dataset 16 database using the 32 largest components.

(b) Results for Local Variances and the Dataset 16 database using the 32 largest components.

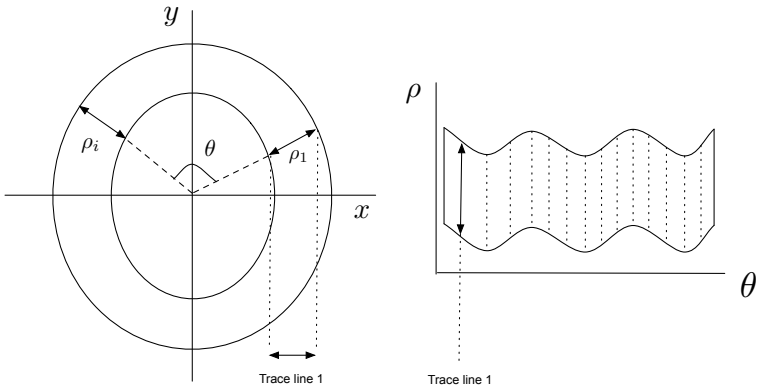
Figure 5.11 – Intra and inter class distances for **Dataset 16**.

then sample the (x, y) accordingly. This will lead to some duplication of the data, as small deviations in (ρ, θ) values will map the same (x, y) coordinate. The implementation of this framework follows the last approach.

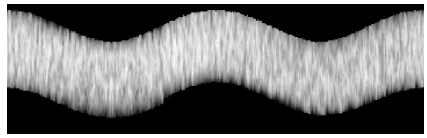
5.5.1 Results

Results for Dataset 14 can be seen in Figure 5.12a. The extracted region is the circular red circle around the R. The circular mapping does not show a dramatic improvement. This in itself is not a bad thing. Earlier tests on the Dataset 14 dataset in Figure 4.14a already show excellent results in terms of synchronisation and separability of the micro-structures. If the micro-structures are truly i.i.d and well synchronized, the specific extraction place and the size of the micro-structure should not influence the results.

The trace transform is used as a basis for binary object signatures in a number of works [25] [24] [48]. In this role the trace transform will remain to be a subject of ongoing and future research.

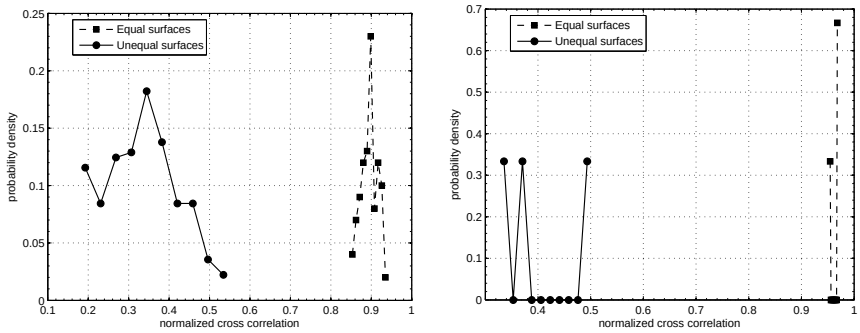


(a)



(b)

Figure 5.12 –



(a) **(b)**

Figure 5.13 – The intra and inter class distance for **Dataset 14** using standard micro-structure extraction and using circular micro-structure extraction. The latter improves results enough to merit further investigation.

You can't always get what you want.
– Mick Jager (1943 - present)

Chapter 6

Conclusion

This work has researched the possibilities and limitations of using physical micro-structures for identification. Using the physical structure itself from object negates the need to add a special marker for identification. Buchanan *et. al.* [10] found that paper documents, packaging and plastic cards contain microscopic surface structures that are unique for the sample. The natural occurring randomness forms both a unique and currently unclonable identification token.

The envisioned architecture that is simulated in this work (Figure 1.5) is comprised as follows. Objects are photographed whilst in the factory with a high speed industrial camera and micro-scope. These images are processed and stored in a database. An agent in the field, who wishes to identify an object, for example a customs officer, uses a small portable micro-scope to take a picture. This picture is send to an identification server that compares the query image against the database with enrolled samples. A successful query informs the user that the object under inspection indeed originated from the source it claims.

In the context of this architecture, this work addresses two research questions. The first goal is to extracted a micro-structure from exactly the right location, independent of how the acquisition was done, i.e

- *How can the geometrical distortions and imaging artifacts be removed.*

The second part of this work deals with the identification system as a whole

and explores the fundamental bounds of such a framework given the real data at hand. The two main concerns are *database scalability* and *identification accuracy*. The main research question that thus will be answered for every dataset, imaging device and processing algorithm therefore is:

- *What is the maximum number of unique distinguishable samples the database will hold.*

6.1 Image Synchronisation

Image synchronisation is achieved by stipulating that some fixed part of the originating sample, such as a letter or logo must be in the field of view. This template is used both in the enrollment and verification stage to ascertain the exact region from which the micro-structure is extracted. Any occurring geometrical transformation needs to be corrected.

An algorithm has been developed based on invariant features. The feature based algorithm combines a good edge map with so called SIFT points followed by robust estimation. The latter is a combination of Hough-pose space clustering and RANSAC. The major advantage is the fact that this algorithm can deal with a substantial amount ($\geq 40\%$) of outliers.

The single biggest challenge that faced the synchronisation algorithms were templates that where manufactured sloppy which caused ambiguity when finding the optimal geometrical transformation to extract the micro-structure.

6.2 Identification Limits

A binarized microstructure image is essentially a (noisy) signal. Its components or pixels X, with alphabet $\mathcal{X}$ can be seen as realizations ('draws') from a distribution $p(\boldsymbol{x}) = Pr[X = \boldsymbol{x}]$. The primary assumption, seen schematically in Figure 1.4, thus is that micro-structures are independently and identically distributed ($i.i.d$) random samples. This enables us to completely model the system using Shannon's communication model [50, 14, 33]. It is used to estab-

lish the boundaries of the number of samples of n length a database can hold while being able to resolve identification queries with negligible error.

The primary test to ascertain the suitability of micro-structures and the quality of synchronisation is the empirical determination of the intra and inter class distance probability density functions. The most promising results where ascertained from Dataset 2 that was acquired by Camera B and the small LED Type A light and with Dataset 11. The worst performing set was Dataset 3 These labels were manufactured sloppy, which hindered synchronisation. Further more, their micro-structures exhibited local correlations which hinders identification.

With the exception of Dataset 3 preliminary results show that near errorless identification is possible for all datasets. Furthermore, initial tests show that their micro-structures hold sufficient information (bit entropy) to cover well over a million samples. This result should just be seen as positive indicator that validates acquiring a truly large dataset, as currently the biggest set has a mere 96 unique samples.

The single most damaging factor to micro-structure quality is blur. Being defocussing blur, blurring induced by excessive interpolation or by the usage of a diffused lighting source. In all cases the blurring acts as a low pass filter, which removes detail from the micro-structures and induces local correlations.

6.3 Future Work

Future works focusses on fingerprinting methods for micro-structures that fulfill either of the following requirements: The fingerprint reduces the dimensionality of the micro-structure. Secondly, a fingerprint should be constructed in such a way that it becomes invariant to minor geometrical transformations. The last goal is to extract a fingerprint from the micro-structure that can be used to speed up database queries in very large databases. Currently these three requirements are researched separately.

Experiments with dimensionality reduction focus on using Random Projections. Although more optimal de-correlation methods exist, they are computationally to intensive for large scale databases.

To create rotation invariant fingerprints from micro-structures, preliminary tests where done using the Trace Transform. This method has been used for invariant fingerprinting in a number of works [25] [24] [48], be it only on real life everyday images, not on micro-structures. Real life images are obviously governed by different statistics.

