ABSTRACT

LIU, RUNZE. Micro Signal Extraction for Surface-based Authentication and Deepfake Detection. (Under the direction of Chau-Wai Wong).

With the fast development of computer vision technology and the mobile phone industry, it becomes desirable to use camera-captured images for forensic applications, such as anti-counterfeiting of paper-based documents and even integrated circuits (IC) chips. On the other hand, the development of deepfake technology has raised public concern. This dissertation will discuss methods for paper/IC authentication and deepfake detection.

The first part of this dissertation introduces authentication systems based on physical unclonable function (PUF). When using the mobile camera as capturing device for paper PUF, we develop two enhanced geometric reflection models that take into account the ambient light for norm map estimation. Also, new physical features, the high spatial frequency subbands, are proposed as discriminative features because they can better characterize the microstructures of paper surfaces. We also use flatbed scanners, which have a well-controlled experimental environment, to investigate key research questions in the paper surface-based authentication system. This dissertation studies the effect of specular reflection, paper patch size, blurring effect, and different physical features in the setup of flatbed scanners. The understanding from the results using scanner helps the design of paper surface-based authentication system in the real-world scenario. Like the paper surface, the microstructure of IC chip surfaces is also random. Therefore, the uniqueness of the surface of IC chips can be used for IC authentication. To the best of our knowledge, optical PUFs have not been used for IC chip authentication. We investigate into using camera/flatbed scanners to capture images of IC chip surfaces, and then obtain physical features of the chip surfaces for authentication. To make it more applicable in the real-world scenario, we also build fast verification schemes for IC chip authentication using a camera to capture videos of the chip surfaces.

The second part of this dissertation studies and exploits the traces left by neural networks for deepfake detection. We first investigate an intuitive but understudied characteristic, "nonsmoothness", caused by max pooling and ReLU activation in the neural networks. We mathematically define the nonsmoothness and use synthetic data to confirm its existence. This dissertation also studies the statistical properties of nonsmoothness and models the events of nonsmoothness. The nonsmoothness can be regarded as a trace left by neural networks, which may be exploited for deepfake detection since high-quality deepfake videos

are generated by neural networks. We propose the double operation method to exploit the traces left by neural network for deepfake detection. For a deepfake video already with traces left by neural network, the reconstructed version of the deepfake video also has traces. For a raw video without traces, the reconstructed version of it will contain traces left by neural network. Therefore, the videos before and after the reconstruction can be compared for deepfake detection. Siamese neural network has been used to build a classifier, and the detection results outperform state-of-the-art methods based on end-to-end CNN structures.

Micro Signal Extraction for Surface-based Authentication and Deepfake Detection

by
Runze Liu

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Electrical Engineering

Raleigh, North Carolina
2022

APPROVED BY:

Tianfu Wu	Edgar Lobaton
Huaiyu Dai	Sujit Ghosh

Chau-Wai Wong
Chair of Advisory Committee

DEDICATION

To my parents.

BIOGRAPHY

The author received his B.S. degree from the Department of Electronic Engineering, Tsinghua University in Beijing in 2015. In Spring 2016, he started Ph.D. study in the field of nanotechnology in the Department of Electrical and Computer Engineering, NC State University. Later, he changed the field of study to signal processing, and joined Dr. Wong's group in 2018. His research interests include image processing, machine learning, and multimedia forensics.

ACKNOWLEDGMENTS

I would like to express my gratitude for the help and support I received in my Ph.D. study. This dissertation would not have been possible without the help and support from many people. First of all, I would like to thank my advisor, Dr. Chau-Wai Wong, for his advice, support, and encouragement through my Ph.D. study. He is a great advisor and passionate researcher, and I have learned a lot from him.

I would like to thank my dissertation committee, Dr. Tianfu Wu, Dr. Edgar Lobaton, Dr. Huaiyu Dai, and Dr. Sujit Ghosh for their valuable feedback and support for my research. I also want to thank all the faculty and staff who have provided help for my Ph.D. study.

Finally, I would like to express my gratitude for the help and support from my labmates: Jisoo Choi, Joe Zhou, Jiawei Gao, and Kai Yue.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER

1

INTRODUCTION

With the fast development of computer vision technologies and the mobile phone industry, mobile camera-based applications have become more applicable in the real world. On the other hand, the rapid development of video synthesizing technology results in better quality deepfake videos, making deepfake detection an important task. This dissertation discusses methods to extract micro signals for different forensic applications, including surface-based authentication and deepfake detection. In Chapters 2 and 3, microstructures of paper surfaces are estimated via images captured by a camera/scanner for paper authentication. In Chapter 4, we propose to estimate microstructures of IC chip surfaces for IC authentication. In Chapter 5, we study a nonsmoothness feature in modern neural networks, which can be regarded as a processing trace left by neural networks. In Chapter 6, we propose a double operation method to exploit traces left by neural networks for deepfake detection.

1.1 Surface-Based Authentication Exploiting Microstructures

This dissertation discusses micro signal extraction for surface-based authentication, including paper surfaces and IC chip surfaces. Mobile cameras and flatbed scanners have been used to capture photos of the surfaces, which will be used to derive the microstructures of the surfaces for authentication. Paper surfaces under the microscopic view are observed to be formed by intertwisted wood fibers. Such structures of paper surfaces are unique from one location to another and are almost impossible to duplicate. Previous work has shown that cameras are capable of capturing such intrinsic roughness in term of surface normal vectors for security and forensics applications. The mobile cameras are flexible and user-friendly capturing devices, making them more favorable in real-world applications.

In Chapter 2, we examine several candidate mathematical models for camera captured images of paper surfaces and compare the modeling accuracies with reference to the measurement by the confocal microscopy. Experimental results show that the model with distinct intensity bias for images captured from different viewpoints can provide the closest result to the confocal measurement. We discover that high-frequency subbands of reconstructed 3D surfaces are more powerful than the norm map in describing the uniqueness of a physical surface. We show through a practical paper surface based authentication system that incorporating these findings can improve the discrimination performance. In future work, performance of the authentication system in a real-world scenario will be investigated.

In Chapter 3, we examine several key research questions of feature extraction in both scientific and engineering aspects to facilitate the deployment of paper surface-based authentication when flatbed scanners are used as the acquisition device. We analytically show that, under the unique optical setup of flatbed scanners, the specular reflection does not play a role in norm map estimation. We verify using a larger dataset than prior work that the scanner-acquired norm maps, although blurred, are consistent with those measured by confocal microscopes. We confirm that when choosing an authentication feature, high spatial-frequency subbands of the heightmap are more powerful than the norm map. Finally, we show that it is possible to empirically calculate the physical dimension of the paper patch needed to achieve a certain authentication performance in equal error rate (EER). We analytically show that log(EER) is decreasing linearly in the edge length of a paper patch.

Besides authentication for paper, the surface-based authentication can also be used for other objects such as integrated circuits (ICs). The semiconductors industry has been developing fast in the past few decades and electronic devices have become increasingly

common. Counterfeiting for the ICs has become a major challenge. The use of counterfeit ICs has raised threats to multiple sectors that use electronic systems, such as public health, banking, and military defense industries. In Chapter 4 we re-purpose paper surface-based authentication for IC chips since the chip surface is also random like the paper surface. We derive physical features of chip surfaces with images captured by a camera for chip authentication. Also, we develop fast authentication schemes to authenticate IC chips with camera-captured videos.

1.2 Exploiting Processing Traces Left by Neural Networks for Deepfake Detection

Modern neural networks have been successful in many regression-based tasks such as face recognition, facial landmark detection, and image generation. In Chapter 5, we investigate an intuitive but understudied characteristic of modern neural networks, namely, the nonsmoothness. The experiments using synthetic data confirm that such operations as ReLU and max pooling in modern neural networks lead to nonsmoothness. We quantify the nonsmoothness using a feature named the sum of the magnitude of peaks (SMP) and model the input–output relationships for building blocks of modern neural networks. Experimental results confirm that our model can accurately predict the statistical behaviors of the nonsmoothness as it propagates through such building blocks as the convolutional layer, the ReLU activation, and the max pooling layer.

The processing traces left by neural networks, such as nonsmoothness, can be exploited for deepfake detection since most deepfake videos are generated using neural network-based tools, which will leave processing traces in the synthetic videos. Deepfake technology has developed fast in recent years and public figures are usually the victims. Detecting deepfake videos, especially for public figures, has become an important task in digital journalism. In Chapter 6, we target the deepfake detection problem for an individual public figure. In this work, we detect deepfake videos using the method of double operations, i.e., we reconstruct a questionable video and exploit the traces before and after the reconstruction. We learn representations of videos before and after reconstruction with a Siamese neural network. The proposed method outperforms the off-the-shelf pretrained and fine-tuned deepfake detector.

CHAPTER

2

ENHANCED GEOMETRIC REFLECTION MODELS FOR PAPER SURFACE BASED AUTHENTICATION

2.1 Introduction

Paper surfaces under the microscopic view has fuzzy random appearance caused by inter-twisted wood fibers [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]. Such randomness is intrinsic and physically unclonable, which makes the paper surface an ideal candidate for anti-counterfeiting applications. Using the microscopic roughness of surfaces, important documents, concert tickets, and packages can be uniquely identified without relying on other extrinsic means.

Two major approaches have been taken to exploit the paper surface for authentication. The first approach [3, 4, 5, 12, 13] directly uses the camera/scanner captured image or derived features of the image to characterize the paper. The second approach [6, 10, 11] characterizes the paper using the normal vector field that quantifies the microscopic roughness estimated using several images of the patch. The physical laws of light reflection is

employed to derived the normal vectors at the surface. In this paper, we focus on the second approach that uses the normal vector field as the matching feature. Previous work used scanners [6, 10] and mobile cameras [11] to capture the reflected light and estimate the normal vector fields for authentication.

It is desirable to have a deeper understanding in the models and algorithms for estimating such physical feature using mobile devices because of their widespread use. The estimated normal vector field from the camera-captured images can achieve satisfactory authentication results [11]. However, the rudimentary model resulted in coarse normal vector estimates when verified against the confocal microscopy results [11] . In addition, the understanding of the normal vector field's key information toward satisfactory authentication performance is lacking.

In this paper, we propose new mathematical models that takes into account both the light reflection and the image acquisition processes to estimate the normal vector field. We reconstruct 3D microscopic paper surfaces from normal vector fields and decompose the surfaces into different spatial-frequency subbands. The surfaces and their frequency subbands estimated from different models are verified against those measured from the confocal microscopy. The proposed models and new features will be tested in a practical authentication system to measure the performance gain.

The rest of this chapter is organized as follows. In Section 5.2, we review the fundamentals on light reflection and tools that will be used. In Section 2.3, we propose new mathematical models. In Section 2.4, we examine the discrimination performance of the proposed models at different subbands. In Section 2.5, we evaluate the performance gain in a practical engineering system when our findings are incorporated. In Section 5.8, we conclude the chapter and discuss future work.

2.2 Background and Preliminaries

2.2.1 Diffuse Reflection Model and Baseline Method for Norm Vector Estimation

Under the fully diffuse model, the perceived intensity l_r at location p is modeled as follows [11, 14, 6]:

$$l_r(p) = \lambda \cdot l(p) \cdot n(p)^T v(p), \tag{2.1}$$

where $n = (n_x, n_y, n_z)$ is the normal direction of the paper surface at the microscopic level, $v(p)$ is the incident light direction arriving at p, $l(p)$ is the strength of the light and λ characterizes the physical capability of reflecting the light.

The authors of [11] decomposed the perceived intensity of a paper patch under a point source into two components: i) a smooth component, named the macroscopic intensity, with a mild spatial change mainly due to the varying distance between each pixel and the point source, and ii) a highly fluctuating component due to the inconsistent orientation of the paper surface at the microscopic level. In order to estimate the microscopic paper surface that leads to the fluctuating component, the authors approached the problem by compensating the effect of the macroscopic intensity, and then reformulating the problem into a linear regression problem.

It was shown that paper's macroscopic intensity $\tilde{y}(p)$ is spatially dominated by $l(p)$, namely,

$$\tilde{y}(p) \approx \mathbb{E}[l_r(p)] = \lambda \cdot l(p) \cdot m_z \cdot v_z(p) \tag{2.2}$$

where m_z is mean of the z component of the normal vectors, $v_z(p)$ is the z component of the incident light direction vector and its value is usually slightly less than 1. With the equality established in Eq. (2.2), the term containing the arriving light intensity $\lambda l(p)$ may be estimated by:

$$\widehat{\lambda l(p)} = \tilde{y}(p)/(m_z \cdot v_z(p)). \tag{2.3}$$

Substituting Eq. (2.3) into Eq. (4.1), the effect of the smooth component can then be compensated and the remaining knowns and estimated values are now in a linear form of the three coordinates of normal vector $n(p)$, namely,

$$\zeta(p) \approx n(p)^T v(p) \tag{2.4}$$

where $\zeta(p) \triangleq l_r(p)/\widehat{\lambda l(p)}$. To solve for the three unknowns in $n(p)$ and to account for some bias for a particular location p, at least four equations, which correspond to four camera-captured photos of the paper surface, are needed. Normal vectors for different locations are calculated separately and there is no spatial smoothness assumption imposed. We refer to this baseline model as Model 0.

2.2.2 Measure of Discrimination Performance

We use the hypothesis testing framework [15, 11] to evaluate discrimination performance of a system. The null hypothesis H_0 is that the test paper surface does not match the reference surface, and the alternative hypothesis H_1 is that the test surface matches the reference surface. The Pearson's correlation between the test and the reference surfaces quantifying the degree of match is used as a test statistic. We repeatedly collect the correlation values under H_0 and H_1, and estimate the probability density functions by calculating the histograms for H_0 and H_1. Thresholding is applied to calculate the probability of false alarm, P_F, and the probability of miss detection, P_M. The discrimination capability is measured by the receiver operating characteristic (ROC) curve and more compactly, by the equal error rate (EER).

When designing a practical engineering system such as an authentication system, we can decide whether the paper surface captured during a test session matches its record in a reference database using the thresholding rule. We calculate the correlation, and compare it against a predefined threshold τ that controls the tradeoff between P_F and P_M. The surface being examined is only considered authentic when the correlation is larger than τ.

2.3 Proposed Models for Estimating Normals

In this paper, we explicitly model factors that may lead to the bias in intensity such as the ambient illumination and cameras' internal brightness/contrast adjustment processes. We propose models with intercepts in addition to the diffuse reflection component for estimating the normal vector field.

We first propose a model with distinct intercept for each image. For image k, $k = 1, \ldots, M$, the intensity of acquired image at pixel location p is modeled as:

$$y^{(k)}(p) = \lambda l^{(k)}(p) n(p)^T v(p) + \beta_0^{(k)}(p). \tag{2.5}$$

We name this model with distinct intercept over k and p Model 1. To estimate the normal vector $n(p)$, we follow a similar procedure as in Section 2.2.1: We first compensate the smooth component in terms of $\lambda l^{(k)}(p) m_z$ and $\beta_0^{(k)}(p)$ and then reformulate the problem into a linear regression problem. By taking an expectation over the randomness of the normal vector, we can obtain a macroscopic intensity field that contain the global factors, i.e., the strength of the arriving light, the overall intensity bias, and the effect of the camera

direction:

$$\tilde{y}^{(k)}(p) \approx \mathbb{E}\big[y^{(k)}(p)\big] = \lambda l^{(k)}(p) m_z v_z(p) + \beta_0^{(k)}(p), \tag{2.6}$$

where all symbols are similarly defined as in Section 2.2.1. In order to estimate $\lambda l^{(k)}(p) m_z$ and $\beta_0^{(k)}(p)$, we make an explicit assumption that these two terms are roughly constant over a small neighborhood of the spatial grid, e.g., a 4-connected set. When the context is clear and the order of the four neighboring points does not matter, we denote the four points as $p_1, \ldots, p_4$ and denote the center point as p_0. Therefore, by explicitly applying the spatial smoothness constraint, we set up the overdetermined system as follows:

$$\begin{bmatrix} \tilde{y}^{(k)}(p_0) \\ \vdots \\ \tilde{y}^{(k)}(p_4) \end{bmatrix} \approx \begin{bmatrix} v_z^{(k)}(p_0) & 1 \\ \vdots & \vdots \\ v_z^{(k)}(p_4) & 1 \end{bmatrix} \begin{bmatrix} \lambda l^{(k)}(p_0) m_z \\ \beta_0^{(k)}(p_0) \end{bmatrix}. \tag{2.7}$$

Solving this linear system using least-squares gives us the estimates for $\lambda l^{(k)}(p) m_z$ and $\beta_0^{(k)}(p)$. Substituting the estimates into Eq. (4.2), we may setup another system of linear equations using M images:

$$\begin{bmatrix} \tilde{y}^{(1)}(p) \\ \vdots \\ \tilde{y}^{(M)}(p) \end{bmatrix} - \begin{bmatrix} \widehat{\beta_0^{(1)}(p)} \\ \vdots \\ \widehat{\beta_0^{(M)}(p)} \end{bmatrix} \approx \begin{bmatrix} \widehat{\lambda l^{(1)}(p)} v^{(1)T}(p) \\ \vdots \\ \widehat{\lambda l^{(M)}(p)} v^{(M)T}(p) \end{bmatrix} \begin{bmatrix} n_x(p) \\ n_y(p) \\ n_z(p) \end{bmatrix} \tag{2.8}$$

in which the normal vector at location p can be obtained by least-squares.

We propose a more stringent Model 2 by assuming that the bias in intensity such as the ambient illumination and cameras' internal brightness/contrast adjustment processes has the same effect for all images, namely, $\beta_0^{(1)}(p) = \ldots = \beta_0^{(M)}(p)$, so that the model becomes:

$$y^{(k)}(p) = \lambda l^{(k)}(p) n(p)^T v(p) + \beta_0(p). \tag{2.9}$$

In this scenario, $\beta_0(p)$ can be jointly estimated using all M images with the linear system

shown as follows:

$$\begin{bmatrix} \tilde{y}^{(1)}(p_0) \\ \vdots \\ \tilde{y}^{(1)}(p_4) \\ \vdots \\ \tilde{y}^{(M)}(p_0) \\ \vdots \\ \tilde{y}^{(M)}(p_4) \end{bmatrix} \approx \begin{bmatrix} v_z^{(1)}(p_0) & 0 & \dots & 0 & 1 \\ \vdots & \vdots & & \vdots & \vdots \\ v_z^{(1)}(p_4) & 0 & \dots & 0 & 1 \\ \vdots & & & & \vdots \\ 0 & \dots & 0 & v_z^{(M)}(p_0) & 1 \\ \vdots & & \vdots & \vdots & \vdots \\ 0 & \dots & 0 & v_z^{(M)}(p_4) & 1 \end{bmatrix} \begin{bmatrix} \lambda I^{(1)}(p_0) m_z \\ \vdots \\ \lambda I^{(M)}(p_0) m_z \\ \beta_0(p_0) \end{bmatrix}. \tag{2.10}$$

Finally, the normal vector at each location p could be estimated following the similar procedure as described for Model 1.