Reliable component extraction using Random Projections and Magnitude Sorting [59, 22] is currently actively tested on the available databases. The reliable components form the fingerprint that is used to drastically limit the search space in which the identification query is being resolved. Tests with all current available databases are promising.

Bibliography

[1] Authentication of articles. PATENT: WO/1997/024699.

[2] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *In ECCV*, pages 404–417, 2006.

[3] Fokko Beekhof, Sviatoslav Voloshynovskiy, Oleksiy Koval, and Renato Villán. Secure surface identification codes. In Edward J. Delp III, Ping Wah Wong, and Nasir D. Memon Jana Dittmann, editors, *Steganography, and Watermarking of Multimedia Contents X*, volume 6819 of *Proceedings of SPIE*, (SPIE, Bellingham, WA 2008) 68190D, 2008.

[4] Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft. When is "nearest neighbor" meaningful? In *In Int. Conf. on Database Theory*, pages 217–235, 1999.

[5] Christian Böhm, Stefan Berchtold, and Daniel A. Keim. Searching in high-dimensional spaces: Index structures for improving the performance of multimedia databases. *ACM Comput. Surv.*, 33(3):322–373, 2001.

[6] P. Brasnett and M. Bober. Fast and robust image identification. 2008. Mitsubishi Electric ITE-VIL.

[7] P. Brasnett and M.Z. Bober. A robust visual identifier using the trace transform. In *Visual Information Engineering Conference*, 2007.

[8] M. Brown and D. Lowe. Invariant features from interest point groups, 2002.

[9] Matthew Brown, Richard Szeliski, and Simon Winder. Multi-image matching using multi-scale oriented patches. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1*, pages 510–517, Washington, DC, USA, 2005. IEEE Computer Society.

[10] J. D. R. Buchanan, R. P. Cowburn, A.-V. Jausovec, D. Petit, P. Seem, G. Xiong, D. Atkinson, K. Fenton, D. A. Allwood, and M. T. Bryan. 'fingerprinting' documents and packaging. *Nature*, 436:475–+, July 2005.

[11] J Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, 1986.

[12] Robert Cockburn, Paul N Newton, E. Kyerematen Agyarko, Dora Akunyili, and Nicholas J White. The global threat of counterfeit drugs: Why industry and governments must communicate the dangers. *PLoS Med*, 2(4):e100, 03 2005.

[13] Donatello Conte, Pasquale Foggia, and Mario Vento. Challenging complexity of maximum common subgraph detection algorithms: A performance analysis of three algorithms on a wide database of graphs.

[14] Thomas M. Cover and Joy A. Thomas. *Elements of information theory*. Wiley-Interscience, New York, NY, USA, 1991.

[15] J. Cox, L. Miller, A Bloom, J. Fridrich, and T. Kalker. *Digital Watermarking and Steganography*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2008.

[16] J. C. Dainty. *Laser Speckle and Related Phenomena*. 1984.

[17] J. Daugman. How iris recognition works. *Circuits and Systems for Video Technology, IEEE Transactions on*, 14(1):21 – 30, jan. 2004.

[18] John Daugman. How iris recognition works. 2004.

[19] D. de Ridder F. van der Heijden, R.P.W. Duin and D.M.J. Tax. *Classification, Parameter Estimation and State Estimation - An Engineering Approach using Matlab.* J. Wiley, 2004.

[20] C. Harris and M. Stephens. A combined corner and edge detection. In *Proceedings of The Fourth Alvey Vision Conference*, pages 147–151, 1988.

[21] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision.* Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[22] T. Holotyak, S. Voloshynovskiy, F. Beekhof, and O. Koval. Fast identification of highly distorted images. In *Proceedings of SPIE Photonics West, Electronic Imaging 2010 / Media Forensics and Security XII*, San Jose, USA, January 21–24 2010.

[23] Piotr Indyk. Nearest neighbors in high-dimensional spaces, 2004.

[24] A. Kadyrov and M. Petrou. Object signatures invariant to affine distortions derived from the trace transform. *Image and Vision Computing*, 21(13-14):1135 – 1143, 2003. British Machine Vision Computing 2001.

[25] Alexander Kadyrov and Maria Petrou. The trace transform and its applications. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(8):811–828, 2001.

[26] Ashraf Masood Kibriya. Fast algorithms for nearest neighbor search. Master's thesis, The University of Waikato, Hamilton, New Zealand, march 2007.

[27] J.J. Koenderink. The structure of images. 50:363–370, 1984.

[28] P. D. Kovesi. MATLAB and Octave functions for computer vision and image processing. School of Computer Science & Software Engineering, The University of Western Australia. Available from: <http://www.csse.uwa.edu.au/∼pk/research/matlabfns/>.

[29] Peter Kovesi. Image features from phase congruency, 1999.

[30] T. Lindeberg. Direct estimation of affine image deformations using visual front-end operations with automatic scale selection. In *ICCV '95: Proceedings of the Fifth International Conference on Computer Vision*, page 134, Washington, DC, USA, 1995. IEEE Computer Society.

[31] Tony Lindeberg. Scale-space theory in computer vision, 1994.

[32] D. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision*, volume 20, pages 91–110, 2003.

[33] D. J.C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.

[34] Wojciech Makowiecki and Witold Alda. New sky pattern recognition algorithm. In *ICCS '08: Proceedings of the 8th international conference on Computational Science, Part I*, pages 749–758, Berlin, Heidelberg, 2008. Springer-Verlag.

[35] Wendy L. Martinez and Angel R. Martinez. *Computational Statistics Handbook with MATLAB, Second Edition (Chapman & Hall/Crc Computer Science & Data Analysis)*. Chapman & Hall/CRC, 2007.

[36] Morgan McGuire. An image registration technique for recovering rotation, scale and translation parameters. *NEC Tech Report*, Feb. 1998.

[37] Mehmet Kivanç Mihçak and Ramarathnam Venkatesan. A perceptual audio hashing algorithm: A tool for robust audio identification and information hiding. In *IHW '01: Proceedings of the 4th International Workshop on Information Hiding*, pages 51–65, London, UK, 2001. Springer-Verlag.

[38] Krystian Mikolajczyk. *Detection of local features invariant to affines transformations*. PhD thesis, INPG, Grenoble, juillet 2002.

[39] Hans Moravec. Obstacle avoidance and navigation in the real world by a seeing robot rover. In *tech. report CMU-RI-TR-80-03, Robotics Institute, Carnegie Mellon University and doctoral dissertation, Stanford University*, number CMU-RI-TR-80-03. September 1980.

[40] Alan V. Oppenheim and Ronald W. Schafer. *Digital Signal Processing.* Prentice-Hall International Editions, 1975.

[41] Marcello Pelillo, Kaleem Siddiqi, and Steven W. Zucker. Matching hierarchical structures using association graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21:1105–1120, 1998.

[42] Maria Petrou and Fang Wang. *HANDBOOK OF TEXTURE ANALYSIS*, chapter A Tutorial on the Practical Implementation of the Trace Transform, pages 313–346. Imperial College Press, 2009.

[43] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing.* Cambridge University Press, 3 edition, September 2007.

[44] Rashid Qureshi, Jean-Yves Ramel, and Hubert Cardot. Graph based shapes representation and recognition, 2007.

[45] Bart M. ter Haar Romeny, Luc Florack, Alfons H. Salden, and Max A. Viergever. Higher order differential structure of images. In *IPMI '93: Proceedings of the 13th International Conference on Information Processing in Medical Imaging*, pages 77–93, London, UK, 1993. Springer-Verlag.

[46] Stephan R. Sain. Multivariate locally adaptive density estimation. *Comput. Stat. Data Anal.*, 39(2):165–186, 2002.

[47] Cordelia Schmid and Roger Mohr. Local grayvalue invariants for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530–535, 1997.

[48] Jin S. Seo, Jaap Haitsma, Ton Kalker, and Chang D. Yoo. A robust image fingerprinting system using the radon transform. *Signal Processing: Image Communication*, 19(4):325 – 339, 2004.

[49] Gregory Shakhnarovich, Trevor Darrell, and Piotr Indyk. *Nearest-Neighbor Methods in Learning and Vision: Theory and Practice (Neural Information Processing).* The MIT Press, 2006.

[50] Claude E. Shannon and Warren Weaver. *A Mathematical Theory of Communication*. University of Illinois Press, Champaign, IL, USA, 1963.

[51] Joshua R. Smith and Andrew V. Sutherland. Microstructure based indicia.

[52] I. Sobel and G Fieldman. A 3x3 isotropic gradient operator for image processing. talk at the Stanford Artificial Project, 1968.

[53] J. Nielsen Sporring and L.M. M. Florack. *Gaussian Scale-Space Theory*, volume 8 of *Series: Computational Imaging and Vision*. Springer, 1997.

[54] R.L. van Renesse. Paper based document security-a review. In *Security and Detection, 1997. ECOS 97., European Conference on*, pages 75–80, Apr 1997.

[55] R. Venkatesan, S. m. Koon, M. H. Jakubowski, and P. Moulin. Robust image hashing, 2000.

[56] P. Viola. *Alignment by Maximization of Mutual Information*. PhD thesis, Massachusetts Institute of Technology, AI lab, Aitr 1548, june 1995.

[57] P. Viola and W. Wells. Alignment by maximization of mutual information. In *International Journal of Computer Vision*, pages 16–23, 1995.

[58] S Voloshynovskiy, O. Koval, and T. Pun. Multimedia security. SIP CVML master course, Universite de Geneve, 2010.

[59] Sviatoslav Voloshynovskiy, Oleksiy Koval, Fokko Beekhof, and Thierry Pun. Random projections based item authentication. In *Proceedings of SPIE Photonics West, Electronic Imaging / Media Forensics and Security XI*, San Jose, USA, 2009.

[60] Sviatoslav Voloshynovskiy, Oleksiy Koval, Taras Holotyak, and Fokko Beekhof. Privacy enhancement of common randomness based authentication: key rate maximized case. IEEE Workshop on Information Forensics and Security, 2010.

[61] A.P. Witkin. Scale-space filtering. In *8th International Joint Conference on Artificial Intelligence,*, pages 1019–1023, Karlsruhe, Germany, 1983.

[62] Haim J. Wolfson and Isidore Rigoutsos. Geometric hashing: An overview, 1997.

[63] B. Zitova. Image registration methods: a survey. *Image and Vision Computing*, 21(11):977–1000, October 2003.

Appendix A

Image Features

A.1 The Harris Corner Detector

The Harris corner detector, or *Harris point* was designed by Harris en Stephens [20] as in improvement of Moravec's [39] corner detector.

Moravec's corner detection looks for changes in intensity while shifting from point to point in an image. For an image I, a window function w, shift vector (u, v) and image point (x, y), it is defined as:

$$E(u, v) = \sum w(x, y)[I(x + u, y + v) - I(x, y)]^2 \tag{A.1}$$

The window function is basically a binary mask that defines a square region of interest. It is 1 within the rectangle, 0 outside. The algorithm takes four shift values for (u, v), namely $(1, 0)$, $(1, 0)$, $(0, 1)$ and $(-1, 1)$. It then proceeds to hunt for local maxima in the set with minimum E values from the four shifts.

This algorithm suffers from a number of drawbacks. Primarily these are the binary window function which causes a noisy response and the limited number of shifts that is taken into consideration. Harris points were designed to overcome these flaws.

Harris corners use a Gaussian window function i.e. $w(x, y) = \exp(\frac{(x^2+y^2)}{2\sigma^2})$. Furthermore, it considers much smaller shifts by using a Taylor expansion of

the image derivatives. For an image I, a window function w, shift vector (u, v) and image point (x, y), it is defined as:

$$E(u,v) \cong (u,v)M \begin{pmatrix} u \\ v \end{pmatrix} \tag{A.2}$$

$$M = w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_y I_x & I_y^2 \end{bmatrix} \tag{A.3}$$

Points are then analyzed using the eigenvalues λ_1 and λ_2 of matrix M. A point on an edge is characterized by the fact that one of the eigenvalues is significantly larger than the other. A corner is indicated by sufficiently large and similar eigenvalues. See Figure A.1a. The 'corners-ness' measure R is defined as follows:

$$R = \det M - k(\text{trace } M)^2 \tag{A.4}$$

$$\det M = \lambda_1 \lambda_2 \tag{A.5}$$

$$\text{trace } M = \lambda_1 + \lambda_2 \tag{A.6}$$

Note that this prevents the explicit calculation of eigenvalues from matrix M which is computationally intensive.

The corner response R is rotation invariant, which naturally, is a direct consequence of the fact that it is based on eigenvalues. Harris points are, however, not invariant to scale. Large corners can also be classified as two big edges pending the scale the Harris detector is working in.

The basic algorithm is as follows:

1. Compute the image derivatives in both x and y direction. This can be done by convolving the image with a first order Gaussian derivative [38] i.e.:

$$I_x = G_\sigma^{'x} * I \tag{A.7}$$

$$I_y = G_\sigma^{'y} * I \tag{A.8}$$

2. For all pixels, calculate the product of the derivatives:

$$I_{x2} = I_x \cdot I_x \tag{A.9}$$
$$I_{y2} = I_y \cdot I_y \tag{A.10}$$
$$I_{xy} = I_x \cdot I_y \tag{A.11}$$

3. Convolve against a Gaussian window function:

$$\mathcal{L}_{x2} = G_{sigma^w} * I_{x2} \tag{A.12}$$
$$\mathcal{L}_{y2} = G_{sigma^w} * I_{y2} \tag{A.13}$$
$$\mathcal{L}_{xy} = G_{sigma^w} * I_{xy} \tag{A.14}$$

4. Form matrix M at each pixel point (x, y):

$$M(x,y) = \begin{bmatrix} \mathcal{L}_{x2}(x,y) & \mathcal{L}_{xy}(x,y) \\ \mathcal{L}_{xy}(x,y) & \mathcal{L}_{x2}(x,y) \end{bmatrix} \tag{A.15}$$

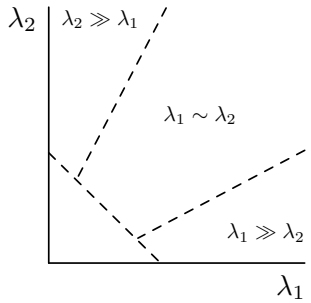
5. Compute the corner response R and threshold it optionally. Most implementations stipulate that potential Harris points should have a significantly stronger response than other points in there region of interest. This is also known as *non max suppression*.

$$R = \det M - k(\text{trace } M)^2 \tag{A.16}$$

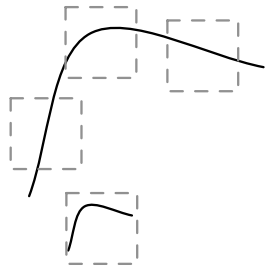
A.2 The Scale Invariant Feature Transform

The Scale Invariant Feature Transform (SIFT point) was conceived by David Lowe and Michel Brown. [32, 8] It has been proven to be lighting, scale and rotation invariant, and to some degree, invariant to affine transformations.

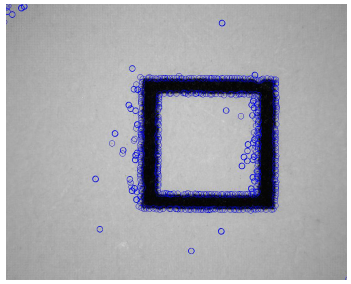
The basic algorithm consists of two parts, the detector that searches for maxima and minima in image scale-space, and the descriptor that transcribes the region of interest around a found extrema.