2.4 Experimental Results

2.4.1 Dataset and Experimental Setup

For the analysis conducted in this section, we use a publicly available dataset on which the confocal microscopy related work in Section VII of [11] was conducted. The dataset contains measurements by a confocal microscope, and images of a paper surface acquired by two commodity scanners and a mobile camera. Measurements from the confocal microscope leads to a topographic map with high spatial resolution. We follow the procedure mentioned in Section VII.C of [11] that calculates the surface direction over the squared coverage of each working pixel to generate a 200-by-200 normal vector field. We use it as the physical reference since the confocal measurement is precise. For the scanner acquired images, we estimate norm maps using the improved method mentioned in Section III.A of [11].

The images of the paper surface by mobile camera were acquired when a flashlight is activated under a normal indoor ambient light environment in 20 different camera locations. In order to evaluate the statistical behaviors of a hypothetical, representative large dataset (i.e., the statistical population) using the limited data from the dataset, we apply the idea of bootstrapping [16, 17] to estimate a set of resampled normal vector fields. Specifically, we use five images randomly chosen from the 20 images to estimate one resampled normal vector field, and repeat such random sampling with replacement 100 times to prepare for the data for the subsequent evaluation for models. Five is the second smallest number needed to estimate a normal vector field. Although the choice of using five images will reveal pessimistic results on the discrimination performance, it reasonably mimics application scenarios that allow capturing fewer images and are with limited computational resources.

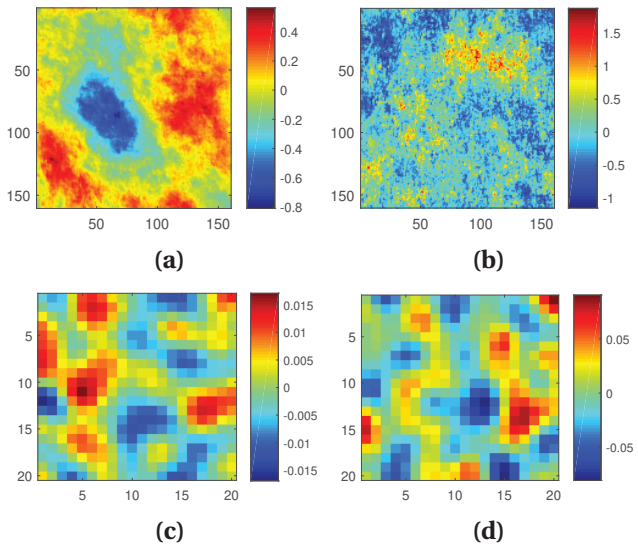


Figure 2.1: Top view of a reconstructed surface by (a) mobile camera with normal vector field estimator derived from Model 1 and (b) confocal microscope. (c) and (d) are SUBBAND #2 of a zoomed-in region of (a) and (b), respectively, exhibiting higher similarity than that between (a) and (b). The colorbar illustrates the relative height of the surface with 84.5 μm as the unit.

Visualizing norm maps or normal vector fields can reveal the surface orientation at each pixel location. For example, when examining a neighborhood of pixels, one can measure the spatial consistency of surface orientations. In comparison, 3D surfaces reconstructed from normal vector fields are more appealing to human eyes and quantitative understandings can be better achieved using off-the-shelf image/surface analysis tools. In this paper, we employ the DoG representation that allows us to separately analyze the discrimination performance at different frequency subbands.

2.4.2 Match Using Full and Subbands of 3D Surface

We use the shapelet method to reconstruct 3D microscopic paper surfaces from normal vector fields. Fig. 2.1(a) and Fig. 2.1(b) show a reconstructed surface by Model 1 and the reference by the confocal microscope, respectively. The two reconstructed 3D surface appears differently at the patch scale: the surface by Model 1 has a large valley in the middle and high peaks around it, whereas the confocal surface is relatively flat at the patch scale and is spatially "busy" with many peaks. The discrepancy in the surface trend may have masked other features that are consistent between the two surfaces.

Next, we the decompose the surface into different spatial-frequency subbands using a DoG representation of 10 levels. High spatial-frequency fluctuations of the microscopic

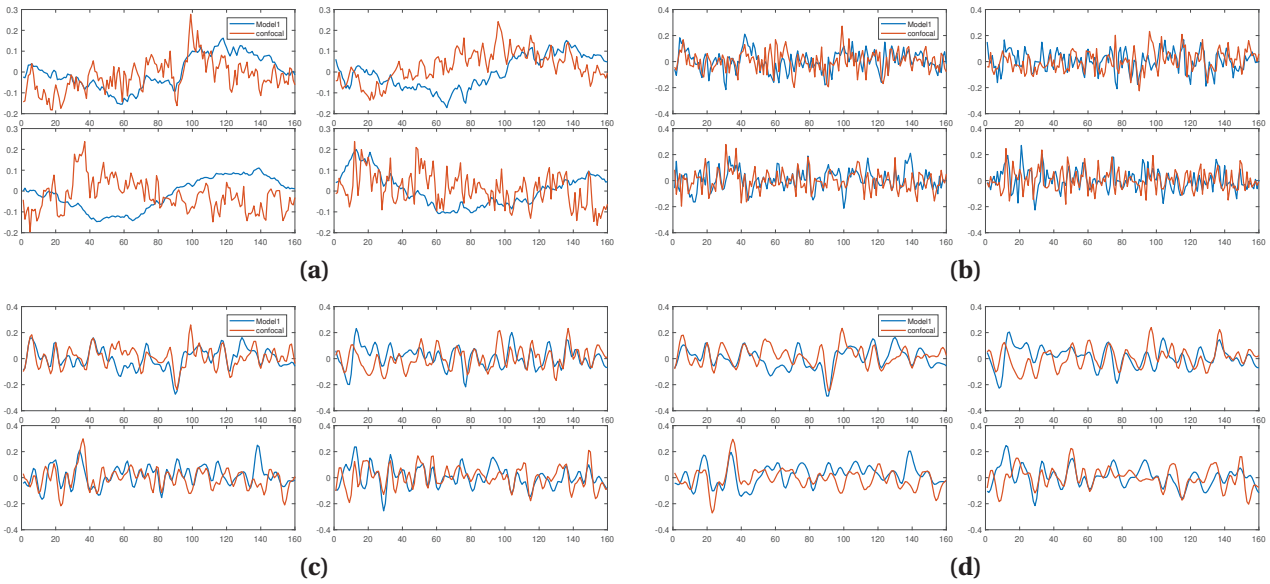


Figure 2.2: Representative slices in x direction from (a) reconstructed surfaces, (b) SUBBAND #1, i.e., the highest frequency subband, (c) SUBBAND #2 and (d) SUBBAND #3. High spatial-frequency subbands have many more overlapped peaks and valleys than the "full spectrum" curves from the original surfaces.

surface are captured in subbands with small index numbers. Every subband of the surfaces reconstructed from the camera estimated and from confocal estimated norm maps is examined.

Fig. 2.1(c) and Fig. 2.1(d) show SUBBAND #2 of a zoomed-in region of the surface by Model 1 and the reference by the confocal microscope, respectively. The two zoomed-in regions are similar that both have small valleys in the middle with peaks around, and there are peaks in the up-right corners.

To illustrate more clearly how well the subbands from camera estimation match those from the confocal measurement, we show in Figs. 2.2(b)–(d) representative slices in the x direction from SUBBANDS #1 to #3. Fig. 2.2 reveals that subbands of high spatial-frequencies have many more overlapped peaks and valleys than the "full spectrum" curves from the original surfaces. This is consistent with what we observed earlier in this subsection.

We then quantitatively examine at each subband the correlation between camera estimation and confocal reference. The bootstrap distributions of the correlation values in matched cases (H_1) and unmatched cases (H_0) are shown in Fig. 2.3. In matched case, the correlation values are relatively large, i.e., around 0.3 at high frequency subbands, and for unmatched cases the correlation values are around 0. The distributions of correlation values for H_0 and H_1 are narrower and farther away from each other at higher frequency

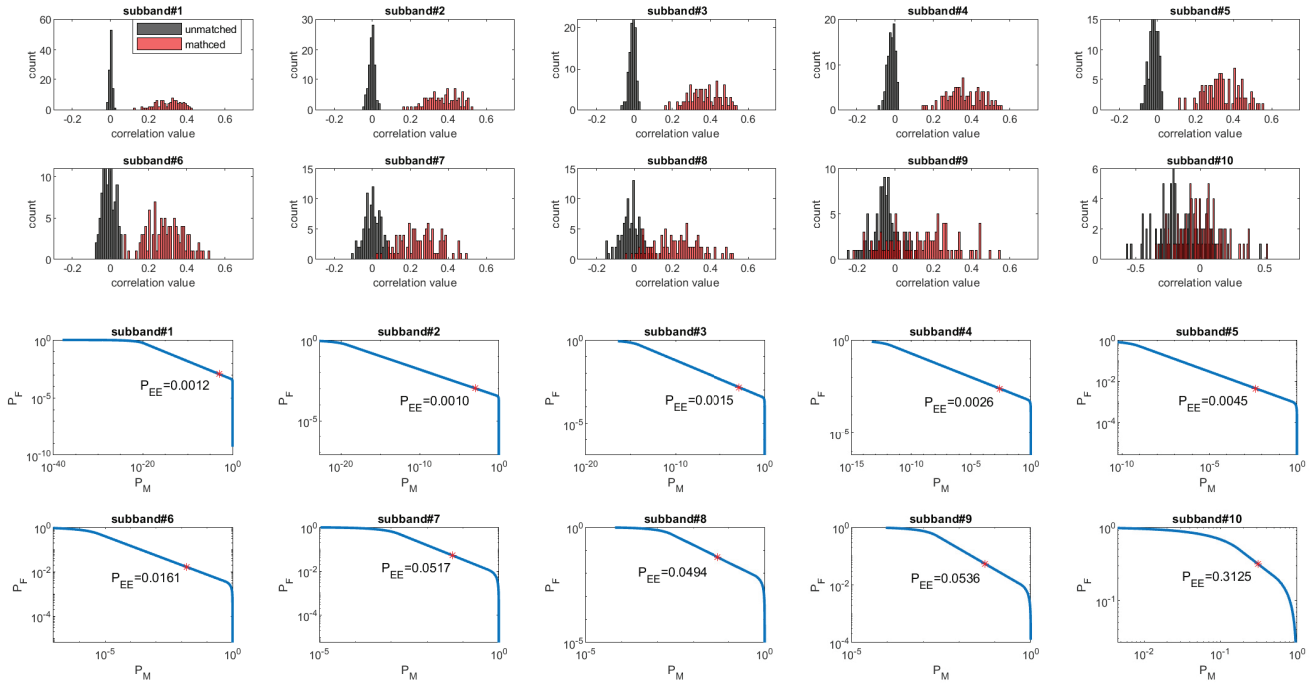


Figure 2.3: Distributions of correlation values for matched cases (H_1) and unmatched cases (H_0) at different subbands for Model 1, and corresponding ROC curves when assuming correlation is Laplacian distributed. Higher spatial-frequency subbands (those with smaller indices) generally are more powerful in describing the uniqueness of physical surfaces.

subbands, suggesting a better discrimination performance, i.e., the capability of describing the uniqueness of physical surfaces. Using the thresholding rule, we plot the ROC curves for all subbands to reveal the discrimination capability under all decision threshold values. EER is annotated on each ROC curve for easy comparison.

Fig. 2.4 shows the EER as a function of the subband index when the correlation values are believed to follow Gaussian and Laplacian distributions, respectively. Both Models 1 and 2 outperform the baseline model at high frequency subbands. As the index of the subband decreases, the performance using Models 1 and 2 increases. Model 1 with EER at 10^{-5} (Gaussian) and 10^{-3} (Laplacian) at the highest frequency subband has the best discrimination capability.

Such observation would naturally trigger a question: Can the best discrimination performance be improved by combining neighboring frequency band(s) with the highest frequency band? We therefore calculate the cumulative subbands of reconstructed surfaces and evaluate the performance. Fig. 2.5 shows the EER values as a function of accumulated subband index when the correlation values are Gaussian and Laplacian, respectively.

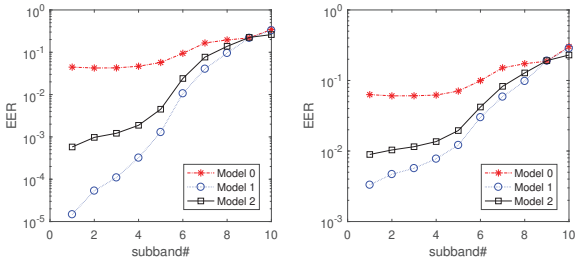


Figure 2.4: Discrimination capability in terms of EER as a function of subband index for (a) Gaussian and (b) Laplacian distributed correlation values. Two proposed models perform significantly better than the baseline model at high-frequency subbands.

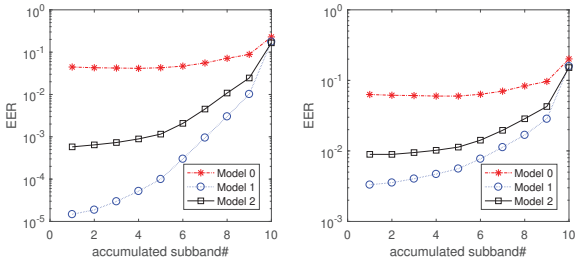


Figure 2.5: Discrimination capability in terms of EER as a function of accumulated subband index for (a) Gaussian and (b) Laplacian distributed correlation values. When viewed with Fig. 2.4, it reveals that combining other subbands to the highest frequency subband cannot improve the best performance.

The plots reveal that the smallest accumulated subband has the best performance, indicating combining other subbands to the highest frequency subband cannot improve the performance.

2.4.3 Match Using Norm Map

For comparison purpose, we also examine the discrimination performance when the norm map is used as the matching feature. When assuming correlation values are Gaussian distributed, EER values are around 10^{-1}, 10^{-4}, and 10^{-3} for Models 0, 1, and 2, respectively. When assuming correlation values are Laplacian distributed, EER values are around 10^{-1}, 10^{-2}, and 10^{-2} for Models 0, 1, and 2, respectively. We observe that using the highest frequency subband as the matching feature performs better than using the norm map by one order of magnitude. This implies that the highest frequency subband is physically more discriminative than the norm map.

2.5 Practical Authentication System

In this section, we examine a practical paper surface based package/label/document authentication system that uses mobile cameras to capture test images and uses scanners to capture the reference. We did not use confocal microscope for capturing the reference because it can be difficult to automate and expensive for commercial applications. In comparison, scanners are easy to automate and inexpensive, and has been shown in [6, 10, 11] to have satisfactory performance when used to capture the reference. Cameras may also be used to capture the reference but may lead to lower performance.

In order to use reconstructed surfaces for authentication, we must have the normal vector field that contains the z-component ready. For precise reference produced by scanners, only x- and y-components are available and these two quantities are scaled. To estimate the z-component, one should properly rescale the x- and y-components and use the relationship $n_z = \left(1 - n_x^2 - n_y^2\right)^{\frac{1}{2}}$ to obtain the estimate. An intuitive approach is to match their probability distributions with those of their counterparts from the normal vector field. In our experiment, we rescale the x-component by the scanner such that its standard deviation becomes the same as that by the confocal microscope. The y-components is scaled similarly.

Section 2.4 reveals that using high frequency subbands as the matching feature with improved models could lead to a better match with the physical measurement. We examine in this section if they can lead to a better performed engineering system. That is, whether they can outperform systems using the norm map directly as the discriminative feature.

Fig. 2.6 shows the EER as a function of the subband index when the correlation values are believed to follow Gaussian and Laplacian distributions. Horizontal lines correspond to the performance when the norm map instead of the reconstructed surface is used as the feature. The plots reveal that the discrimination capability has significantly improved when the practical authentication system uses the improved models and high-frequency subbands of the reconstructed surface as the feature. When assuming correlations are Gaussian distributed, EER values are improved for about one and four orders of magnitude for Models 1 and 2 at SUBBAND #1, respectively. In contrast, no improvement is observed for the baseline model even the highest frequency subband of the reconstructed surface is used as the matching feature. When assuming correlations are Laplacian distributed, EER values are also improved at SUBBAND #1 for the proposed models but not for the baseline model.

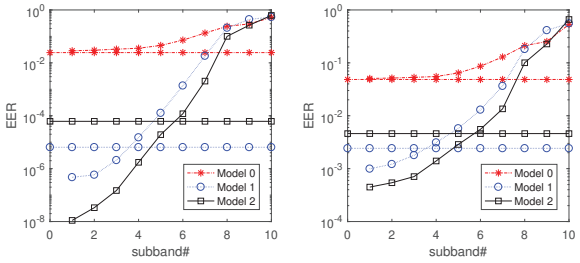


Figure 2.6: Authentication performance in terms of EER as a function of subband index for (a) Gaussian and (b) Laplacian distributed correlation values. Horizontal lines correspond to the performance when the normal vector instead of the reconstructed surface is used as the matching feature.

2.6 Conclusion

In this paper, we have shown that the improved models taking into account the effect of ambient lights and cameras' brightness/contrast adjustment processes can provide better modeling accuracies with reference to the measurement by the confocal microscopy. We have discovered that the high-frequency subbands of the reconstructed surface is a better discriminative feature than the norm map. When such discovery is incorporated into the design of a practical engineering system, it can improve the authentication performance. For future work, we plan to build a large dataset containing confocal measurements, scanner images, and camera images with a variety of paper types, camera models, and acquisition conditions. We also plan to explore the effect of the specular reflection at the paper surface. Such studies could lead to more accurate models for camera captured images and may improve the performance of a practical authentication system.

CHAPTER

3

ON MICROSTRUCTURE ESTIMATION USING FLATBED SCANNERS FOR PAPER SURFACE BASED AUTHENTICATION

3.1 Introduction

When viewed under a microscope, mundane-seeming paper surfaces come to life, and a maze of intertwisted wood fibers creates a complicated random jungle of structure [18, 3, 4, 6, 7, 8, 9, 19, 20, 21, 22, 12, 13]. The unique microscopic structure of the paper surface is physically unclonable and may be considered as a "fingerprint", which can be used for protecting valuable merchandise such as drugs and wines and important documents such as birth certificates and checks. Two categories of methods have been used to capture such unique structure of paper surfaces for authentication, namely, the optical/visual feature approach and the physical feature approach.

The optical/visual approach relies on the visual appearance of the paper surface or handcrafted features derived from the visual appearance for paper identification. Buchanan

et al. [4] used a laser scanner to capture the reflected intensity due to a moving focused line shined on the paper surfaces, and used cross-correlation of digitized intensity fluctuations for identification. As a proof-of-concept effort for paper-based identification, lasers achieved good performance, however, they are expensive and not ubiquitous to be used in practical applications. Beekhof et al. [5] used macrolens-aided mobile phones to capture images of the rough paper surfaces. Minimum reference distance decoding and reference list decoding were used for the identification problem, with a huge reduction in complexity compared to classic minimum distance decoding while maintaining the performance. Sharma et al. [23] used paper speckles, i.e., the dark and bright spots on paper when illuminated by light, as a fingerprint for the paper surface, where images of the paper surface were taken by a camera with aid of a microscope with a built-in LED. The Gabor transform was applied to the captured image, and a binary image was obtained by using the complex phase of the Gabor transform and zero thresholding. The fractional hamming distance was used to compare different binary images. Instead of analyzing the light reflected from the paper surface, Toreini et al. [24] captured optical features of paper texture using the light transmitted through the paper and had satisfying authentication performance. However, it can only be applied in the scenarios that a sheet of paper is not glued to a surface and the paper is relatively transparent. For example, it is difficult to capture the transmissive light for a label stuck to a bottle or for stock paper packaging. The aforementioned methods for identifying paper surfaces are based on the optical/visual features, while their underlying physical features, such as the orientation of a microscopic surface, have been shown to possess greater discriminative power [6, 21].

The orientations of the microscopic surfaces of a paper patch may be quantified by the *norm map*, a collection of uniformly spaced surface normals projected to the xy-plane. Clarkson's et al. [6] proposed a method for estimating a scaled version of the norm map of a paper patch by acquiring the paper in opposite orientations using a flatbed scanner assuming light reflection is fully diffuse. Instead of using a bulky flatbed scanner, Wong et al. [21] used a mobile camera to take multiple photos from different perspectives of a paper patch, estimating the norm map with the diffuse reflection model [14] and the camera geometry [25]. The estimated norm map was also verified by ground truth, a norm map acquired by a confocal microscope. Liu et al. [22] formulated two improved norm map estimators by taking into account the ambient light and cameras' internal brightness and contrast adjustment processes. They also used estimated surface normals to reconstruct heightmaps (3D surfaces) of paper patches and discovered that using the high spatial-frequency components of heightmaps as the authentication feature can achieve better

performance than using the norm map.

Fig. 3.1 demonstrates two potential designs of real-world paper surface-based authentication systems, namely, a client-server model and a local model. The authentication systems, by designating a small paper-based surface area for the purpose of authentication, can be used for protecting merchandise and important documents. For example, a customer can use a mobile phone with an app to obtain the feature of a drug package, and then compare it with the reference feature to verify the authenticity of the packaging. In the client-server model, a mobile phone as the client can acquire images of the paper patch, derive the test feature, and send the test feature to the server using a locally installed app. The server will search in its database whether the test feature matches an existing reference feature upon receiving it from the client. If the reference feature ID is also provided together with the test feature, the server can directly access the reference feature and use it for comparison, which can save the feature retrieval time and increase the authentication accuracy. The authentication result based on the matching outcome will be sent back to the client. In the client-server model, the communication channel between the two parties is protected by cryptographic protocols such as the transport layer security (TLS) to ensure the trustworthiness. In the local model, the encrypted communication is not needed, but an additional QR code is used to store the reference feature protected by the public-key encryption. After decoding the QR code, the user will use the public-key from the vendor to unlock the reference feature. The test feature will be compared with the reference feature to generate the authentication result. Although in this local model the reference feature may be exposed to an untrusted user that tries to tap into the memory to intercept the decrypted reference feature, the attacker still needs to forge a paper patch from which the intercepted feature can be derived, which is impossible because the microstructure is physically unclonable.

To facilitate the deployment of paper surface-based authentication, we examine four keys research questions of feature extraction in both scientific and engineering aspects when flatbed scanners are used as the acquisition device. First, does ignoring the specular reflection have a destructive effect on the authentication performance? Prior approaches for estimating norm maps were based on the assumption that paper reflects the light in a fully diffuse way [6, 19, 20, 21, 22]. In [6], it was argued that the fully diffuse assumption largely holds, but without justifications using experimental results or theoretical derivations. In [21], the strengths of diffuse versus the specular components were estimated to be about six to one, but the specular was not compensated for in the norm map estimation. Since the specular reflection could also be practically observed for paper surfaces even by naked

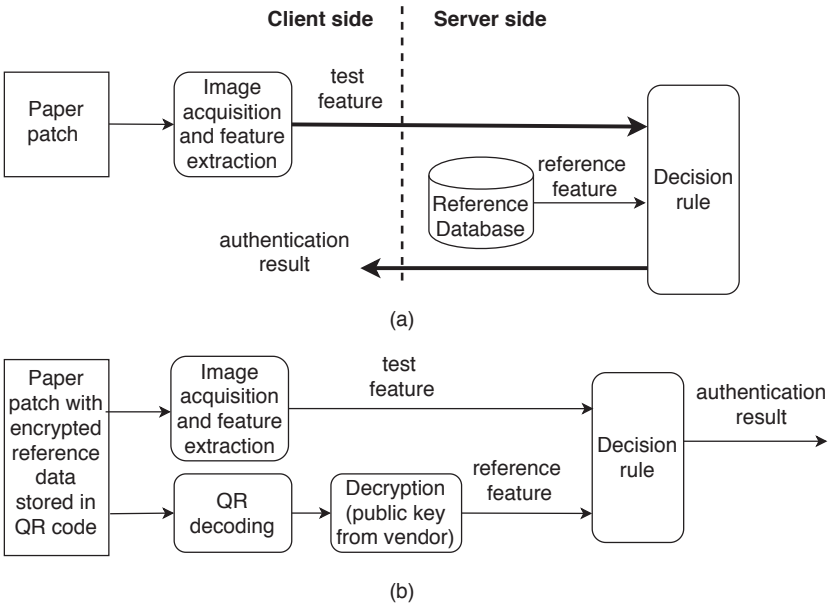


Figure 3.1: Examples of paper surface-based authentication systems: (a) a client-server model, and (b) a local model. The thick arrows are encrypted communication links and the normal arrows are local communication links. The diagrams focus on the verification stage. The reference data are stored in the reference database or the QR code at an earlier enrollment stage.

eyes, it is interesting to investigate whether explicitly taking the specular reflection into the estimator design may improve the accuracy. Second, does the estimated normal vector resemble the real quantity with physical interpretations? Prior work in [21] with a small dataset shows that norm maps acquired by scanners are consistent with those measured by confocal microscopes. In this work, we use a confocal dataset of one order of magnitude larger to obtain a more confident conclusion and extend the inquiry into the scanner's blurring effect. Third, can feature engineering on the estimated normal vectors yield higher authentication performance? The result in [22] demonstrated that the heightmap and its higher-frequency subbands as features outperform norm maps for the mobile cameras. We investigate whether a similar conclusion can be drawn for flatbed scanners. Fourth, we also study how the paper patch size affects authentication performance and investigate the justification for digitizing resolutions for paper patches.

We summarize the contributions of this work compared to previous work [21, 22] in both scientific and engineering aspects. The scientific contributions are as follows:

- We prove mathematically that the effect of the specular reflection can be ignored because of the unique imaging setup of flatbed scanners (but such result is not true for the camera setup);

- We investigate quantitatively the performance drop due to the existence of the blurring effect in the scanner, and use a one order of magnitude larger dataset than that of [21] to confirm that scanners can capture meaningful physical quantities of paper surfaces;

And the engineering contributions are as follows:

- We justify and give a guide to the choices for paper patch size and resolution with mathematical and experimental results, and investigate quantitatively the performance drop due to spatial registration error;
- We confirm that using the heightmap as the feature proposed in [22] is also more discriminative than using the norm map for the flatbed scanner use case.

The rest of the chapter is organized as follows. In Section 5.2, we give some background reviews. In Section 3.3, we analytically investigate the effect of specular reflection in the optical setup of flatbed scanners. In Section 3.4, we investigate the consistency between estimated norm maps from scanners and the confocal microscope with a focus on the blurring effect. In Section 3.5, we examine the performance of physical features such as the heightmap and their subbands. In Section 3.6, we investigate the digitizing resolution and the size of paper patch needed for achieving a certain performance level. Section 5.8 concludes the chapter.

3.2 Background and Preliminaries

Symbol conventions are as follows. Nonitalic bold letters denote vectors and all vectors in this paper are column vectors. For example, $\mathbf{n} = (n_x, n_y, n_z)^T$ defines a column vector with elements n_x, n_y and n_z.

3.2.1 Difference-of-Gaussians (DoG) Representation

In DoG representation [26, 27], the nth level subband is obtained by taking differences of the Gaussian-blurred matrix of numbers as follows:

$$\mathbf{L}_n = \mathbf{G}_n - \mathbf{G}_{n+1},\ n = 1, ..., N \tag{3.1}$$

where $\mathbf{G}_1$ is defined to be the original matrix, $\mathbf{G}_{N+1} = 0$, and $\mathbf{G}_n$, $n = 2, ..., N$, is the result of blurring the original matrix by a Gaussian filter with standard deviation σ^{n-1}, where $\sigma > 1$.

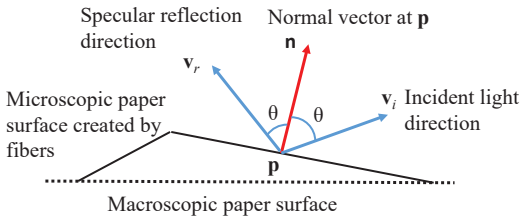


Figure 3.2: A microscopic view of a paper surface with annotated quantities related to light reflection at location **p**. The vectors are all unit vectors.

The DoG representation of a matrix allows us to investigate the different spatial-frequency subbands of the matrix, as shown in Section 3.5.2 and the supplementary document.

3.2.2 Generalized Light Reflection Model

Fig. 3.2 illustrates a microscopic portion of a paper surface containing small surfaces that usually orient differently than the macroscopic paper surface. Picking an arbitrary location $\mathbf{p} \in \mathbb{R}^2$ on the surface and assuming both diffuse and specular reflection types, the perceived intensity l_r for a sensor or an eye at a fixed distance away from $\mathbf{p}$ may be written as the following generalized light reflection model, i.e., the Phong shading model without the ambient light [14]:

$$l_r = \frac{l}{||\mathbf{o}-\mathbf{p}||^2}\left\{ w_d \cdot (\mathbf{n}^T \mathbf{v}_i)^+ + w_s \cdot (\mathbf{v}_c^T \mathbf{v}_r)^{k_e} \right\}, \tag{3.2}$$

where $\mathbf{n} = (n_x, n_y, n_z)^T$ is the microscopic normal direction of the paper surface at location $\mathbf{p}$, $\mathbf{o} = (o_x, o_y, o_z)^T$ is the position of the light source, $\mathbf{v}_i = (\mathbf{o}-\mathbf{p})/||\mathbf{o}-\mathbf{p}||$ is the incident light direction, l is the strength of the light, $1/||\mathbf{o}-\mathbf{p}||^2$ is a light-strength discounting factor as the received energy per unit area from a point light source is inversely proportional to the squared distance. $x^+ = \max(0, x)$, and $k_e > 0$ controls the gloss level of the surface. w_d and w_s are the weights for diffuse and specular components, and they have taken into account the effect of a constant surface albedo and other scaling factors. $\mathbf{v}_c$ is the camera's/sensor's direction, and $\mathbf{v}_r$ is the specular reflection direction which can be written in terms of the incident light direction $\mathbf{v}_i$ and the normal vector $\mathbf{n}$, i.e., $\mathbf{v}_r = (2\mathbf{n}\mathbf{n}^T - \mathbf{I})\mathbf{v}_i$, where $\mathbf{I}$ is the identity matrix. All $\mathbf{n}$, $\mathbf{v}_i$, $\mathbf{v}_c$, and $\mathbf{v}_r$ are unit-length column vectors.

3.2.3 Norm Map Estimation Using Photometric Stereo

A surface normal is a vector perpendicular to the tangent plane at a location of the surface, and a normal vector field is a collection of 3D surface normals over a 2D grid. A norm map

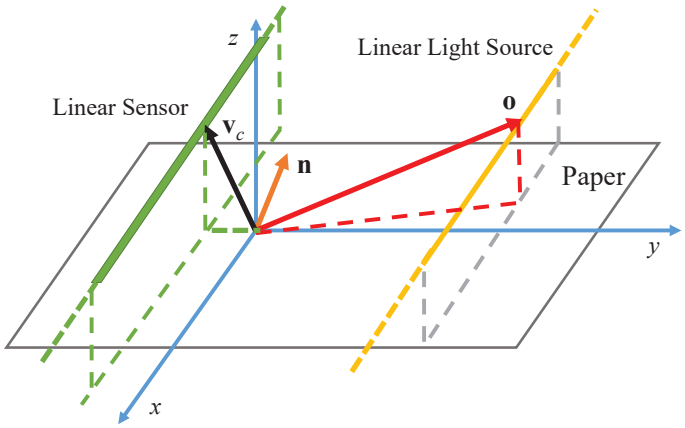


Figure 3.3: Configuration of the optical system of a flatbed scanner for scanning a paper sheet. The point of interest is located at the origin. The microscopic surface normal, $\mathbf{n}$, the camera/sensor direction, $\mathbf{v}_c$, and the location of one point on the linear light, $\mathbf{o}$, are shown.

is the normal vector field projected onto the xy-plane, which is a 2D vector field. The norm map has been shown to be a powerful discriminative feature for paper surfaces [6, 21, 22].

The state-of-the-art method for estimating norm maps of paper surfaces using commodity flatbed scanners [6, 19, 21, 22, 20] is described as follows. We assume the paper to be scanned is placed on the xy-plane passing through the origin as shown in Fig. 3.3. Without loss of generality, we assume that the point of interest is located at the origin. A linear light source is positioned in parallel with the x-axis and moving along the y-axis. We denote a specific location on the linear light source as $\mathbf{o}$ and the incident light direction is therefore $\mathbf{v}_i = (o_x, o_y, o_z)^T / ||(o_x, o_y, o_z)||$. Since the light source is very close to the paper surface, the linear light source appears to a point on the paper infinitely long in the x-direction.

Under the fully diffuse model, the intensity I of the reflected light of the point placed at the origin under the linear light of a flatbed scanner is a superposition of all rays diffusely reflected originating from the light source located at $\mathbf{o} = (o_x, o_y, o_z)$ for $o_x \in [-a, b]$, where $-a$ and b are the x-coordinates of the two ends of the linear light source and assume $0 < a < b$:

$$I = \int_{-a}^{b} l_r \, d\, o_x \approx l \cdot w_d \int_{-a}^{a} \mathbf{n}^T \frac{(o_x, o_y, o_z)^T}{||(o_x, o_y, o_z)||^3} \, d\, o_x, \tag{3.3}$$

where the approximation makes use the fact that the intensity of the point of interest contributed by the far portion $o_x \in (a, b]$ of the linear light source is very small, namely, $\int_a^b \mathbf{n}^T \mathbf{o} / ||\mathbf{o}||^3 d\, o_x \approx 0$.

In [6, 19, 21, 22, 20], images acquired using a scanner from two opposite directions are

used to estimate the x- or y-components of a norm map. Two images, I_{0° and I_{180°, are obtained when the paper is orientated at 0° and 180° on the xy-plane when being scanned. For a pixel of interest on the paper surface, the normal vector is $\mathbf{n}$, and a specific location on the light source is $\mathbf{o} = (o_x, o_y, o_z)^T$. When scanning the paper at 180°, it is equivalent that for the pixel of interest, the normal vector remains the same, while flipping the light's y coordinate, namely, changing the specific location on the light source into $\mathbf{o}' = (o_x, -o_y, o_z)^T$. [1] Their difference, $I_{0^\circ} - I_{180^\circ}$, can be shown to be in proportion to the y-component of the norm map, n_y, and therefore can be used as an estimator for n_y [6]:

$$I_{0^\circ} - I_{180^\circ} = l \cdot w_d \int_{-a}^{a} \mathbf{n}^T \frac{\mathbf{o} - \mathbf{o}'}{||(o_x, o_y, o_z)||^3} \, d\, o_x = s\, n_y, \tag{3.4}$$

where $s = 2l \cdot w_d\, o_y \int_{-a}^{a} ||\mathbf{o}||^{-3} d\, o_x$ is a constant. The x-component of the normal vector, n_x, can be estimated similarly using $I_{90^\circ} - I_{270^\circ}$.

3.3 Cancellation of Specular Components Under Flatbed Scanner Geometry

The state-of-the-art norm map estimation method [6, 21] reviewed in Section 3.2.3 assumes that paper surfaces reflect light in a fully diffuse way. However, if one observes carefully a paper patch at a close distance under a strong light while constantly changing the observation angle, he/she may observe some discrete spots with significant intensity fluctuation. These discrete spots are not fully diffuse, since perceived intensity due to diffuse reflected light should not depend on the location of the eye/sensor. For a spot dominated by the specular reflection, the perceived intensity could be much stronger or weaker than its neighboring spots dominated by the diffuse reflection. This is because the intensity given by the specular reflection has a different cause that depends on the angle between the directions of the eye and the reflected light, namely, $\arccos(\mathbf{v}_c^T \mathbf{v}_r)$. For these spots with a specular reflection component, the estimation of the normal vector may be very different from the true value if the specular component is neglected. To demonstrate this phenomenon, we contrast in Fig. 3.4 real photos captured by a mobile camera and their corresponding synthesized versions by only considering the diffuse component. The photos were captured in different camera orientations with different incident light directions. The synthesized versions were

[1] Note that this equivalence by flipping the y coordinate of the light is only valid for the fully diffuse model. In Section III that incorporates the specular component, we do not use this equivalence.