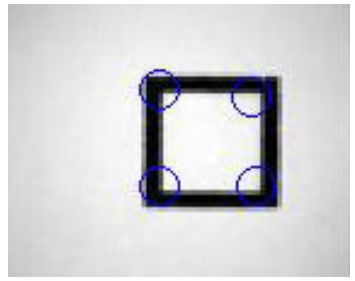
(a) Eigenvalues from matrix M.



(b) Harris detector and scale.



(c) Harris corners versus scale for big paper sample.



(d) Harris corners versus scale for small paper sample.

Figure A.1 – Harris corner detector properties. The scale figures can be generated with `demo_harris_scale.m`

Appendix B

Imaging

B.1 Parzen Window Density Estimation

Parzen estimation belongs to the class of non-parametric density estimation functions. These functions estimate conditional probability densities without any assumptions about the underlying model or its parameters. Instead they sample directly from a training set. The most elementary form of this technique is *histogramming*. The measurement space is partitioned in a number of disjoint sets, or *bins*. The number of samples that fall in each bin is counted. The number of samples per bin is then proportional to the estimated probability density for that partition of measurement space.

Suppose one wishes to estimate the conditional probability density function $p(\boldsymbol{x}|m_k)$: the probability that a certain measurement $\boldsymbol{x}$ originated from a sample belonging to class m_k [19]. Given that i denotes the number of bins R_i and $N_{k,i}$ is the total number of samples from class m_k that fall in bin i, the estimation can than be estimated as:

$$\hat{p}(\boldsymbol{x}|m_k) = \frac{N_{i,k}}{\text{Volume}(R_i) \cdot N_k} \tag{B.1}$$

Histogramming produces good results if the number of samples per bin is sufficient. This means that if there are little measurements, the bins should be broad

to accumulate enough votes. As a consequence the resolution of the estimate drops.

Parzen estimation can be thought of as an advanced form of histogramming. The main idea is that if one has a measurement $\boldsymbol{x}$ we can say something about the $p(x)$ at that point and at the points in the direct vicinity of $\boldsymbol{x}$. However, the further one moves away from $\boldsymbol{x}$, the less one can ascertain about $p(x)$. This behavior is governed by the *kernel*. The kernel K is usually the standard normal distribution, but in principle any probability density may be used.

Formally, the general form of the Parzen estimator is:

$$\hat{f}_{Ker}(\boldsymbol{x}) = \frac{1}{Nh} \sum_{i=1}^{n} K\left(\frac{\boldsymbol{x} - \boldsymbol{X}_i}{h}\right) \tag{B.2}$$

Where $\boldsymbol{X}_i$ is the stochastic variable for which the probability density is estimated over domain $\boldsymbol{x}$. Function K is the kernel and h is the so-called *smoothing* function. Using the standard Gaussian as kernel function, Equation B.2 can be written as:

$$\hat{f}_{Ker}(\boldsymbol{x}) = \frac{1}{Nh} \sum_{i=1}^{n} \frac{1}{\sqrt{2\pi}} \exp\left(\frac{(\boldsymbol{x} - \boldsymbol{X}_i)^2}{2h^2}\right) \tag{B.3}$$

The specific value of the smoothing function h is important. It more or less controls the variance of the Kernel function. If h its value is small, the kernel function gets a peak. This means that samples have great influence locally, and the variance of the estimator becomes large. If the value of h is chosen large, samples influence a larger part of the domain. Consequently the estimate becomes a smoothed version of the true probability density function. Pending the application, the designer can thus choose between variance and bias [19]. An example of this behavior can be seen in Figure B.1.

If a Gaussian distribution is used as kernel, the optimal smoothing parameter $\hat{h}$ and $\hat{\sigma}$ may be chosen as [46, 35]:

$$\hat{\boldsymbol{\sigma}} = 1.4826 \cdot \tilde{\boldsymbol{x}} \tag{B.4}$$

$$\hat{\boldsymbol{h}} = \hat{\sigma} \cdot \left(\frac{4}{3n}\right)^{\frac{1}{5}} \tag{B.5}$$

Where $\tilde{\boldsymbol{x}}$ is the median and n the total number of points in the domain.

The general algorithm, as presented by [35], for the Parzen estimator is:

- Determine a kernel K, a smoothing parameter h and the domain $\boldsymbol{x}$ over which to determine the Parzen estimate
- For each measurement from $\boldsymbol{X}_i$ with $i \in \{1..n\}$, evaluate the kernel K over the entire domain $\boldsymbol{x}$:

$$K_i = K\left(\frac{\boldsymbol{x} - \boldsymbol{X}_i}{h}\right) \tag{B.6}$$

  This results in n curves per measurement $\boldsymbol{X}_i$.
- Normalize all curves with the smoothing function: $1/h$.
- For each point $\boldsymbol{x}$ in the domain, take the average of alle normalized curves.

B.2 Error Metrics

There exist many more metrics to determine the difference between two images. In the remainder of this section we will explore the relation between mutual information and two other well known error metrics:

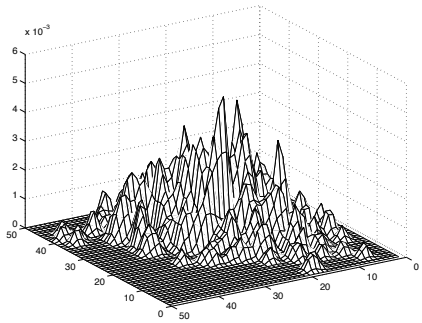
- the euclidian distance
- the cross correlation

Keeping the channel model (Figure B.2) in mind and the fact that mutual information is defined as:

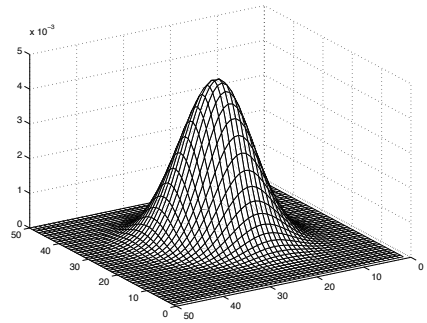
$$I(X;Y) = E\left(\log \frac{p(\boldsymbol{x},\boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})}\right) \tag{B.7}$$

When registering two images we are of course interested in the image that maximizes the average empirically ascertained mutual information. Formally one wishes:

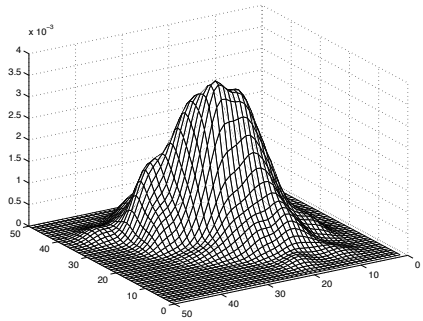
$$\max_X \bar{I}(X;Y) = \max_X \frac{1}{N} \sum \log \frac{p(\boldsymbol{x},\boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})} \tag{B.8}$$



(a) $\boldsymbol{h} = (0.1, 0.1)$.



(b) $\boldsymbol{h} = (1, 1)$.



(c) $\boldsymbol{h}$ determined via Equation B.4.

Figure B.1 – Using various smoothing values h with a Parzen estimator to empirically determine the joint probability density function $P_{x_1 y}$ where $\boldsymbol{y} = \rho \cdot \boldsymbol{x}_1 + \sqrt{1-\rho^2} \cdot \boldsymbol{x}_2$ and $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are *i.i.d* from the Normal distribution. The figures can be generated with `demo_parzen_m`.

Via Bayes' rule, $p(x, y) = p(y|x)p(x)$, and the fact that in this case $p(x)$ is a constant, one can refractor Equation B.8 to:

$$\max_X \bar{I}(X;Y) = \max_X \log p(\boldsymbol{y}|\boldsymbol{x}) \tag{B.9}$$

Now assuming that the channel $p(\boldsymbol{y}|\boldsymbol{x})$ induces gaussian noise the relation between $\boldsymbol{x}$ and $\boldsymbol{y}$ becomes:

$$\boldsymbol{y} \quad = \quad \boldsymbol{x} + n \tag{B.10}$$

Where:

$$n \quad \sim \quad N(0, \sigma_n^2) \tag{B.11}$$

Then $\boldsymbol{y}$ can be written as a function of $\boldsymbol{x}$:

$$\boldsymbol{y} \quad \sim \quad N(\boldsymbol{x}, \sigma_n^2) \tag{B.12}$$

$$\boldsymbol{y} \quad \sim \quad \frac{1}{\sigma\sqrt{2\pi}} \exp \frac{-|\boldsymbol{y} - \boldsymbol{x}|^2}{2\sigma_n^2} \tag{B.13}$$

The maximum likelihood $p(\boldsymbol{x}|\boldsymbol{y})$ for certain $\boldsymbol{x}$ can than be written as:

$$\max_{\boldsymbol{x}} p(\boldsymbol{x}|\boldsymbol{y}) \quad \sim \quad \frac{1}{\sigma\sqrt{2\pi}} \exp \frac{-|\boldsymbol{y} - \boldsymbol{x}|^2}{2\sigma_n^2} \tag{B.14}$$

Removing independent terms:

$$\max_{\boldsymbol{x}} p(\boldsymbol{x}|\boldsymbol{y}) = \exp -|\boldsymbol{y} - \boldsymbol{x}|^2 \tag{B.15}$$

And taking the log:

$$\max_{\boldsymbol{x}} p(\boldsymbol{x}|\boldsymbol{y}) \quad = \quad -|\boldsymbol{y} - \boldsymbol{x}|^2 \tag{B.16}$$

$$\min_{\boldsymbol{x}} p(\boldsymbol{x}|\boldsymbol{y}) \quad = \quad |\boldsymbol{y} - \boldsymbol{x}|^2 \tag{B.17}$$

Which in terms results in the Euclidian distance between $\boldsymbol{x}$ and $\boldsymbol{y}$. This means that if the channel distortions are gaussian, the euclidian distance is a suitable estimator for the mutual information between $\boldsymbol{x}$ and $\boldsymbol{y}$. Calculating the euclidian distance is of course much more computational efficient.

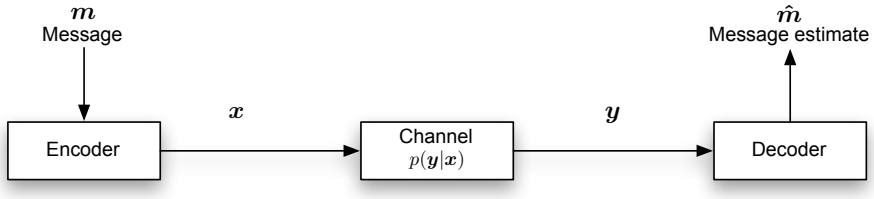


Figure B.2 – Channel model.

Cross correlation between vector $\boldsymbol{x}$ and $\boldsymbol{y}$ is defined as:

$$\rho = \frac{\sum\sum(\boldsymbol{x} - \mu_{\boldsymbol{x}})(\boldsymbol{y} - \mu_{\boldsymbol{y}})}{\sqrt{\sum\sum(\boldsymbol{x} - \mu_{\boldsymbol{x}})^2(\boldsymbol{y} - \mu_{\boldsymbol{y}})^2}} \tag{B.18}$$

From which it is apparent that r can be maximized by maximizing the nominator term $\boldsymbol{x'y'}^T$ and minimizing denominator $\|\boldsymbol{x'} - \boldsymbol{y'}\|$. The euclidian distance can be written as:

$$d = (\boldsymbol{x} - \boldsymbol{y})(\boldsymbol{x} - \boldsymbol{y})^T \tag{B.19}$$

$$d = \boldsymbol{x}\boldsymbol{x}^T - 2\boldsymbol{x}\boldsymbol{y}^T + \boldsymbol{y}\boldsymbol{y}^T \tag{B.20}$$

Ignoring the first and last term, one can minimize the euclidian distance by maximizing $\boldsymbol{x}\boldsymbol{y}^T$.

$$\min \|\boldsymbol{x} - \boldsymbol{y}\| = \min -2\boldsymbol{x}\boldsymbol{y}^T \tag{B.21}$$

$$\min \|\boldsymbol{x} - \boldsymbol{y}\| = \max \boldsymbol{x}\boldsymbol{y}^T \tag{B.22}$$

So, minimizing the euclidian distance equals maximizing the cross correlation.

B.3 The Nearest Neighbor Search

The Nearest Neighbor Search (NNS) problem is of significant importance to several topics in computer science including pattern recognition, data mining,

information retrieval and searching in multimedia data. The usage of multimedia databases has of course steadily increased over the years in numerous areas. An important topic in this field is content based retrieval of text, images or video. Content based retrieval systems requiere additional functionality to handle the search for similar objects. Most approaches transform the objects to a feature vector in a high dimensional space. The similarity search is thus a search for similar vectors in a high dimensional space given a query vector. This is an instance of nearest neighbor searching. There is a lot of research devoted to finding optimal algorithms in terms of time and space complexity [26] and designing database structures to that support efficient nearest neighbor queries over multimedia data [5].

Formally the Nearest Neighbor Search deals with the following problem: Given a set P of n points in some d-dimensional space $\Re^d$, a distance metric M and a query point $q \in P$, find the point $p \in P$ that is most closest to the query point q.

The distance metric is usually a ℓ_s norm i.e. the distance between p and q is:

$$\|p - q\|_s \tag{B.23}$$

where:

$$\|x\| = \left(\sum_{i=1}^{d} |x_i|^s \right)^{\frac{1}{s}} \tag{B.24}$$

The most naive solution is an algorithm that computes the distance from all point $p \in P$ to query point q and returns the point with smallest distance. It has a time complexity of $\Theta(dn)$ which is unacceptable for large datasets. Consequently a lot of research has gone into finding algorithms that perform better. Many algorithms, such as KD-trees, are geared to exploiting the fact that points lie in a plane. The nearest neighbor problem can then, under certain constraints, be solved in $O(\log n)$ time per query [49]. Unfortunately as the dimension becomes 'high enough' the time and space constraints become exponential in the dimension. The work of [5] surveys many efficient data structures that use the optimal $O(dn)$ space. Even so, the query times becomes linear for high dimen-

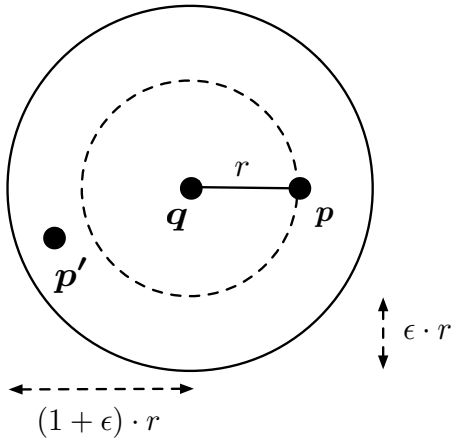


Figure B.3 – Approximate Nearest Neighbor searching.

sional datasets. The failure to rid this exponential dependance on the dimension d had led to research to find out if it is possible to get faster solutions if one doesn't look for the exact nearest neighbor match, but an approximate match.