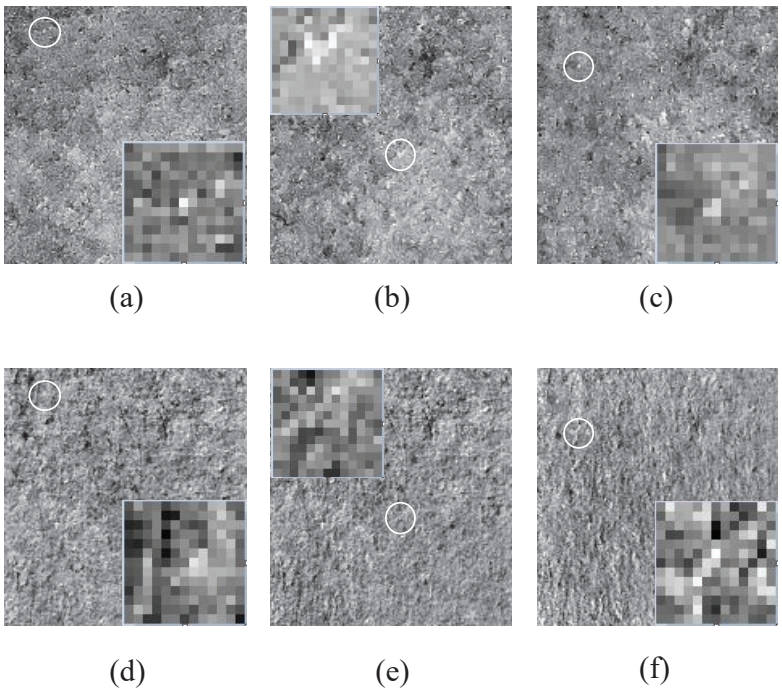


Figure 3.4: (a)–(c) Photos of a paper patch captured by a mobile camera from different angles with flashlight. (d)–(f) Synthetic images that consider only the diffuse reflection. The real photos have high-contrast spots that may be caused by specular reflection, whereas their contrast in respective synthetic images is much lower. Vertical paired images are to be compared, with circles highlighting collocated spots for visual comparison. The zoomed in versions in the circled areas are put in the corners of the images. (All pictures have undergone perspective transform, detrending, and contrast enhancement to better illustrate the idea.)

generated by first estimating the normal vector field assuming the fully diffuse Model 2 proposed in [22], and then rendering diffuse reflection images. It is revealed in Fig. 3.4 that the real photos in the first row have more highlights than the synthesized images in the second row. That could be due to the specular reflection. We circled some locations of high contrast in real photos that are surrounded by dark pixels. The corresponding locations in synthesized images do not have such high contrast.

We have demonstrated that in general geometric setups for capturing paper surfaces such as using cameras, there will be high-contrast spots in the captured images due to the specular reflection component. Blindly ignoring specular reflections in modeling and estimation may lead to imprecise norm map estimates. Next, we show analytically that, for the flatbed scanner geometry, the image subtraction approach remains a precise estimator even if specular reflection is taken into consideration. Using the generalized light reflection model (3.2) that contains the specular reflection term, the reflected intensity under a

scanner's linear light becomes:

$$\begin{aligned} I &= \int_{-a}^{a} l_r \, d\, o_x = l \int_{-a}^{a} \left(w_d \mathbf{n}^T \mathbf{v}_i + w_s \mathbf{v}_c^T \mathbf{v}_r \right) \frac{1}{||\mathbf{o}||^2} \, d\, o_x \\ &= l \int_{-a}^{a} \Big(w_d \mathbf{n}^T + w_s \mathbf{v}_c^T (2\mathbf{n}\mathbf{n}^T - \mathbf{I}) \Big) \mathbf{v}_i \frac{1}{||\mathbf{o}||^2} \, d\, o_x. \end{aligned} \tag{3.5}$$

Note that we set $(\mathbf{n}^T \mathbf{v}_i)^+ = \mathbf{n}^T \mathbf{v}_i$ when invoking (3.2) since the angle between $\mathbf{n}$ and $\mathbf{v}_i$ are rarely greater than 90°. We set $k_e = 1$ to capture the dominating linear relationship while ignoring the higher-order terms for analytic tractability.

When scanning the paper in two opposite directions, a more natural and direct modeling approach is not to flip the light's y coordinate as proposed in [6] and reviewed in Section 3.2.3 of this paper; instead, following the illustration of Fig. 3.3, we should capture the 180° rotation operation in the xy-plane resulting $\mathbf{n}' = (-n_x, -n_y, n_z)$ while leaving the incident light direction $\mathbf{v}_i$ and the camera direction $\mathbf{v}_c$ unchanged. Following the traditional procedure of subtracting one scanned image from another, we obtain:

$$I_{0^\circ} - I_{180^\circ} = s\, n_y + 2l \int_{-a}^{a} \left(w_s \mathbf{v}_c^T (\mathbf{n}\mathbf{n}^T - \mathbf{n}'\mathbf{n}'^T) \mathbf{v}_i \right) \frac{1}{||\mathbf{o}||^2} \, d\, o_x. \tag{3.6}$$

The x-component of camera direction $v_{cx} = 0$ since the camera/sensor in the scanner catches the light that is parallel to the yz-plane, and $n_z \approx 1$ since normal vectors are close to pointing straight up, as is revealed by Fig. 3.5—a histogram for n_z obtained from measurements using a confocal microscope. Substituting $\mathbf{v}_c = (v_{cx}, v_{cy}, v_{cz})^T$, $\mathbf{v}_i = \mathbf{o}/||\mathbf{o}||$ and $\mathbf{n}\mathbf{n}^T - \mathbf{n}'\mathbf{n}'^T = \begin{bmatrix} 0 & 0 & 2n_x n_z \\ 0 & 0 & 2n_y n_z \\ 2n_x n_z & 2n_y n_z & 0 \end{bmatrix}$ into (3.6), we obtain:

$$I_{0^\circ} - I_{180^\circ} = s\, n_y + 4l \int_{-a}^{a} w_s (v_{cz} n_x n_z, v_{cz} n_y n_z, v_{cx} n_x n_z + v_{cy} n_y n_z)(o_x, o_y, o_z)^T ||\mathbf{o}||^{-3} \, d\, o_x \tag{3.7a}$$

$$= s\, n_y + 2s' n_z \Big\{ n_y \Big[v_{cz} + v_{cy} o_z / o_y \Big] + n_x v_{cx} o_z / o_y \Big\} \tag{3.7b}$$

$$\approx \big[s + 2(v_{cz} + v_{cy} o_z / o_y) s' \big] n_y \tag{3.7c}$$

where $s' = 2l \cdot w_s o_y \int_{-a}^{a} ||\mathbf{o}||^{-3} d\, o_x$. We followed the procedure outlined in [21] to generate normal vectors from the heightmap acquired by a confocal microscope. Note that o_z and o_y are device-specific constants since the distance from the light source to the point being

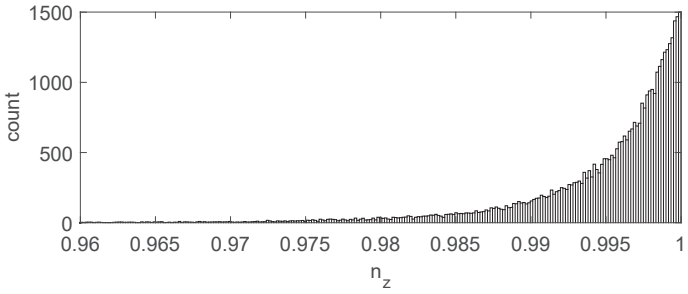


Figure 3.5: A histogram for the z-component of the normal vector field of a 2/3-by-2/3 inch^2 paper patch from confocal laser scanning microscope Keyence VKx1100 digitized at a spatial resolution of 5.38 μm.

captured in the xz-plane is fixed by the design of the scanner geometry. The final result in (3.7c) reveals that even though the specular reflection is taken into account, the traditional estimator is still linear in n_y due to the unique imaging setup by flatbed scanners. This would not be possible if v_{cx} were not zero since both n_x and o_z are usually nonzero.

Note that the result that the specular component does not play a role is largely contributed by the approximately symmetric integration bound from $-a$ to a demonstrated in (4.2), which is in turn guaranteed by the fact that the linear light is very close to the paper to be scanned in the z direction. The result we obtained in this section does not apply to more general geometric setups such as using mobile cameras discussed in other literature [7, 8, 9, 20, 21, 22]. This result also justifies the use of a flatbed scanner to obtain norm maps for surfaces other than paper that contain stronger specular components.

3.4 Scanner and Confocal Consistency

A preliminary study was reported in Section VII.C of [21] examining whether the norm map estimated from scanner acquired images are consistent with the ground truth, i.e., the norm map measured by the confocal microscope. The overall correlation between the scanner estimates and the reference was 0.28 (we reproduced this number in Table 3.1 for easy reference and comparison), indicating that the estimation, even though not that precise, was indeed related to the ground truth. However, in [21], only one physical paper patch was investigated. In this paper, we extended the inquiry of [21] by using a confocal collected dataset of one order of magnitude larger and investigate the blurring issue, aiming to confirm with higher confidence that the scanner estimated norm map are meaningful physical quantities and to gain better understandings about the characteristics

Table 3.1: Comparison of Performance of Various Features When Test Data From Scanner Correctly Match with Reference Data From Confocal Microscope

Feature	Correlation
Norm Map Based:	
Raw (dataset of [21])	0.28
Raw (new dataset)	0.357 (x), 0.301 (y)
Deblurred (new dataset)	0.442 (x), 0.396 (y)
Heightmap Based:	
Reconstructed heightmap	0.358
Detrended reconstructed heightmap	0.499
Third-highest spatial-frequency subband	0.714

of the scanner estimated norm maps.

3.4.1 Dataset Collection

In this paper, we created a new dataset of paper surfaces that will be made publicly available on the authors' websites after the publication of this paper. We collected data for 9 different paper patches of size $\frac{2}{3}$-by-$\frac{2}{3}$ inch^2 using flatbed scanners and a confocal microscope. The patches are from the same sheet of ordinary office printing paper. This is a more difficult case than the case where paper patches are obtained from different sheets of printing papers because the paper patches from the same sheet exhibit less variations due to the same manufacturing condition, time, and raw materials used. The papers with printing are not considered since we aim to derive the intrinsic physical features caused by the intertwisted wood fibers on the paper surface. Four out of nine paper patches were stuck to a microscope glass slide to create a rigid and consistently flat surface. A card stock was put between the paper and the glass slide to block any light from the backside of the paper. The other five paper patches were not stuck to anything. These two different setups mimic the conditions of patches in real-world scenarios.

Data related to the flatbed scanner include scanner acquired images. For image acquisition, we used a Canon CanoScan LiDE 110 flatbed scanner to acquire each patch from four orientations, i.e., 0°, 90°, 180° and 270°, and repeat such process three times for each physical patch to obtain three norm maps. The norm maps were estimated by taking the difference of images scanned in opposite directions, which is based on the fully diffuse model since we have analytically proved in Section 3.3 that the specular component can be neglected in the optical setup of a scanner. Then we repeated the image acquisition process

by using two other consumer-grade flatbed scanners that are the most popular on Amazon.com as of the summer of 2019: CanoScan Lide 300, and Epson Perfection V39. Using the three scanners, we obtain a total of nine norm maps for each paper patch. We resized the acquired patch images to 200-by-200 pixels. Data related to the confocal microscope include heightmaps of paper surfaces and norm maps derived from heightmaps that are accurate enough to be considered as ground truth. We used a Keyence VKx1100 confocal microscope with a 404 nm violet laser source to obtain heightmaps of paper patches. We followed the procedure in [21] to derive a 200-by-200 norm map from the heightmap for each paper patch: We estimated the normal vector for a pixel of interest by fitting a plane to the corresponding height values located in the z direction. The resolution in the z direction of the heightmap used in this data acquisition was 0.1 nm, which is much higher than 6 μm used in [21] and can therefore provide more accurate aggregated results for confocal generated norm maps. Due to the optical principles of confocal microscopy, the confocal norm map is accurate enough to be considered as the ground truth.

3.4.2 Initial Consistency Verification

We evaluated the consistency of the scanner estimated norm maps to the confocal measurements on the newly collected dataset following the same procedure as in [21]. For each paper patch, we calculated the correlation between the x-/y-component of the nine norm maps obtained from the scanners and the ground-truth norm map from the confocal microscope. Histograms of the correlation values are shown using the "original" legend of Fig. 3.6. The averaged correlation is 0.357 for the x-component of the norm map and 0.301 for the y-component, as summarized in Table 3.1, with the sample standard deviation 0.10 and 0.11, respectively. The averaged correlation values are close to the result, 0.28, reported in [21]. Our experimental results by using nine different paper patches confirm with higher confidence that the scanner estimated norm maps are meaningful physical quantities.

3.4.3 Consistency Verification by Compensating Blurring

Although the previous subsection confirms that scanner norm maps are meaningful estimates of physical quantities, the correlation slightly greater than 0.3 implies that there are still non-negligible factors contributing to the inconsistency. One such factor may be spatial blurring. In this subsection, we investigate the blurring effect due to the imaging pipeline of flatbed scanners on the accuracy of the estimated norm maps. Images captured by flatbed

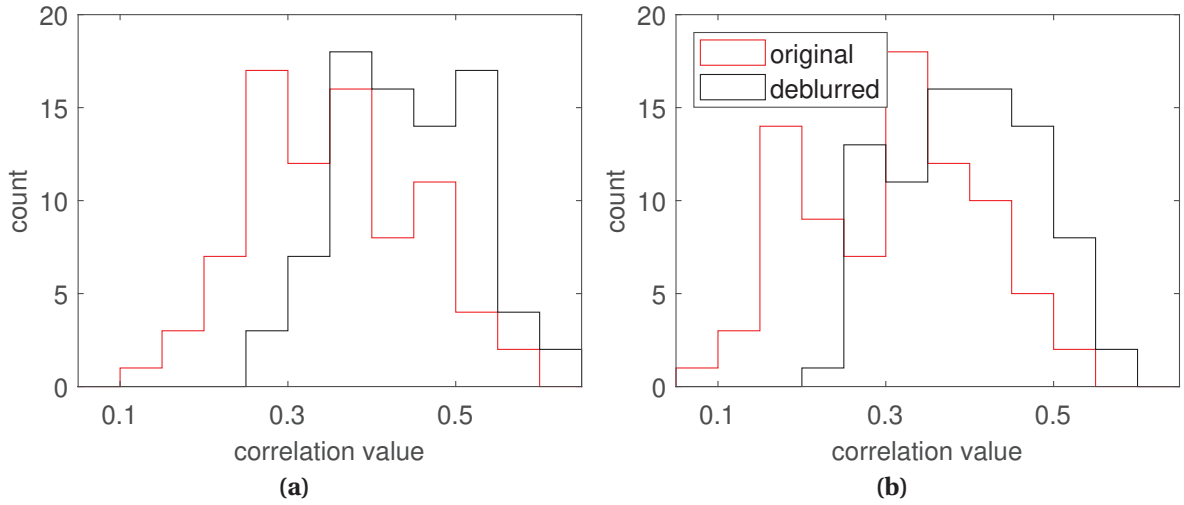


Figure 3.6: Histograms of correlation values between (a) x- or (b) y-component of norm maps estimated from scanner and confocal measurements. The averaged correlation increased from 0.357 to 0.442 for the x-component and from 0.301 to 0.396 for the y-component after deblurring.

scanners may be blurred due to out of focus, sensor/light/scanning platform motion, and the blooming effect of CCD sensors. Norm maps derived from blurred scanned images will therefore be a blurred version of the ground-truth norm maps. Below, we examine whether deblurring is possible with the help of confocal norm maps and investigate the characteristics of blurring filters.

Deblurred Norm Map

We explore using confocal norm maps to assist the deblurring process and evaluate the quality of deblurred norm maps. We denote the norm map from the confocal measurement as $\mathbf{C}$, and the norm map estimated by subtracting the two images scanned in opposite directions as $\mathbf{S}$. We model the relation between the ground truth $\mathbf{C}$ and the scanner norm map considered to be blurred using the linear model:

$$\mathbf{C} = \mathbf{H}_{\text{deblur}} * \mathbf{S} + \mathbf{e}, \tag{3.8}$$

where $\mathbf{H}_{\text{deblur}}$ is a linear spatial invariant (LSI) deblurring filter, $\mathbf{e}$ is an error term, and $*$ is the 2D convolution operator. We create separate models for x- and y-components of a norm map and for each paper patch. Regarding the size of the deblurring filter, we empirically set the dimension such that the pixels with significant contributions to the convolutional

result will be retained. Specifically, we use an oversized filter, i.e, 25-by-25, to preliminarily estimate filter coefficients when the filter dimension is not significantly constrained. Since it is a deblurring filter, the coefficient of the pixel in the center must dominate in magnitude when compared to other pixels. We observe that most coefficients with magnitude greater than 10% of that of the centering pixel are located in the centering 7-by-7 area. Hence, we will use 7-by-7 as the size to formally estimate the deblurring filters as follows.

To avoid model overfitting, we estimate the deblurring filter $\mathbf{H}_{\text{deblur}}$ using cross-validation with the cost function in the ridge regression form:

$$\min_{\mathbf{H}_{\text{deblur}}} ||\mathbf{C} - \mathbf{H}_{\text{deblur}} * \mathbf{S}||_F^2 + \lambda ||\mathbf{H}_{\text{deblur}}||_F^2, \tag{3.9}$$

where $||\cdot||_F$ is the Frobenius norm and λ is a regularization parameter controlling model complexity. With norm map of size 200-by-200, the filter size to be 7-by-7, there are 34596 data points to solve for $\mathbf{H}_{\text{deblur}}$. We first use 10-fold cross-validation to find the regularization parameter that minimizes the cross-validation error. We then apply one standard error rule to choose an updated regularization parameter that corresponds to the most parsimonious model and use the coefficients at this time as the final estimate for the deblurring filter, $\hat{\mathbf{H}}_{\text{deblur}}$.

We use the trained filter $\hat{\mathbf{H}}_{\text{deblur}}$ to derive the deblurred norm map, $\hat{\mathbf{C}} = \hat{\mathbf{H}}_{\text{deblur}} * \mathbf{S}$, and compare it with the ground truth, the confocal norm map $\mathbf{C}$. The histograms of the correlation values between $\mathbf{C}$ and $\hat{\mathbf{C}}$ in x- and y-directions are shown using the "deblurred" legend of Figs. 3.6(a) and (b), respectively. Due to deblurring, the averaged correlations increased from 0.357 to 0.442 for the x-component and from 0.301 to 0.396 for the y-component. Their sample standard deviations also both decreased to 0.08. The increased correlations and decreased standard deviations after deblurring indicate that blurring is a factor to the lowered quality of scanner estimated norm maps. It is also noted that, in light of the non-negligible but limited improvement of the correlation due to deblurring, more investigations are needed to reveal other factors limiting the accuracy of the scanner norm maps. In the practical authentication system in Section 3.5.3, we do not apply deblurring due to the limited improvement of correlation.