Formally the ϵ-approximate Nearest Neighbor Search is defined as follows. Given a set P of n points in some d-dimensional space $\Re^d$, a distance metric M and a query point $q \in P$, return each point within at most ϵ times the distance between p and q, where p is the point closest to query point q. See Figure B.3.

Several researches have shown that using approximate nearest neighbors reduces the exponential dependance on the dimension to a polynomial one [23]. The 'curse of high dimensionality' remains however. It is widely known that the distances of a set of uniformly distributed points to a certain query point is nearly equal in high dimensional datasets [4]. This means that for sufficiently high dimensions, the approximate nearest neighbor search degenerates to picking a random sample from the database.

B.4 Matching Image Patches

Image patches are matched and aligned using features that are found in these images. This means that next to defining and localizing the specific feature, the features must also be matched individually. Any feature is comprised of a location and some kind of vectorized descriptor payload. In the most simple case, which we will use here, the descriptor vector is build from the image values around the feature location. This image segment will be referred to as a patch. See Figure B.4.

Patches from the target image $\boldsymbol{I}$ and the distorted image $\boldsymbol{I'}$ are compared using Nearest Neighbor Search (NNS). This means that for any given patch $\boldsymbol{I}$, all the patches from $\boldsymbol{I'}$ are exhaustively compared using the Euclidian distance. The basic assumption is that:

$$\boldsymbol{I}'(x', y') = \boldsymbol{I}(x, y) + n \tag{B.25}$$

Where (x', y') is the distorted position of (x, y) and n is independent gaussian noise. Figure B.5a shows the probability density of the smallest Euclidian (e_{1-NN}) distance between patches from image $\boldsymbol{I}(x, y)$ and $\boldsymbol{I}'(x', y')$ for both correct matches (inliers) and incorrect matches (outliers) for the images shown in Figure C.1. Both distributions overlap completely, making a classification schema based on the e_{1-NN} metric infeasible. Intuitively this can be explained as follows. Although correct matches will exhibit a smaller error on average than incorrect matches, the overall scale of the error, that is feature dependent, varies to much.

As mentioned in Section B.3 and Figure B.5, the database work from [4] shows that any nearest neighbor query is unstable if the distance from the query point to most other data points is less then $(\epsilon + 1)$ times the distance from the query point to its true nearest neighbor. Figures B.5e shows a histogram of the difference between the best match for a patch and all the distances to all other patches. The majority of the distances is smaller than one. Figure B.5f stresses the same point by showing that the majority of points fall with in the $(\epsilon + 1)$ norm for small values of ϵ.

Both [32] and [8] advocate the use of the metric e_{1-NN}/e_{2-NN}: the ratio of the closest Euclidian match by the second best match. This approach hinges on the thought that e_{1-NN} potentially is the correct match and e_{2-NN} is an incorrect match. Using their 'abbey' dataset they report that the distributions of in- and outliers separate with this metric. Our tests, on the dataset in Figure C.1 and C.2, show some improvement over the e_{1-NN} metric. Notable is also that the average e_{2-NN} distance remains about constant for a dataset with similar images. For the 10ATM image set, Figure C.2, the average e_{2-NN} distance was 21.9 with a standard deviation of 4.8. As mentioned in Section B.3, this phenomenon is known as the shell property. The average e_{2-NN} distance can thus be used as an early threshold to easily discard outliers. Discarding outliers early in the processing chain greatly reduces the burden on further algorithms such as RANSAC.

The best results are given by the best-match (bm) algorithm. This algorithm was proposed by [32] and dictates that the best match e_{1-NN} for any point must be at least a certain percentage better than the second best match e_{2-NN}. With a percentage threshold of 60 percent, this algorithm performs nearly flawless. Table B.1 shows the probability of error, P_e, for all metrics. Intuitively this algorithm works by dictating that matching points must be each others best match unambiguously. The only downside to the best-match algorithm is the fact that it can suffer from a significant number of false rejects leaving very little points to process further.

B.5 Hough Pose-space

Given the matching SIFT point pairs:

$$\begin{matrix}[x_{object}\ y_{object}\ \sigma_{object}\ \theta_{object}] \\ [x_{image}\ y_{image}\ \sigma_{image}\ \theta_{image}]\end{matrix} \tag{B.26}$$

The 4D dimensional Hough pose space $\{h_x, h_y, h_\sigma, h_\theta\}$ is derived as follows.

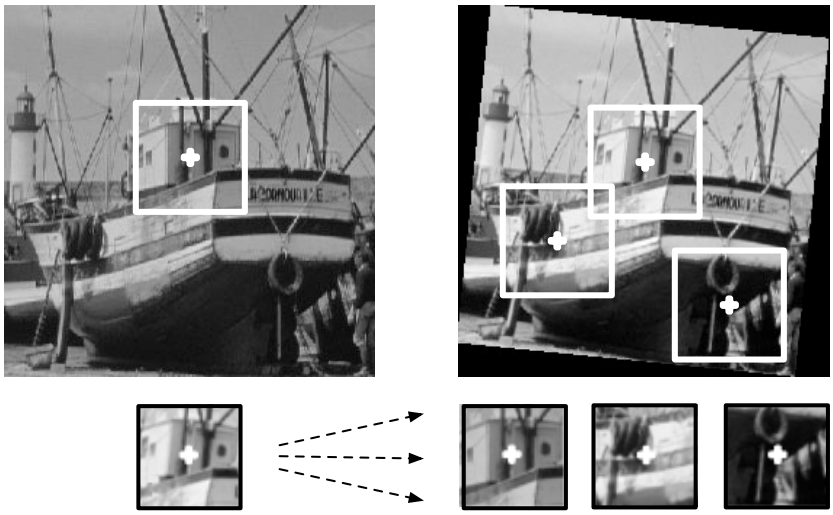


Figure B.4 – Exhaustively matching the image patches from features between an image and its distorted counterpart.

The prediction of the location in pose-space is governed by rotation matrix $\mathbf{R}$

$$\mathbf{R} = \begin{pmatrix} cos(\theta) & -sin(\theta) \\ -sin(\theta) & cos(\theta) \end{pmatrix} \tag{B.27}$$

The first two pose-space parameters h_x and h_y are then calculated:

$$\begin{pmatrix} h_x \\ h_y \end{pmatrix} = \mathbf{R} \begin{pmatrix} x_{obj} \\ y_{obj} \end{pmatrix} - \mathbf{R} \begin{pmatrix} x_{sc} \\ y_{sc} \end{pmatrix} \tag{B.28}$$

The scale parameter h_σ:

$$h_\sigma = \frac{\sigma_{obj}}{\sigma_{sc}} \tag{B.29}$$

And finally the orientation parameter h_θ

$$h_\theta = \theta_{obj} - \theta_{sc} \tag{B.30}$$

B.6 Direct Linear Transform

Given a reference image and its distorted counterpart, the idea is to use $n \geq 4$) 2D point correspondences $\{\boldsymbol{x}_i \leftrightarrow \boldsymbol{x}_i'\}$ to determine the 2D homography matrix H such that $x_i = Hx_i$. The algorithm from [21] to achieve this is as follows:

1. Normalization of $\boldsymbol{x}$ (top view image points)
2. Normalization of $\boldsymbol{x}'$ (camera view image points)
3. Apply Direct Linear Transformation (DLT) to retrieve H
4. De-normalization

Here two distinct steps are identified, the normalization of the chosen points in the image and model and the derivation of the matrix H by applying the DLT algorithm, we will elaborate on these steps.

Normalization

The first essential step of the algorithm is normalization. It consists of scaling and transforming the image coordinates. Apart from improving the results, it also ensures that the results from the algorithm are invariant with respect to the scale and chosen origin of the dataset. The normalization is done separately for the found matches in both images and consists of the following steps:

1. The points are translated so that their centroid lies at the origin.
2. The points are scaled so that the average distance from the origin is $\sqrt{(}2)$.
3. This transformation is applied to the coordinate vector.