Shape of Blurring Filter

It is also interesting to estimate the blurring filter to directly reveal the characteristic of blurring. First, we use a nonparametric approach to determine the shape of the blurring filter, which can avoid bias due to imposing a parametric model that may potentially cause

mismatch. We estimated a 7-by-7 LSI filter $\mathbf{H}_{\text{blur}}$ such that $||\mathbf{S}-\mathbf{H}_{\text{blur}}*\mathbf{C}||_F^2$ was minimized. Since the coefficients in the blurring filter should all be non-negative, we estimated the blurring filter $\mathbf{H}_{\text{blur}}$ using non-negative least-squares. Because the blurring filter has a lowpass nature and is an inverse filter of the deblurring filter, even a filter smaller than 7-by-7 should be sufficient to adequately capture the blurring effect.

After obtaining an estimate of the blurring filter defined on a 7-by-7 grid, we interpolated the filter spatially and drew the 3D meshes and contours/level curves to visualize its shape. Figs. 3.7(a) and (b) depict two typical 3D meshes for blurring filters derived from the x- and y- components of the norm map of one paper patch, respectively. Figs. 3.7(c) and (d) show one contour per filter for all paper patches used in our experiments. The shapes of the contours reveal that the blurring filters for the x-component of the norm maps have larger spread in the y-direction and the blurring filters for the y-component of the norm maps have larger spread in the x-direction. The shapes of the contours are similar, so different scanners have similar blurring effects.

Since the blurring filters are close to bell-shaped, we further obtain a quantitative description of the spread for the blurring filters using parametric Gaussian filters. Let us assume a blurring filter that is generated by discretizing and normalizing a separable bivariate Gaussian function on a 7-by-7 grid. The bivariate Gaussian is parameterized by $\mu_x, \mu_y, \sigma_x, \sigma_y$, where (μ_x, μ_y) describes the location of the filter, σ_x and σ_y are the standard deviations of the Gaussian filter in the x and y directions. We assume the Gaussian to be separable based on the fact that blurring in the x and y directions have different causes due to the geometry of the flatbed scanner, and the observations from Fig. 3.7 that nonparametrically estimated filters' contours are oriented horizontally or vertically. We estimate $\mathbf{H}_{\text{blur}}^{\text{Gaussian}} = \mathbf{G}(\mu_x, \mu_y, \sigma_x, \sigma_y)$ by solving the following minimization problem:

$$\min_{\mu_x,\mu_y,\sigma_x,\sigma_y} ||\mathbf{S}-\mathbf{G}(\mu_x, \mu_y, \sigma_x, \sigma_y)*\mathbf{C}||_F^2. \tag{3.10}$$

Since this problem is nonconvex, we numerically solve it with the following starting point configurations by taking into consideration the nonparametric results summarized in Fig. 3.7: $\sigma_x = \sigma_y = 1$, and μ_x, μ_y uniformly randomly drawn from -0.5 to 0.5. The estimated standard deviations in the x- and y-components of the norm maps for different paper patches are shown in Fig. 3.8, which are consistent with the results in Fig. 3.7.

The results of parametric Gaussian filters confirmed the following patterns obtained from the nonparametric least-squares method: i) the variance in the x-direction is smaller for the x-component of the norm map, and ii) the variance in the y-direction is smaller

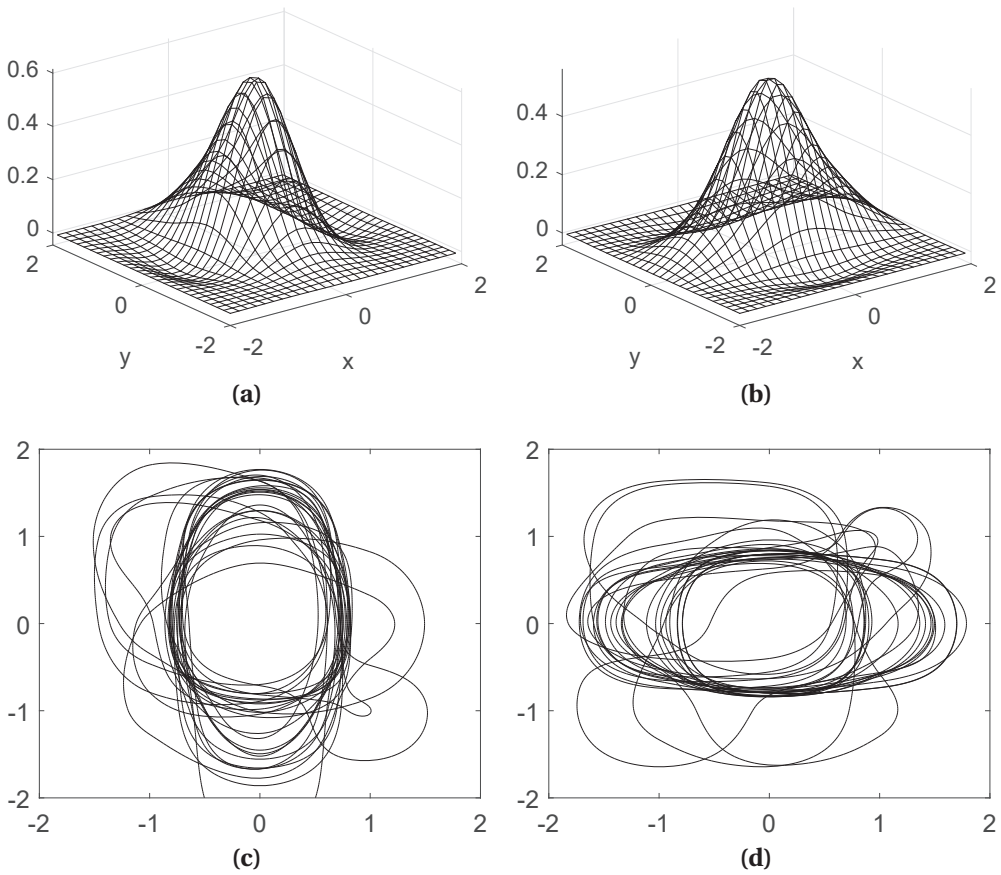


Figure 3.7: Typical 3D mesh for the blurring filter for (a) x- or (b) y-component of the norm map of a paper patch. We also overlay the contour graphs (one contour per contour graph) for all nine paper patches to illustrate the shape of blurring filters for (c) x- or (d) y-component of norm maps. The blurring filters for x-component of norm maps have larger variance in y-direction and the blurring filters for y-component of norm maps have larger variance in x-direction.

for the y-component of the norm map. Note that a smaller variance indicates a weaker blurring effect. This phenomenon could be explained by the unique optical setup of flatbed scanners. The y-component of the norm map of the paper patch is estimated by taking the differences of two images scanned in the opposite directions along the y-direction. During the scan, the linear sensor bar in the scanner is parallel to the x-axis, as shown in Fig. 3.3. The optical blurring along the direction of the linear light and the CCD blooming effect may result in more blurring along the x-direction, making the variance in the x-direction larger in the y-component of norm map.

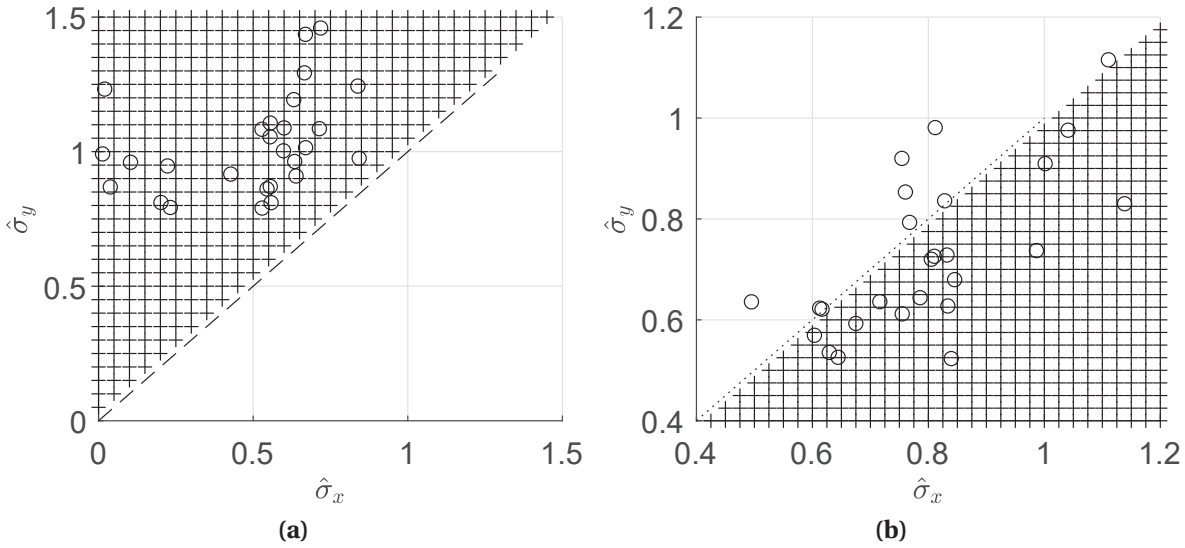


Figure 3.8: Scatter plot of $(\hat{\sigma}_x, \hat{\sigma}_y)$ for the blurring filters of (a) x- or (b) y-component of the norm maps for all nine paper patches. In x-component of norm map the variance in x-direction is smaller, and in y-component of norm map the variance in y-direction is in general smaller, as illustrated through the shaded regions.

3.5 Heightmap as a Discriminative Feature

Although the norm map has been shown to be a powerful discriminative feature [6, 21], when it is used in a practical authentication system, it is desirable to further increase the discriminative power to ensure a better performance. Previous work in [22] used the estimated norm map to reconstruct the 3D surface/heightmap and discovered that high-frequency subbands of reconstructed heightmap are more powerful than the norm map in describing the uniqueness of a physical surface. The result in [22] was demonstrated for mobile cameras and in this section, we investigate whether a similar conclusion can be drawn for flatbed scanners.

3.5.1 Z-Component Estimation From Norm Map

In this subsection, we propose an estimator for the z-component of the normal vector field based on a known norm map for surface reconstruction. Surface reconstruction in general requires a normal vector field containing at each location a 3-D description about the orientation [14, 28]. However, using images acquired by scanners and the estimation technique presented in Section 3.3, only the norm map, i.e., the scaled versions of the

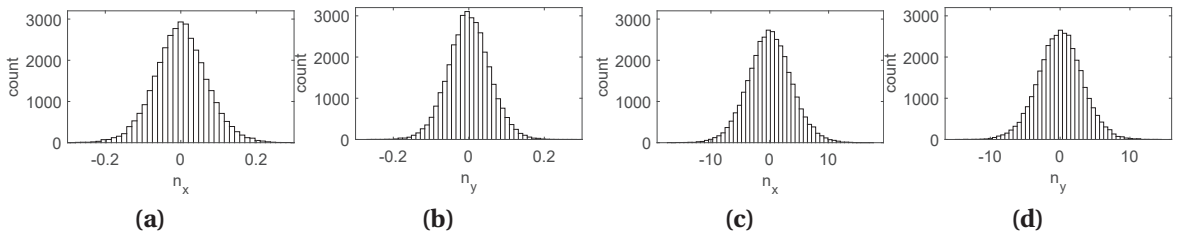


Figure 3.9: Histograms for (a) x- and (b) y-components of norm map from confocal microscope. Histograms for (c) x- and (d) y-components of norm map from scanner. Note that the components calculated from scanner are off by an unknown scaling factor. The distributions are Gaussian-like and roughly centered around zero.

x- and y-components of the normal vector field, $(n_x^{(s)}, n_y^{(s)})$, are available. The authors of [22] proposed a distribution matching approach to estimate scalars α_x and α_y that correctly normalizes the norm map so that the z-component can be calculated using $\hat{n}_z = \left[1-(n_x^{(s)}/\hat{\alpha}_x)^2-(n_y^{(s)}/\hat{\alpha}_y)^2\right]^{1/2}$, where the quantities with hats are the corresponding estimated values. The distribution matching approach finds the best $\hat{\alpha}_x$ and $\hat{\alpha}_y$ such that the standard deviations of $n_x^{(s)}/\hat{\alpha}_x$ and $n_y^{(s)}/\hat{\alpha}_y$ will match those of the confocal. However, details for obtaining $\hat{\alpha}_x$ and $\hat{\alpha}_y$ were not given. Below, we justify the approach proposed in [22] and propose a least-squares formula for estimating a shared scalar α for both directions. We first examine the real data to support subsequent model design. We show histograms for the x-, y-, and z-components of the normal vector field in Figs. 3.9(a), 3.9(b), and 3.5(a), respectively. From the histograms, we can see that normal vectors are on average pointing straight up due to large n_z and are without obvious bias in both x- and y-directions. The distributions are Gaussian-like and centered around zero. We also plot the histograms for x- and y-components of the norm map that are scaled. We observe that they are similarly distributed as those from the confocal but scaled, centered around 0. The above observation on the real data implies that a scaling relation is enough to connect the norm map to the first two components of the normal vector field, namely $n_x^{(c)}$ and $n_y^{(c)}$. Since the x- and y-components of the norm map are obtained by the same scanning process with the only difference being scanning directions, a shared multiplicative scalar should be used for both dimensions, namely,

$$(n_x^{(s)}, n_y^{(s)}) \approx \alpha \cdot (n_x^{(c)}, n_y^{(c)}), \tag{3.11a}$$

$$(\sigma_x^{(s)}, \sigma_y^{(s)}) \approx \alpha \cdot (\sigma_x^{(c)}, \sigma_y^{(c)}), \tag{3.11b}$$

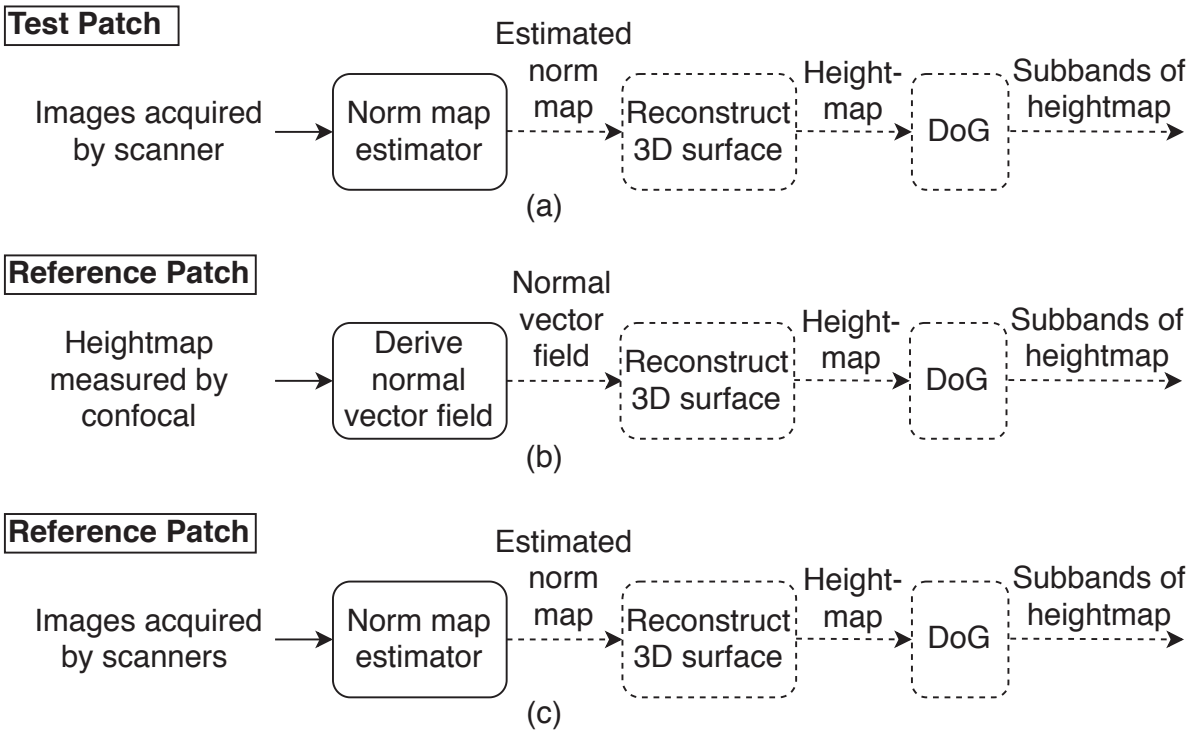


Figure 3.10: (a) Block diagram for obtaining features from test patch using images acquired by flatbed scanner. Block diagrams for obtaining features from reference patch using (b) measurement from confocal microscope, or (c) images acquired by flatbed scanner. The norm map, the heightmap, or the subbands can be used as discriminative features. The blocks/processes with dashed boundaries should be ignored when their inputs are used as features.

where (3.11b) was obtained by considering α as a constant and other components in (3.11a) as random variables, and by applying the variance operation to both sides of (3.11a). Estimating α using least-squares from (3.11b), we obtain

$$\hat{\alpha} = (\sigma_x^{(s)}\sigma_x^{(c)} + \sigma_y^{(s)}\sigma_y^{(c)}) \Big/ (\sigma_x^{(c)^2} + \sigma_y^{(c)^2}), \tag{3.12}$$

which blends in the scaling effect in both directions. This formula allows the calculation of a scalar for a scanner norm map by using merely two summary statistics of the paper surface, $\sigma_x^{(c)}$ and $\sigma_y^{(c)}$, that are determined by physical characteristics of papers and are stable numbers for papers of the same type [29].

3.5.2 Heightmap and Subbands as Discriminative Features

In [22], the authors have shown experimentally for mobile camera acquired images that the high frequency subbands have been proved to be powerful discriminative features for authentication. In this work, we validate the method of [22] using flatbed scanner acquired images. We follow the procedure in [22] to reconstruct 3D heightmaps of paper patches and derive the subbands of the reconstructed heightmaps for authentication. The reference data is from a confocal microscope. Each paper patch was scanned once by the confocal microscope. The test data is obtained from scanners. Each paper patch was scanned by one scanner three times, and there are three different scanners used. Thus, each paper patch has one ground-truth heightmap from the confocal microscope and nine reconstructed heightmaps from scanners.

Here, we summarize the benefit of using detrended heightmaps and more details are given in Section 3.9 of the supplementary document. The correlation value using reconstructed heightmaps improved to 0.358 from 0.357 or 0.301 when using the norm map as the discriminative feature, as shown in Table 3.1. When using the detrended heightmap as discriminative feature, the correlation value further improved to 0.499. This result is consistent with that reported in [22] where a mobile camera was used as the acquisitions device. Hence, the detrended heightmap is a more powerful discriminative feature than the norm map.

We also summarize the benefit of using high spatial-frequency subbands of heightmaps and more details are given in Section 3.10 of the supplementary document. We decomposed the reconstructed heightmap into ten spatial subbands corresponding to a DoG representation as reviewed in Section 3.2.1. Using the third-highest spatial-frequency subband instead of the detrended heightmap, the correlation value improved from 0.499 to 0.714. The estimated EER as a function of subband index is shown in Fig. 3.11, and a smaller subband index corresponds to a higher spatial frequency. When using the third-highest spatial-frequency subband, the EER achieved 10^{-36} or 10^{-8} under the Gaussian or Laplacian tail extrapolation assumption, which is a large improvement over 10^{-11} or $10^{-4.5}$ when using a detrended heightmap. The high spatial-frequency subbands are more powerful discriminative features than detrended heightmaps when using flatbed scanners for paper surface-based authentication.

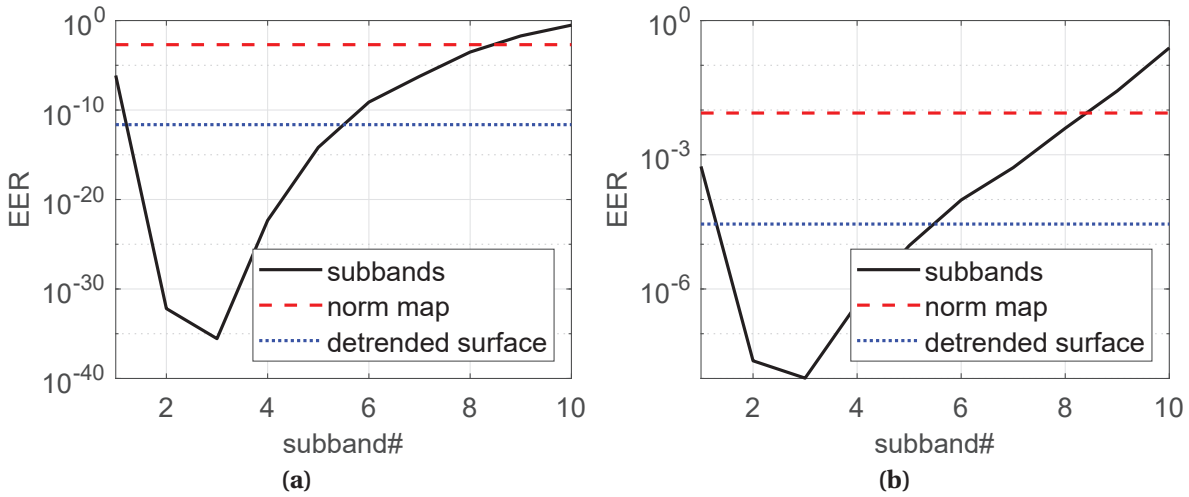


Figure 3.11: EER calculated for every subband when correlation values are believed to follow (a) Gaussian or (b) Laplace distributions. The reference data is obtained by a confocal microscope and the test data is acquired with flatbed scanners. The third-highest spatial-frequency subband is the most powerful in describing the uniqueness of physical surfaces. Horizontal lines correspond to the performance when the norm map or detrended surface/heightmap is used as the discriminative feature.

3.5.3 Practical Authentication System

In this subsection, we examine a practical authentication system that uses flatbed scanners to acquire both test and reference data. We compare using every subband of the heightmap as the discriminative feature to the traditional feature, i.e., the norm map, and measure the authentication performance in EER. The diagrams for generating the subbands in the authentication system for test and reference patches are shown in Figs. 3.10(a) and (c), respectively. The reference and test patches are both images acquired by scanners.

In the practical authentication system, we use scanners to capture the reference data instead of using the confocal microscope because scanners are easier to automate and more affordable for practical deployment. Each paper patch was scanned three times by each of the three scanners. We obtained nine norm maps using scanners for each paper patch. For the matched case, we chose two norm maps from the nine norm maps each time as a test-reference pair, forming a total of 36 pairs for each paper patch. Given the nine physical pieces of paper patches, this leads to a total of $36 \times 9 = 324$ data points of correlation values for statistical analysis. For the unmatched case, each paper patch pair gives $9 \times 9 = 81$ data points, and there are $\binom{9}{2} = 36$ paper patch pairs. Theoretically there are totally $81 \times 36 = 2916$ data points if using all paper patches. To mimic a practical scenario, we

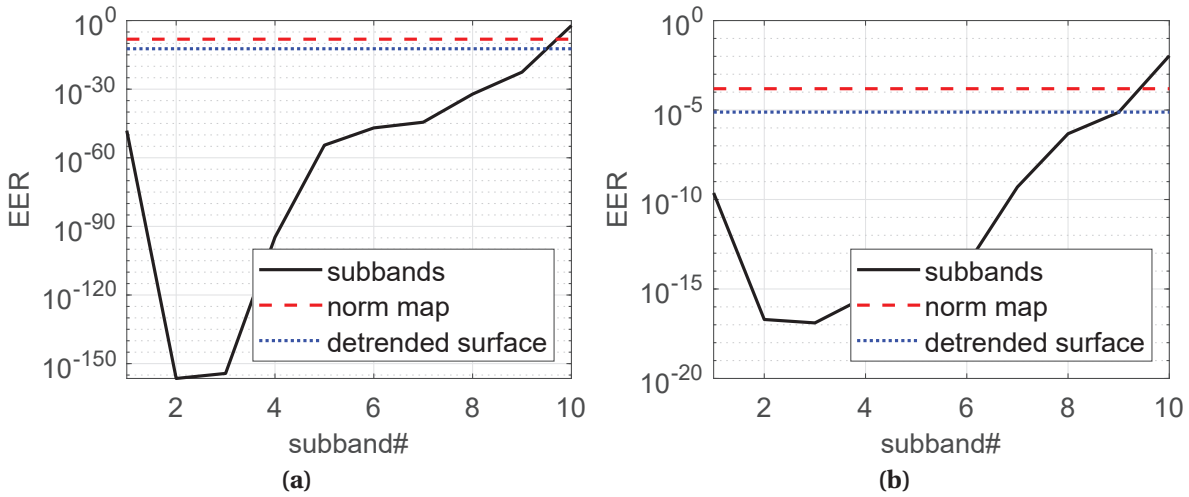


Figure 3.12: EER calculated for every subband when correlation values are believed to follow (a) Gaussian or (b) Laplace distributions. The second-highest spatial-frequency subband has the most powerful authentication capability in a practical setup that scanners are used to acquire reference data. Horizontal lines correspond to the performance when the norm map or detrended surface/heightmap is used as the discriminative feature.

randomly chose one paper patch from the rest of the paper patches to obtain the reference data, leading to a random subset of 729 data points for the unmatched case.

We reconstructed 3D surfaces from the norm maps and obtained the subbands of the heightmap as the discriminative features. We calculated the correlation values of subbands between the test and reference data. We calculated the EER for every subband and plotted the results in Fig. 3.12. When correlation values are believed to follow Gaussian or Laplace distributions, the EER are about 10^{-157} and 10^{-17} at the second-highest spatial-frequency subband, respectively. We also compared the performance of subbands of heightmap to that of the norm map and detrended heightmap, as shown by horizontal lines in Fig. 3.12 and they are much larger than using the second-highest spatial-frequency subband. Hence, in the practical system that uses a scanner to acquire reference data, the authentication performance of the second-highest spatial-frequency subband is much better than that of the norm map or detrended surfaces in terms of EER. In Table 6.1, we summarized the authentication performance of the practical authentication system in this work. For comparison, we also reproduced the results in [22] where mobile cameras instead of scanners were used to obtain the test data. We compared the best EER of the subbands when assuming the correlation values are Gaussian and Laplacian distributed. The performance of the practical authentication system in this work using scanners to obtain test data is

Table 3.2: Comparison of Performance of Practical Authentication System When Test Data is Obtained from Mobile Camera or Scanner

Test device	Reference device	Feature	EER (Gaussian and Laplacian)
mobile camera	scanner	norm map	10^{-5} and 10^{-3}[22]
mobile camera	scanner	subband	10^{-8} and 10^{-3}[22]
scanner	scanner	norm map	10^{-9} and 10^{-4}
scanner	scanner	subband	10^{-157} and 10^{-17}

much better than using a mobile camera in terms of EER.

3.6 Size of Paper Patch, Digitization Resolution, and Perturbation of Alignment

3.6.1 How Large Should the Size of the Paper Patch Be?

Throughout the experiments of this work, the size of the paper patch was fixed to be $\frac{2}{3}$-by-$\frac{2}{3}$ inch^2 and discretized to 200-by-200 pixels. A natural research question pertaining to a practical deployment is: How does the size of the paper patch affect the authentication performance? To investigate this question, we successively cut one heightmap into four heightmaps, empirically calculated the EER using the smaller heightmaps after each cut, and examined how the EER changes as the number of cuts increases. More specifically, we regarded the heightmap's center 160-by-160 pixels as the root patch that had not been cut. After the first cut, the resulting heightmaps were of the size 80-by-80 pixels. At each cut level, we calculated the correlation values against confocal references. We observed that, after each cut, the mean of correlation values were almost unchanged, whereas the standard deviation would increase by a factor of ~ 2 times for unmatched cases and ~ 1.5 times for matched cases. We plotted the sample standard deviations of the correlation values as a function of the number of cuts in Fig. 3.13. We further calculated EERs at each cutting level and plotted EERs against the block edge size in Fig. 3.14, in which block edge size $= 1$ corresponds to using 160 pixels. As expected, the authentication performance in EER improves as the block size increases.

Below, we analytically show that the EER is exponentially decreasing in the size of paper patch when correlation values are assumed to be Laplacian distributed. Using (3.23b) and the variance formula of a Laplace random variable, $\lambda = \sqrt{2}/\sigma$, the EER can be rewritten as

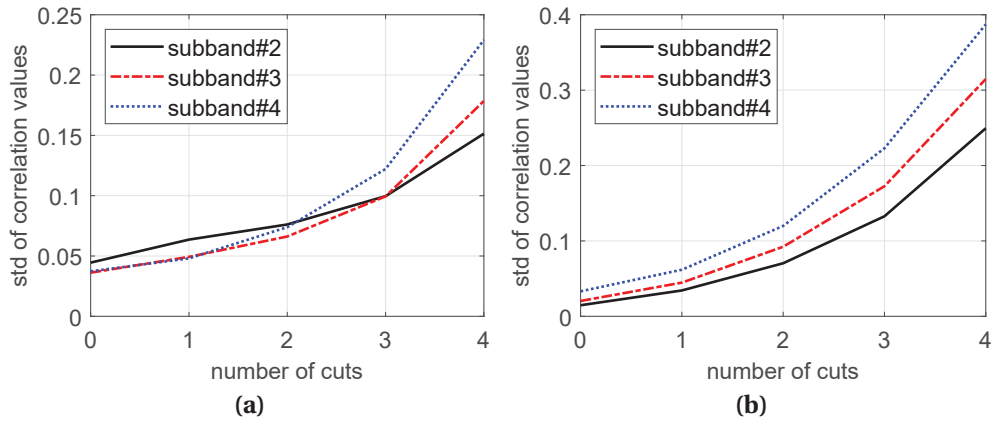


Figure 3.13: Sample standard deviations of the correlation values when cutting the paper patch into blocks under (a) matched and (b) unmatched cases. The standard deviations of the correlation values in spatial-frequency subbands #2–#4 increase exponentially when cutting paper patches into small blocks.

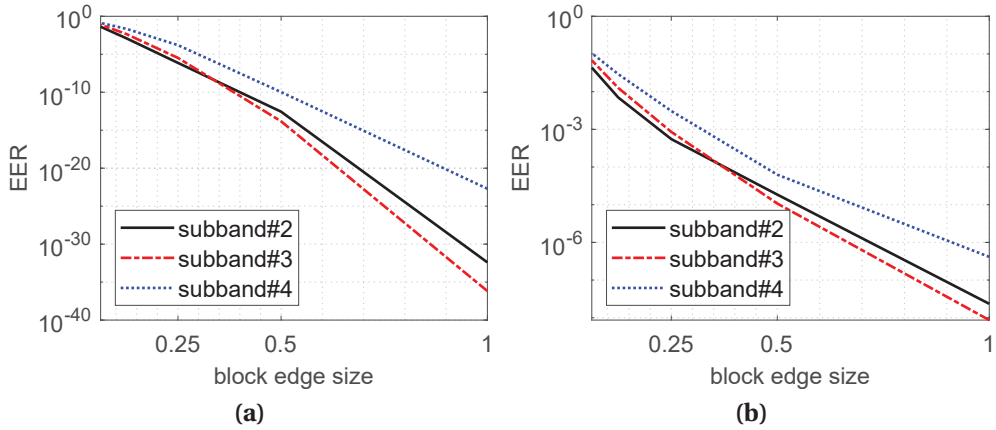


Figure 3.14: After cutting paper patches into blocks, EERs against the block edge length when assuming (a) Gaussian and (b) Laplace distributions. Size of 1 corresponds to the edge length of the original patch. The EERs decrease when the block edge length increases.

$\text{EER} = \frac{1}{2}\exp[\frac{\sqrt{2}}{\sigma_0+\sigma_1}(\mu_0-\mu_1)]$. After n cuts, the ERR can be expressed as

$$\text{EER}(n) = \frac{1}{2}\exp[\frac{\sqrt{2}}{2^n\sigma_0 + 1.5^n\sigma_1}(\mu_0-\mu_1)] \tag{3.13a}$$

$$\approx \frac{1}{2}\exp[\sqrt{2}\cdot 2^{-n}(\mu_0-\mu_1)/\sigma_0], \tag{3.13b}$$

where (3.13a) incorporates the empirically observed exponential increase of the standard deviations in the previous paragraph, and (3.13b) is approximately true for large n. Since

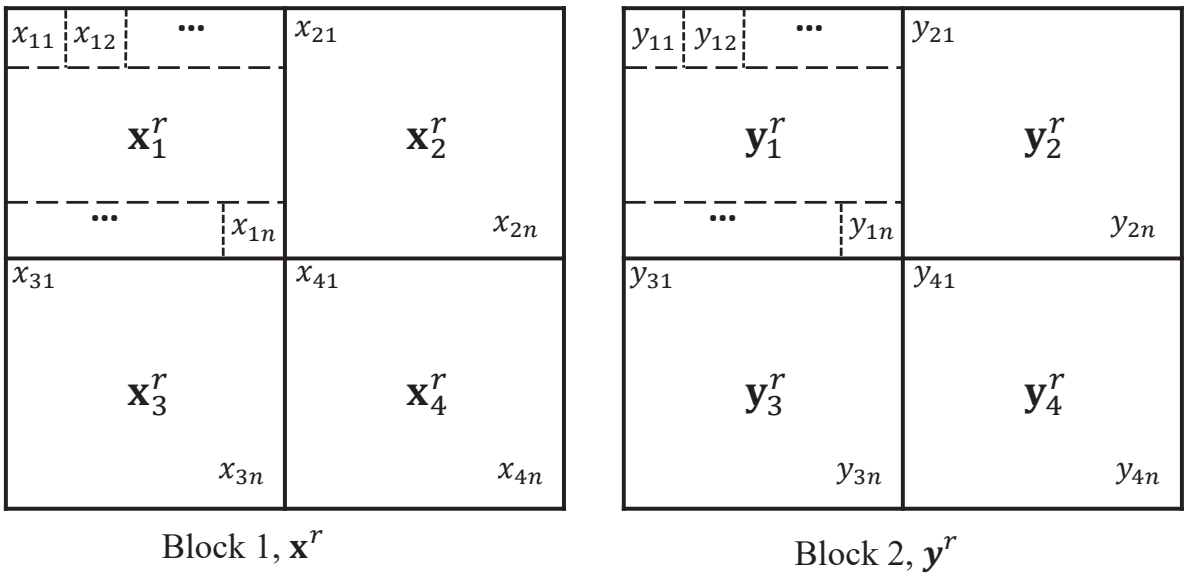


Figure 3.15: Sample correlation coefficients: ρ, between two blocks; ρ_i, between two collocated subblocks with index i. Detailed definitions are as follows: $\rho_i = \mathrm{Corr}(\mathbf{x}_i^r, \mathbf{y}_i^r),\ i = 1,\ldots,4$, and $\rho = \mathrm{Corr}(\mathbf{x}^r, \mathbf{y}^r)$, where the superscript "r" stands for the raw image data before sample mean is removed. $\mathbf{x}_i^r$ and $\mathbf{y}_i^r$ are length-n column vectors containing all pixel values of the respective subblocks. $\mathbf{x}^r$ and $\mathbf{y}^r$ are concatenated column vectors where $\mathbf{x}^r = (\mathbf{x}_1^r, \cdots, \mathbf{x}_4^r)$ and $\mathbf{y}^r = (\mathbf{y}_1^r, \cdots, \mathbf{y}_4^r)$.

2^{-n} is proportional to the block edge size after n cuts, $\log(\mathrm{EER}(n))$ is linearly decreasing in the block edge length, which is consistent with Fig. 3.14(b). When the edge length decreases from 160 pixels (or 0.53 inches) to 80 pixels (or 0.27 inches), the performance drops from around 10^{-9} to 10^{-5} in EER. To conclude, a larger patch size will lead to better authentication performance, and given a certain paper type, experiments similar to the one demonstrated in this subsection may be conducted to determine the patch size needed to achieve a certain performance level.