This can be done as follows. Given a vector $\boldsymbol{x}$ with 2D inhomogeneous coordinates, where μ_x and σ_x are calculated. Then:

$$\boldsymbol{x} = \boldsymbol{x} - \mu_x \quad f = \frac{\sqrt{(}2)}{\sigma_x} \tag{B.31}$$

The transformation matrix $\mathbf{T}$ is then defined as:

$$\mathbf{T} = \begin{pmatrix} f_x & 0 \\ 0 & f_y \end{pmatrix} \tag{B.32}$$

After which the vector $\boldsymbol{x}$ is transformed by $\boldsymbol{x}' = \boldsymbol{x} * \mathbf{T}$. This normalization is done for both images, the resulting points can be used by the Direct Linear Transformation algorithm to calculate the homography matrix H that we eventually need to do the projection.

Direct Linear Transformation algorithm

The Direct Linear Transformation (DLT) algorithm from [21] can be used to compute the homography matrix H which is needed to perform a projection of the camera view to the top view. The DLT algorithm uses the normalized corresponding points of the two images (which are chosen by hand). The basic DLT algorithm is as follows:

1. Given $n \geq 4$ $2D$ to $2D$ (normalized) correspondances $\{\boldsymbol{x}_i \leftrightarrow \acute{\boldsymbol{x}}_i\}$, form a 2×9 matrix $\boldsymbol{A}_i$:

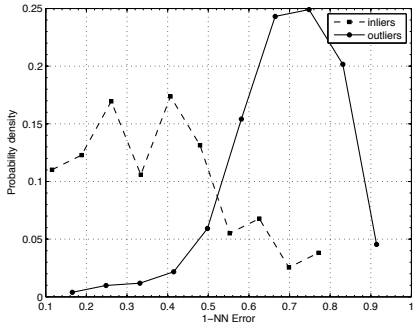
$$\boldsymbol{A}_i = \begin{bmatrix} \mathbf{0}_T & -w_i^{'}\boldsymbol{x}_i^T & y_i^{'}\boldsymbol{x}_i^T \\ w_i^{'}\boldsymbol{x}_i^T & \mathbf{0}_T & -x_i^{'}\boldsymbol{x}_i^T \end{bmatrix} \begin{pmatrix} \boldsymbol{h}^1 \\ \boldsymbol{h}^2 \\ \boldsymbol{h}^3 \end{pmatrix} = 0 \tag{B.33}$$

2. Stack all n matrices $\boldsymbol{A}_i$ to form a single big $2n \times 9$ matrix $\boldsymbol{A}_i$.

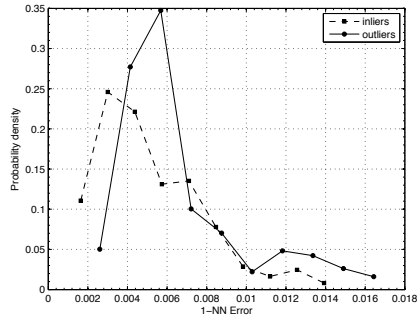
3. Calculate the Single Value Decomposition (SVD) of matrix $\boldsymbol{A}_i$, the unit singular vector corresponding to the smallest singular value is the solution $\mathbf{h}$.

4. The homography matrix H can be derived from $\mathbf{h}$, since $\mathbf{h}$ is a 9-vector made up of entries of the matrix H.

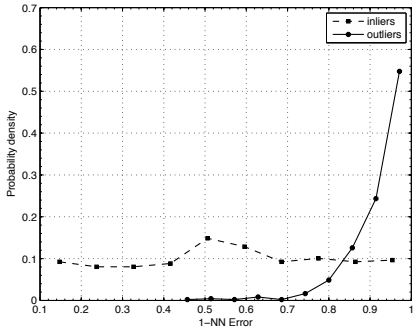
The resulting matrix H now has to be de-normalized to get the homography matrix H that we can use to perform the desired projection. De-normalization is done by setting $H = T'^{-1}T$.



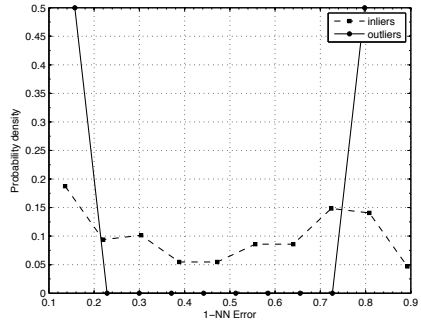
(a) Probability density functions of the e_{1-NN} error.



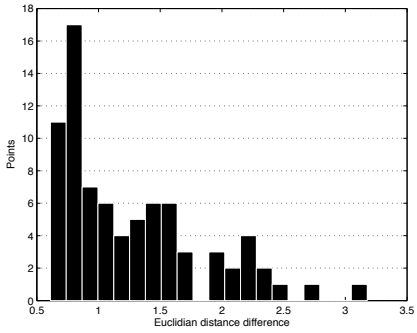
(b) Probability density functions of the $e_{1-NN}/\mu(e_{2-NN})$ error.



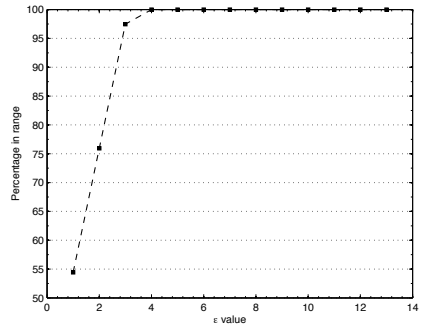
(c) Probability density functions of the e_{1-NN}/e_{2-NN} error.



(d) Probability density functions of the e_{1-NN}/e_{2-NN} error.

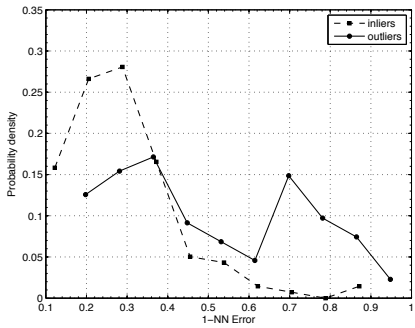


(e) The factor with which the euclidian distances vary between the nearest neighbor of a point, and all other points.

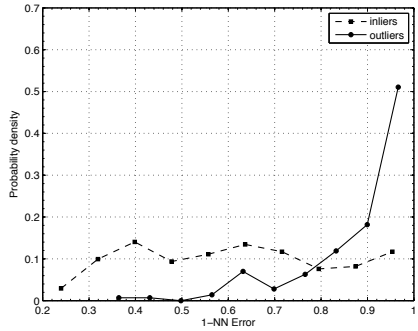


(f) The percentage of data points that fall within a certain ϵ norm.

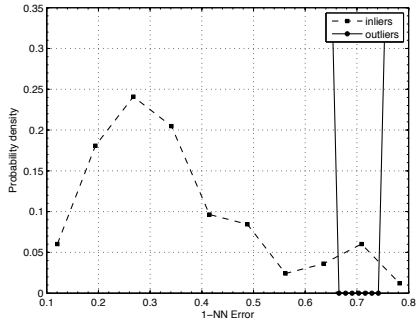
Figure B.5 – Using various Euclidian distance metrics on the image pair of Figure C.1a and C.1b to match patch regions around found features.



(a) Probability density functions of the e_{1-NN} error.



(b) Probability density functions of the $\frac{e_{1-NN}}{\mu(e_{2-NN})}$ error.



(c) Probability density functions of the $bm(60)$ error.