Below, we justify the exponential increase of the standard deviation for correlation values as the number of cuts increases. First, we claim the following finite-sample relation between the sample correlation coefficient of a block, ρ, and the sample correlation coefficients of its nonoverlapping, equal-sized subblocks, $\{\rho_i\}_{i=1}^4$, namely,

$$\rho \approx \frac{1}{4}\sum_{i=1}^{4}\rho_i. \tag{3.14}$$

The blocks and subblocks are illustrated in Fig. 3.15, and ρ and ρ_i's are defined in the caption. The relation of (3.14) is justified in the Appendix with a proof in the asymptotic

case and an observation in the finite-sample case. With the claimed relationship (3.14), we investigate the increase in variance after one cut. We consider the correlation values $\{\rho_i\}_{i=1}^4$ as random variables that are identically distributed. In the unmatched scenario, the correlation values should have a zero mean and correlation values produced by neighboring blocks that do not have reasons to be dependent. We used experimental results to confirm that $\mathrm{Cov}(\rho_i, \rho_{i'}) = 0, \forall i \neq i'$, for the unmatched case. After cutting the heightmap into four subblocks, we calculated correlation values $\{\rho_i\}_{i=1}^4$. There were 81 correlation values for the ith block location, and we ordered them into a vector $\boldsymbol{\rho}_i$. We used the sample correlation value $\mathrm{Corr}(\boldsymbol{\rho}_i, \boldsymbol{\rho}_{i'})$ to estimate the theoretical quantity $\mathrm{Corr}(\rho_i, \rho_{i'})$. The sample mean and standard deviation values of correlation values $\mathrm{Corr}(\boldsymbol{\rho}_i, \boldsymbol{\rho}_{i'})$ for subbands #2–#4 are around -0.1 and 0.2, respectively. A t-test shows that the correlation values are not significantly different from zero (p-value $= 0.249$), which supports our hypothesis. Hence, by applying the variance operation to (3.14) and using $\mathrm{Cov}(\rho_i, \rho_{i'}) = 0$, we obtain for the unmatched scenario:

$$\mathrm{Var}(\rho_1) = 4\,\mathrm{Var}(\rho). \tag{3.15}$$

Therefore, after one cut the standard deviation of the correlation values will increase by a factor of 2, which is consistent with the aforementioned empirical observation. For the matched case, the correlation values produced by neighboring blocks should be positively correlated, i.e., $\mathrm{Cov}(\rho_i, \rho_{i'}) > 0$ for $i \neq i'$. For example, $\{\rho_i\}_{i=1}^4$ are likely to be simultaneously all high or all low, but it is less likely to have two high values and two low values. We calculated the sample correlation value $\mathrm{Corr}(\boldsymbol{\rho}_i, \boldsymbol{\rho}_{i'})$. The sample mean and standard deviation values of correlation values $\mathrm{Corr}(\boldsymbol{\rho}_i, \boldsymbol{\rho}_{i'})$ for subbands #2–#4 are around 0.4 and 0.2. A t-test shows that the correlation values are significantly larger than zero (p-value $= 8.48 \times 10^{-8}$), which also supports our hypothesis. Applying the variance operation to (3.14) and considering the postive correlation among $\rho_i s$, we obtain for the matched scenario:

$$\mathrm{Var}(\rho_1) = 4\,\mathrm{Var}(\rho) - \frac{1}{2}\sum_{i \neq i'} \mathrm{Cov}(\rho_i, \rho_{i'}) < 4\,\mathrm{Var}(\rho), \tag{3.16}$$

which corresponds to an increase in standard deviation by a factor of less than 2 after one cut, which is also consistent with the empirical observation of a factor of 1.5.

3.6.2 Resolution of Norm Map

Another research question closely related to the issue of the patch size studied in the previous subsection is the choice of resolution for digitizing the patch. The resolution used in the experiments of this work is 300 pixels per inch (ppi) or 84.7 μm per pixel, i.e., a patch of $\frac{2}{3}$-by-$\frac{2}{3}$ inch^2 is digitized to 200-by-200 working pixels. According to Section VII.C and Fig. 14 of [10], within the squared regions of the size of a working pixel, most surfaces "were not flat because the scale of fibers is smaller than the area of a working pixel." Shall we reduce the size of working pixels so that the surfaces correspond to pixels can be more flat so as to improve the characterization of the structure of the paper, and in turn, improve the authentication performance?

We first examine the distribution of the orientations of squared areas of the size of a working pixel when a paper patch of size $\frac{2}{3}$-by-$\frac{2}{3}$ inch^2 is digitized to 300 ppi. We use the tangent plane algorithm in [21] to obtain surface normal vectors using a heightmap captured by a confocal microscope. We denote the angle formed by the surface normal vector and z-axis by θ. A histogram for the sine of the working pixel's orientation, $\sin\theta$, is shown in Fig. 3.16(a), with a sample mean of 0.078 (or 4.5°) and sample standard deviation of 0.045 (or 2.6°) for $\sin\theta$. These estimated angles are very small compared to the actual angles that could be formed by intertwisted fibers. However, when considering a relatively larger area covered by a working pixel that may contain multiple fibers segments, it is reasonable that prominently tilted structures are smoothed out.

Next, we increase the resolution of the norm map to see how the distribution of surface orientation may change, and whether there exists any resolution that outperforms others. We vary the resolution ranges from 150 to 1200 ppi to cover a practical working range for consumer-grade flatbed scanners. As we increase the resolution, working pixels will shrink in size, leading to larger estimated angles. At each resolution level, we calculate the sample mean and sample standard deviation of $\sin\theta$ and plot the results in Fig. 3.16(b). The plot reveals that both average angle and the angle variation increase as the resolution increases, which is reasonable since finer details of the microstructure of paper surface are captured. This means that using higher resolution (and a fixed number of pixels), a digitized normal vector field is likely to contain more randomness and therefore can potentially lead to higher authentication performance by reducing the false negative rate. However, this monotonic smoothly increasing curve does not strongly justify the use of a particular resolution among others within the interior of the interval ranges from 150 to 1200 ppi. In our proof-of-concept work, we stick to the current digitization resolution, i.e., 300 ppi,

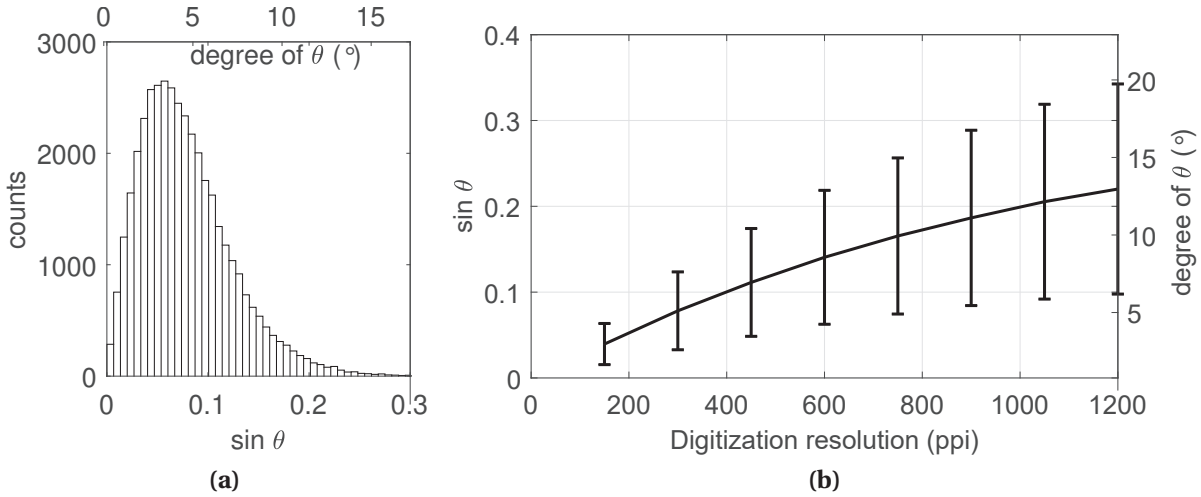


Figure 3.16: (a) Histogram of the orientation of squared area covered by a working pixel when a paper patch of size $\frac{2}{3}$-by-$\frac{2}{3}$ inch^2 is digitized to 200-by-200 working pixels or 300 ppi. (b) The averaged orientation as a function of digitization resolution. Error bars correspond to one sample standard deviation above and below the average. The monotonic smoothly increasing curve does not strongly justify the use of a particular resolution among others within the interior of [150, 1200] ppi.

so that the resolution is adequate for authentication while keeping the computational complexity at a reasonable level.

3.6.3 Impact of Spatial Registration Error

In this subsection, we investigate the performance drop due to the error of spatial registration for the paper patch. Clarkson et al. [6] applied a lowpass filter to the extracted image and downsampled it to reduce the impact of the registration error, but its effect was not explicitly studied. Fig. 3.17 shows an image of a piece of paper scanned by a flatbed scanner, which shows the design of a registration pattern we used in this work. The square patch to the left of the QR code patch is the area of interest that we use for paper surface-based authentication. To locate the position of the area of interest, we need to estimate the positions of the intersections. We first use a Hough transform to find the lines and then the intersections. We then refine the estimations for the positions of intersections by finding the centers of the circles. The intersections of the lines in the paper patch are printed in the center of respective circles.

In the real-world application, the estimations for the positions of the paper patch may be inaccurate, and as a result, the performance will drop. To investigate the effect of imprecise

Figure 3.17: The design of a registration pattern used in this work. The image was captured by a flatbed scanner. The square patch on the left of the QR code is the area used by authentication. By detecting the QR code, the location of the pattern in the image can be roughly estimated, then the precise location is estimated using the lines and circles. Also, the QR code can be used to store information such as paper ID and the reference feature.

estimations for the positions, we perturbate the estimated locations of the four corner positions of the paper patch. For each of the estimated corner locations (x, y), we add some noise to it, namely, $x' = x + e_1$, $y' = y + e_2$, where $e_1, e_2 \sim \mathcal{N}(0, L^2)$, L is standard deviation to indicate the level of perturbation strength. We follow the procedure in the practical authentication system in Section 3.5.3 while adding perturbations to the estimated corner positions in each scanned image of the paper patch. We increase the perturbation strength L and calculate the EER at each perturbation strength level, and plot the results in Fig. 3.18. When the perturbation strength is small, within 0.3 pixels, the EERs do not change much. This may be due to the fact that the estimated corner positions of the paper patch were not very accurate in the first place, thus adding small perturbations did not result in much performance drop. As the perturbation strength increased beyond 0.4 pixels, the EERs will increase significantly, indicating that deploying a precise image alignment algorithm is one important factor to achieving satisfactory authentication performance.

3.7 Conclusion

In this work, we have shown by analytic derivations that the specular component of light reflection does not play a role in the estimation of the norm map of paper surfaces in the unique optical setup of a flatbed scanner. We used a larger dataset to confirm that flatbed scanners can capture meaningful physical quantities of paper surfaces, and we investigated the blurring effect due to the scanner. We have shown that the high frequency subbands of the reconstructed surfaces are better discriminative features than the norm map, which we verified in a practical engineering system that uses flatbed scanners. We

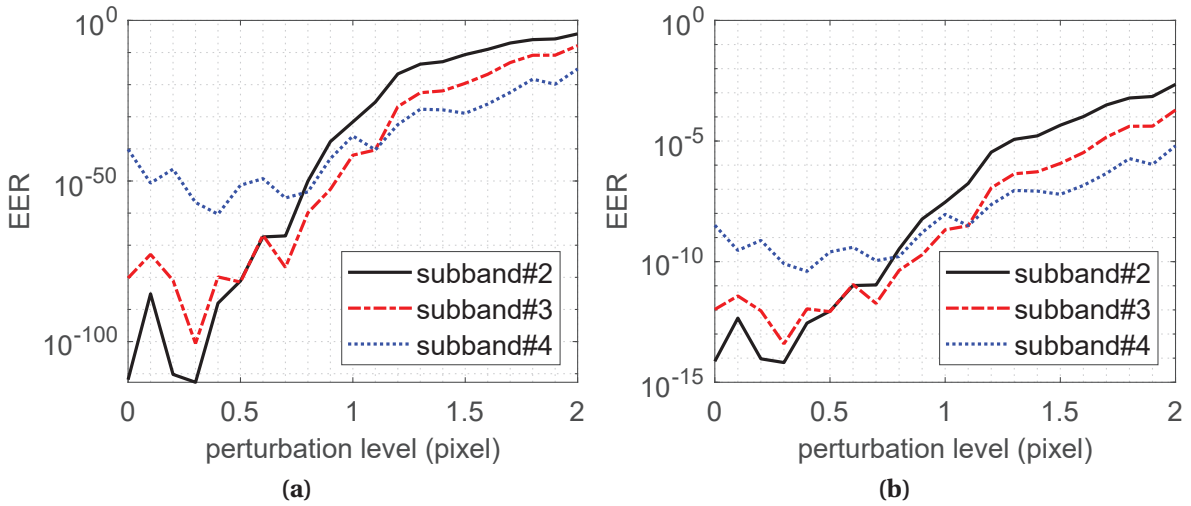


Figure 3.18: The impact of spatial registration error: EERs against the perturbation strength L when assuming (a) Gaussian and (b) Laplace distributions. The length of a pixel edge is $\frac{1}{300}$ inches. When there is more registration error (or larger perturbation), the discriminative performance is significantly lowered.

have shown that larger paper patches will yield better authentication performance in EER, and a precise image alignment algorithm is important for achieving satisfactory authentication performance. s The flatbed scanners instead of mobile cameras have been used as the acquisition device for the studies in this work. Although flatbed scanners are less flexible in terms of portability and acquiring images of objects with irregular shapes such as wine bottles, they have a better-controlled experimental setup. This allows easier investigations into paper surface-based authentication, which is hard to achieve when using mobile cameras in designed experiments. The findings in this work using flatbed scanners may give us insights into how to study research questions for the mobile camera-based authentication system, which are more challenging due to the lower signal-to-noise ratios. In future work, we plan to investigate key research questions on using mobile cameras to acquire the microstructure, e.g., how the specular reflection can be taken into consideration to improve the estimation accuracy of the norm map.

3.8 Appendix

We will provide justification for the relation (3.14) between the sample correlation coefficient of a block, ρ, and the sample correlation coefficients of its nonoverlapping, equal-sized

subblocks, $\{\rho_i\}_{i=1}^4$, namely, $\rho \approx 1/4 \sum_{i=1}^4 \rho_i$. We will argue in the finite-sample case that the residual $r_n = \rho - \frac{1}{4}\sum_{i=1}^4 \rho_i \approx 0$. We will also prove that in the asymptotic case $|r_n|$ converges to 0 in probability.

We denote, for ith subblock, the raw data $\mathbf{x}_i^r = (x_{i1}, x_{i2}, \ldots, x_{in})$, the sample mean $x_{i\cdot} = \frac{1}{n}\sum_{j=1}^n x_{ij}$, and the mean-removed data $\mathbf{x}_i = \mathbf{x}_i^r - x_{i\cdot}$, where $i \in \{1,2,3,4\}$. The mean-removed data for the parent block can be represented as follows:

$$\mathbf{x} \overset{(a)}{=} \begin{bmatrix} \mathbf{x}_1^r \\ \vdots \\ \mathbf{x}_4^r \end{bmatrix} - \frac{1}{4}\sum_{i=1}^4 x_{i\cdot} \overset{(b)}{=} \begin{bmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_4 \end{bmatrix} + \begin{bmatrix} \epsilon_1 \mathbb{1} \\ \vdots \\ \epsilon_4 \mathbb{1} \end{bmatrix} \overset{(c)}{=} \mathbf{x}' + \epsilon, \tag{3.17}$$

where $\mathbb{1}$ is length-n vector of all ones, and $\epsilon_i = \frac{1}{4}(3x_{i\cdot} - \sum_{i' \neq i} x_{i'\cdot})$ is a perturbation term. Here, (3.17a) and (3.17c) is by definition. (3.17b) connects the mean-removed terms $\mathbf{x}$ and $\{\mathbf{x}_i\}_{i=1}^4$ at two scales. With the definitions of $\mathbf{x}$ and $\{\mathbf{x}_i\}_{i=1}^4$, ρ and ρ_i defined in the caption of Fig. 3.15 can be rewritten as:

$$\rho = \frac{\mathbf{x}^T\mathbf{y}}{\|\mathbf{x}\|\|\mathbf{y}\|}, \quad \rho_i = \frac{\mathbf{x}_i^T\mathbf{y}_i}{\|\mathbf{x}_i\|\|\mathbf{y}_i\|}, \quad i = 1, \cdots, 4. \tag{3.18}$$

Finite-sample approximation For a finite sample size, we justify the following relationship by showing perturbation terms are close to zero and $\|\mathbf{x}_i\|$s are close to $\|\mathbf{x}\|/2$:

$$r_n = \sum_{i=1}^4 \mathbf{x}_i^T\mathbf{y}_i \left(\frac{1}{\|\mathbf{x}\|\|\mathbf{y}\|} - \frac{1}{4\|\mathbf{x}_i\|\left\|\mathbf{y}_i\right\|} \right) + \left[\mathbb{1}^T\left(\epsilon_i\mathbf{x}_i + \varepsilon_i\mathbf{y}_i\right) + n\epsilon_i\varepsilon_i\right] / \|\mathbf{x}\|\|\mathbf{y}\| \approx 0. \tag{3.19}$$

Assume that x_{ij}'s are independent and identically distributed with mean value μ and variance σ^2. Note that $\|\mathbf{x}_i\|^2 = \sum_{j=1}^n x_{ij}^2 - n x_{i\cdot}^2$. It is easy to show using the strong law of large number that $\|\mathbf{x}_i\|^2$ converges to $n\sigma^2$ almost surely, and $\|\mathbf{x}\|^2$ and converges to $4n\sigma^2$ almost surely. Hence, the term in the parentheses of (19) is close to zero. Both perturbation terms ϵ_i and $\varepsilon_i \sim \mathcal{N}(0, 0.75\sigma^2/n)$ are zero mean with very tiny variance for large n, e.g., $n = 10000$ in our application scenario. Hence, the term in the brackets is also close to zero. We also used real data to verify that $r_n \approx 0$. We followed the procedures in Section 3.6.1 to cut the subbands into four subblocks and calculate the sample correlation values ρ and ρ_i, $i = 1, \cdots, 4$. Under the matched case, i.e., the population correlation is larger than zero, the sample mean and standard deviation of r_n for subbands #2–#4 were around 10^{-5} and 10^{-3}, respectively. Under the unmatched case, i.e., the population correlation is zero, the sample mean and standard deviation of r_n for subbands #2–#4 were around 10^{-4} and 10^{-3},

respectively. The small residuals confirmed that $r_n \approx 0$ for the finite-sample scenario.

Lemma 1.[30] When population correlation value ρ_t of a bivariate Gaussian pair is nonzero, the expectation and variance of sample correlation value ρ can be expressed in the form of series:

$$\begin{aligned} \mathbb{E}[\rho] &= \rho_t - \frac{\rho_t\left(1-\rho_t^2\right)}{2(n-1)} + \cdots, \\ \mathrm{Var}(\rho) &= \frac{\left(1-\rho_t^2\right)^2}{n-1}\left[1 + \frac{11\rho_t^2}{2(n-1)} + \cdots\right], \end{aligned} \tag{3.20}$$

where n is the sample size.

Convergence in mean Denote the population correlation value to be ρ_t. The sample size is $4n$ for the block and n for a subblock. From Lemma 1, we have:

$$\mathbb{E}\left[\rho - \frac{1}{4}\sum_{i=1}^{4}\rho_i\right] = \left(\rho_t - \frac{\rho_t\left(1-\rho_t^2\right)}{2\cdot(4n-1)} + \cdots\right) - \frac{1}{4}\sum_{i=1}^{4}\left(\rho_t - \frac{\rho_t\left(1-\rho_t^2\right)}{2(n-1)} + \cdots\right) \to 0 \tag{3.21}$$

as $n \to \infty$.