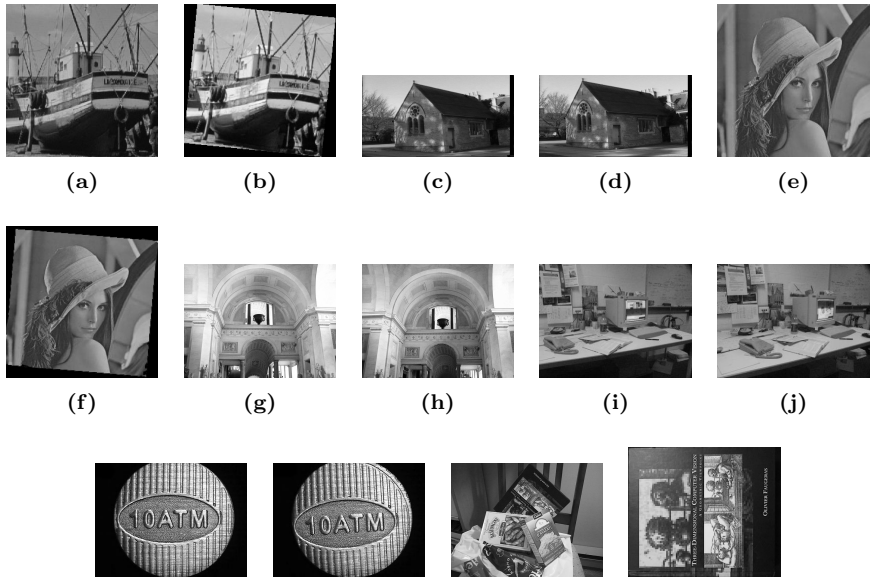
Figure B.6 – Using various Euclidian distance metrics on the 10ATM image set (Figure C.2). It has a total of 250 manually annotated matches.

Image		e_{1-NN}	$\frac{e_{1-NN}}{\mu(e_{2-NN})}$	$\frac{e_{1-NN}}{e_{2-NN}}$	$bm(t=60)$
boat	(C.1a)	0.47	0.42	0.42	0
chappel	(C.1c)	0.35	0.35	0.35	0
lena	(C.1e)	0.30	0.28	0.28	0.03
sixtine	(C.1g)	0.88	0.85	0.85	0.28
office	(C.1i)	0.54	0.58	0.53	0
10ATM	(C.1k)	0.56	0.45	0.45	0.035
scene	(C.1m)	0.96	0.96	0.96	0.17

Table B.1 – The probability of error, p_e, for the image set in Figure C.1 and all Euclidian Nearest Neighbor methods.

Appendix C

Image sets

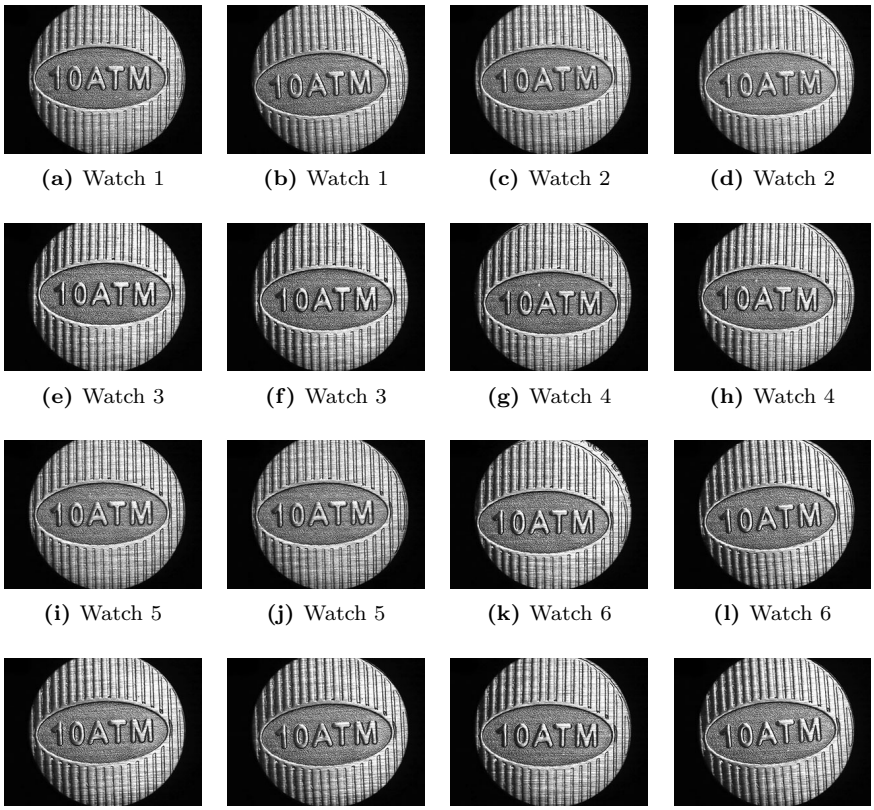


(a) **(b)** **(c)** **(d)** **(e)**

(f) **(g)** **(h)** **(i)** **(j)**

(k) **(l)** **(m)** **(n)**

Figure C.1 – Nearest neighbor test set.



(a) Watch 1 **(b)** Watch 1 **(c)** Watch 2 **(d)** Watch 2

(e) Watch 3 **(f)** Watch 3 **(g)** Watch 4 **(h)** Watch 4

(i) Watch 5 **(j)** Watch 5 **(k)** Watch 6 **(l)** Watch 6

(m) Watch 7 **(n)** Watch 7 **(o)** Watch 8 **(p)** Watch 8

Figure C.2 – 10 ATM Image test set.

Dataset 1	
Origin	Security token
Material	Coated paper
Device	Camera A microscope
Quantity	3*12
Dataset 2	
Origin	Security token
Material	Coated paper
Device	Camera A microscope
Quantity	3*12
Dataset 3	
Origin	Rubber stamp
Material	Coated paper
Device	Camera A microscope Industrial cam scanner
Quantity	3*100
Dataset 4	
Origin	Security token
Material	Paper
Device	Camera A microscope
Quantity	120

Table C.1 – Datasets

Dataset 4	
Origin	Security token
Material	Paper
Device	Industrial cam
Quantity	72
Dataset 5	
Origin	Security token
Material	Paper
Device	Camera A microscope Industrial cam
Quantity	40
Dataset 6	
Origin	Security token
Material	Paper
Device	Camera A microscope
Quantity	120
Dataset 6	
Origin	Label
Material	Paper
Device	Camera A
Quantity	72

Table C.2 – Datasets

Dataset 7	
Origin	Security token
Material	Paper
Device	Camera A microscope
Quantity	40
10ATM	
Origin	Watch
Material	Steel
Device	Camera A microscope
Quantity	20
Alpha	
Origin	Watch
Material	Brushed steel
Device	Camera A
Quantity	20
Dataset 11	
Origin	Part
Material	Dataset 11 plate
Device	Camera B microscope
Quantity	10*10

Table C.3 – Datasets

Tree	
Origin	Label
Material	Paper
Device	Camera A microscope
Quantity	66
Blue letters	
Origin	Label
Material	Paper
Device	Camera A microscope
Quantity	66
Dataset 14	
Origin	Package
Material	Paper
Device	Camera A
Quantity	6*96
Dataset 14	
Origin	Label
Material	Paper
Device	Camera B microscope Type A LED ring lighting
Quantity	3*96

Table C.4 – Datasets

Dataset 14	
Origin	Label
Material	Paper
Device	Camera B microscope Difussed ringlight
Quantity	6*96
Dataset 14	
Origin	Label
Material	Paper
Device	Camera C
Quantity	2*12
Dataset 15	
Origin	Label
Material	Paper
Device	Camera B
Quantity	14*3
Dataset 16	
Origin	Label
Material	Paper
Device	Camera B microscope Type A LED ring lighting
Quantity	50*3

Table C.5 – Datasets