Convergence in probability For a sample correlation ρ in a block, from Lemma 1 and Markov's inequality we can derive:

$$\mathbb{P}(|\rho-\rho_t| > \varepsilon) \le \left(\mathrm{Var}(\rho) + (\mathbb{E}(\rho)-\rho_t)^2\right)/\varepsilon^2 = \frac{1}{\varepsilon^2}\cdot\left[\frac{\left(1-\rho_t^2\right)^2}{4n-1}\left(1 + \frac{11\rho_t^2}{2(4n-1)} + \cdots\right) + \left(\frac{\rho_t\left(1-\rho_t^2\right)}{2(4n-1)} + \cdots\right)^2\right], \tag{3.22}$$

which is easy to show that $\mathbb{P}[|\rho-\rho_t| > \varepsilon] \to 0$ and hence ρ converges to ρ_t in probability, or in a slightly different form $|\rho-\rho_{\mathrm{t}}| \xrightarrow{\mathrm{p}} 0$. Similarly, $|\rho_i-\rho_{\mathrm{t}}| \xrightarrow{\mathrm{p}} 0$. From triangle inequality, we have $|\rho_i-\rho| \le |\rho_i-\rho_{\mathrm{t}}| + |\rho_{\mathrm{t}}-\rho| \xrightarrow{\mathrm{p}} 0$. Applying triangle inequality again, we conclude the proof: $|r_n| = \frac{1}{4}\left|\sum_{i=1}^{4}(\rho_i-\rho)\right| \le \frac{1}{4}\sum_{i=1}^{4}|\rho_i-\rho| \xrightarrow{\mathrm{p}} 0$.

3.9 Reconstructed Heightmap Leads to Higher Correlation

We follow [22] to reconstruct heightmaps with normal vector fields generated from scanners and a confocal microscope using shapelets [28] that can be considered as a robust integration algorithm. The diagrams for generating the heightmaps/3D surfaces for test and reference patches are shown in Figs. 3.10(a) and (b), respectively, excluding the last blocks. The images for the test patch are acquired by scanners and the heightmap for the reference patch is measured by a confocal microscope. We correlated the reconstructed heightmaps between scanner and confocal microscope, obtaining the correlation at 0.358 as shown in Table 3.1, which is higher than the correlation at 0.357 or 0.301 using the norm map as the feature. The improved correlation values indicate that the heightmap with integrated information in both x- and y-directions is a better discriminative feature than the norm map.

Fig. 3.19(a) shows a reconstructed heightmap from images acquired by a scanner. It is observed that the right part of the paper patch has a higher elevation than the left part. This may be caused by the nonflat shape of the paper when scanned that is not a stable characteristic and may change every time the paper is handled. The global trend due to the nonflat shape is also problematic from the perspective of the similarity measure using correlation coefficient: i) if two surfaces have similar trends, the correlation between the two surfaces will be high even if their local structures are very different; ii) if trends are different, correlation will be low even if their local structures are similar. Hence, the trend of the heightmaps must be removed before correlation is calculated.

We removed the trend of the heightmap in Fig. 3.19(a) and a detrended version is shown in Fig. 3.19(b). The detrending process contains two steps. First, a Gaussian blur was applied to generate a surface capturing the overall trend of the heightmap but not capturing the local structures. In the experiments of this paper in which $\frac{2}{3}$-by-$\frac{2}{3}$ inch^2 patches are digitized to 200-by-200 pixels, a standard deviation of 25 pixels was a reasonable value. Second, the trend surface was subtracted from the heightmap to generate the detrended heightmap.

The correlation resulting from using the detrended heightmap is 0.499, which is a further improvement over 0.358 resulting from using the raw heightmap. This result is consistent with that reported in [22] that studied cameras as acquisition devices. Note that the detrended surface retains the middle to high spatial-frequency contents of the raw heightmap that corresponds to local structures, since the trend surface containing the low frequency contents was removed.

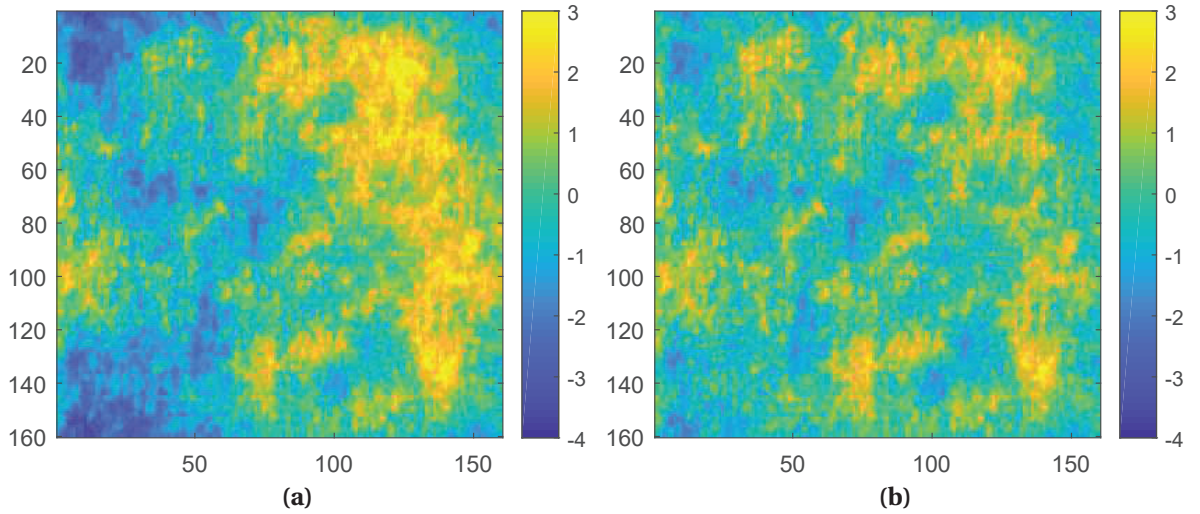


Figure 3.19: (a) Reconstructed heightmap from a norm map estimated from images acquired by a scanner, and (b) a detrended version of (a). The detrended heightmap is more flat and local peaks and valleys are more visible.

3.10 Discrimination Using Subbands of Heightmap

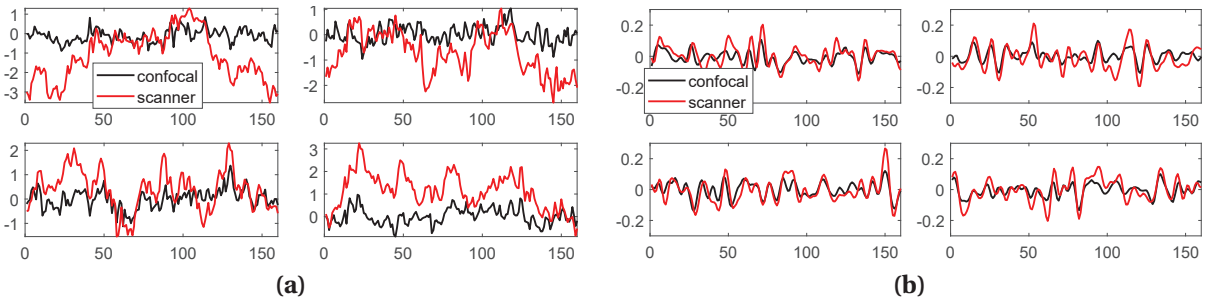


Figure 3.20: Representative slices in x direction from (a) original heightmap and (b) SUBBAND#3. The slices in the heightmaps of scanner have trends. The peaks in the high-spatial frequency subbands overlap much better than in the original heightmaps.

The diagrams for generating the subbands of heightmaps/3D surfaces for test and reference patches are shown in Figs. 3.10(a) and (b), respectively, including the last blocks. We decompose the reconstructed heightmap into ten spatial subbands corresponding to a DoG representation. We plot representative slices of the original heightmap in Fig. 3.20(a) and the third-highest subband, i.e., SUBBAND#3 in Fig. 3.20(b).

Fig. 3.20(a) reveals the trends in the reconstructed surface from scanners. Fig. 3.20(b)

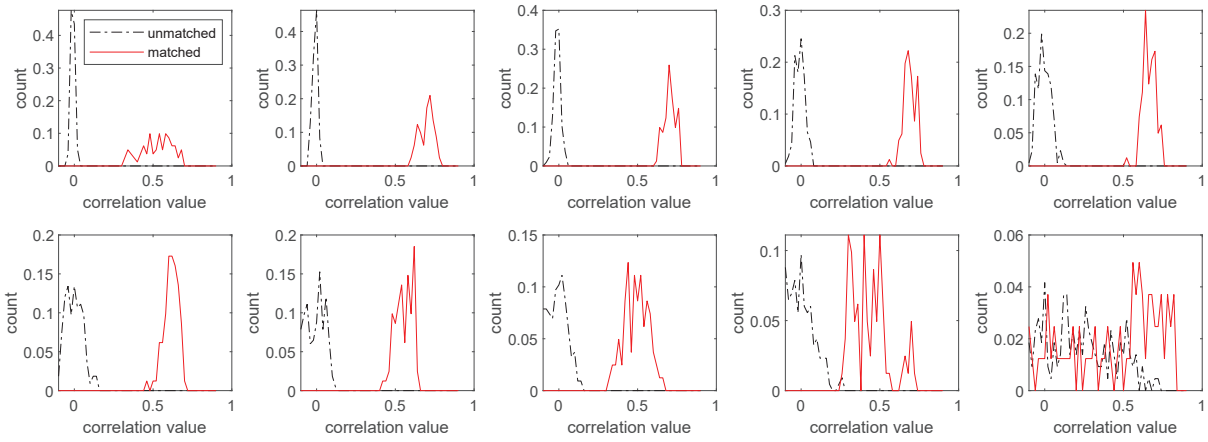


Figure 3.21: Distributions of correlation values for matched cases and unmatched cases at different subbands. The second and third-highest spatial-frequency subbands are more powerful in describing the uniqueness of physical surfaces.

shows that the high-spatial frequency subbands from the scanner and confocal microscope match well with each other. We calculated the correlation when the scanner matches (H_0) or does not match (H_1) the confocal microscope for every subband. The distributions of correlation values for each subband is shown in Fig. 3.21. The distances of the distributions for matched and unmatched in high spatial-frequency subbands are far, indicating a good discriminative capability. The averaged correlation for best performing subband, i.e., the third-highest spatial-frequency subband, is 0.714 as shown in Table 3.1.

We quantitatively evaluate the discriminative performance of each spatial subband of the heightmap. For the majority of them, i.e., Subbands #1 to #8, the empirical distributions for the two hypotheses do not have overlap as shown in Fig. 3.21. This also poses a difficulty in estimating discrimination quantities such as the probability of false alarm or miss when the threshold used is in the middle of two distributions. This issue is caused by the fact that the overlapping tails are too tiny. We follow the procedure laid out in [21] to obtain the maximum likelihood estimator (MLE) of the EER using summary statistic quantities of each hypothesis. Since the EER is achieved when both false-alarm and miss rates are small and equal, the characteristics of extrapolated tails affect the final result significantly. Since there are not enough data for determining the behaviors of the tails, we use a light-tailed distribution, Gaussian, and a heavy-tailed distribution, Laplacian, to quantify the EER in an optimistic way and a pessimistic way, respectively. It is not difficult to show that when correlation is assumed to be Gaussian and Laplacian and using a simple thresholding rule,

the EER can be written as

$$\text{EER} = \Phi[(\mu_0 - \mu_1)/(\sigma_0 + \sigma_1)], \tag{3.23a}$$

$$\text{EER} = \frac{1}{2}\exp[(\mu_0 - \mu_1) \cdot \lambda_0\lambda_1/(\lambda_0 + \lambda_1)], \tag{3.23b}$$

respectively, where $\Phi(\cdot)$ is the cumulative density function for the standard Gaussian distribution, and μ_i and σ_i, $i = 0, 1$ are mean and standard deviation for the ith hypotheis. By the invariance principle[31], we could substitute MLE estimates for μ_i and σ_i into the above equations to obtain the MLE for the EER.

The estimated EER as a function of subband index is shown in Fig. 3.11. It is revealed that the third-highest spatial-frequency subband is the most discriminative, achieving an EER at 10^{-36} or 10^{-8} under the Gaussian or Laplacian tail extrapolation assumption. We also compare the performance of subbands of heightmap to that of other physical features, i.e., norm map and detrended heightmap, as shown by horizontal lines in Fig. 3.11. Their EERs are much worse than using the third-highest spatial-frequency subband.

CHAPTER

4

SURFACE-BASED AUTHENTICATION SYSTEM FOR INTEGRATED CIRCUIT CHIPS

4.1 Introduction

The semiconductors industry has been developing fast in the past few decades and electronic devices have become increasingly common. Counterfeiting for the integrated circuits (ICs) has become a major challenge. The use of counterfeit ICs poses threats to multiple sectors with heavy deployment of electronic systems, such as public health, banking, and military defense industries. According to the Semiconductor Industry Association, counterfeiting costs US-based semiconductor companies more than $7.5 billion per year and nearly 11,000 jobs [32]. Recently, the shortage of supply of IC chips forces supply-chain participants to purchase IC chips from untrustworthy sources, increasing the risk of acquiring counterfeit IC chips. The reconstructing of the distributed supply chain of IC chips may take a long time. Therefore, developing effective and efficient anti-counterfeiting techniques for

IC chips has become increasingly important.

Electronic physically unclonable functions (PUFs) have been used for anti-counterfeiting for ICs. Due to the manufacturing variations of ICs, the electronic measurements of each device, such as voltages, resistance, digital time delays, and power-up states of cells are unique and unpredictable [33]. Such manufacturing variations are impossible to duplicate. However, electronic PUFs require the ICs to be put into working status to obtain the measurements, which usually needs trained personnel to operate them in a working laboratory environment. In the real-world scenario, such as in a supply chain, getting measurements conveniently is desired. The electronic PUFs have also been shown to be sensitive to aging and environmental variations, such as thermal noise and power supply noise [34]. Such disadvantages of electronic PUFs make them less effective in real-world applications for anti-counterfeiting for ICs.

The surfaces of objects are random and uneven due to their unique microstructures, which can be regarded as the "fingerprint". Optical PUF was first proposed to identify 3D structures with laser speckles, which is determined by the microstructure of the object [35]. Later, optical PUFs have been successfully used for the identification of paper surfaces [4, 6, 21, 36]. Since the surface of IC chips are also random like paper surfaces due to the manufacturing variations, we hypothesize that optical PUFs can also be used for IC identification. The optical response of the IC chip surface will not be affected by temperature and power supply noise and the acquisition of the optical response is fast when using optical devices such as a camera, which is an advantage for verification in the supply-chain applications. Also, the development of computer vision technology and the photo acquisition devices make it convenient to exploit optical PUFs for IC identification. The challenges, such as registering captured images of chip surfaces, need to be addressed when using optical PUF for IC chip identification.

In this work, we propose to use the optical PUF for the identification of IC chips, where we obtain photos of the IC chip surface with a mobile camera or a flatbed scanner to obtain physical features of chip surfaces. To the best of our knowledge, this is the first work to use optical PUF for IC authentication. This work using optical PUF for IC authentication contains two parts. In the first part, we investigate whether the images captured with flatbed scanner can capture meaningful physical features of IC surfaces by comparing with measurements from a confocal microscope. In the second part, we use the physical features of the IC chip surface from the camera/scanner-captured images for IC surface authentication. To make the optical PUF more applicable in the real world, we also explore fast verification schemes using videos captured by a camera.

We summarize the contributions of this work as follows:

- we confirm experimentally that cameras and scanners can capture meaningful physical features of IC chip surfaces;
- we address new challenges such as image alignment when re-purposing the surface-based authentication system for IC chips from the paper surfaces;
- we propose effective and lightweight verification schemes for IC chips by using camera-captured videos or the estimated specular reflection locations.

4.2 Related Work

4.2.1 Electronic PUFs for ICs

Electronic PUFs, based on analog measurements, digital delays, and states in memory cells, have been used for IC identification. Lofstrom et al. [37] exploited the effect of manufacturing variations on threshold voltages of transistors for IC identification. Helinski et al. [38] proposed using chip's resistance variations in the power grid as a PUF, such variations are also caused by manufacturing variations. Besides the analog measurements of physical parameters, the PUFs based on the digital delay measurements of the IC chips have been proposed. Gassend et al. [39] proposed arbiter PUF, which exploits statistical variations in the delays of devices and wires within the IC. Lee et al. [40] exploited statistical delay variations of wires and transistors across ICs to build a secret key for IC identification. Patel et al. [41] proposed a PUF based on the number of variability-dependent glitches on the output of a combinational multiplier. There have also been PUFs based on destabilized memory cell. Holcomb et al. [42] used the initial states of static random-access memory (SRAM) as a physical fingerprint. Besides of using SRAM cells, other storage elements have also been used for IC authentication. Yamamoto et al. [43] proposed a novel PUF structure based on a Butterfly PUF with multiple reset-set (RS) latches. Maes et al. [44] proposed using flip-flop powerup values for reconfigurable devices. Simons et al. [45] introduced a new type of PUF based on the buskeeper cells. However, these methods will be largely affected by factors such as temperature and the need to put devices into electronic operating condition for verification.

4.2.2 Optical PUF for Paper Surfaces

Optical PUFs have been used for paper surface identification for about two decades. The methods using optical PUFs can be categorized as optical feature approach and the physical feature approach. The optical feature approach exploits the visual appearance of paper surface under the light. As a proof-of-concept effort using optical PUF for paper authentication, Buchanan et al. [4] scanned the paper surface with laser beam and compared the reflected laser speckle for paper authentication. Instead of using an expensive laser system, Beekhof et al. [5] used macrolens-aided mobile phones to capture images, and minimum reference distance decoding and reference list decoding were used for identification. Sharma et al. [23] used a camera with the aid of a microscope with a built-in LED to capture paper speckles as the fingerprint for paper surface. Toreini et al. [24] captured optical features of paper texture using the light transmitting through the paper instead of reflecting from paper surface for paper authentication. However, this approach is not applicable for IC chip authentication because there will not be transmissive lights for opaque IC chip packages.

The physical feature approach have been shown to possess better authentication performance for paper authentication than the optical feature approach [6, 21]. The orientations of the microscopic surfaces of a paper patch, which can be quantified in terms of norm map, have been used for paper authentication. The norm map is the projection of uniformly spaced surface normals onto the xy-plane. Clarkson et al. [6] used a flatbed scanner to scan paper surfaces and derived the norm map of a paper surface for paper identification. Instead of using a bulky scanner, Wong and Wu [21] used mobile cameras to obtain the norm map of paper surface by taking photos of the paper patch in several different camera directions. The estimated norm map was also verified against the ground truth, which is the norm map obtained from a confocal microscope. In [22], the high-frequency components of the subbands of the reconstructed 3D surface of paper patch was for paper authentication and achieved better performance than using the norm map. The success of optical PUF in paper surface identification suggests the possibility of IC surface identification using optical PUF. In this work, we aim to exploit the physical features using camera-capture images for IC chip authentication.

4.2.3 Encapsulation for IC Chips

Epoxy-molding compound (EMC) material has been used to encapsulate and protect the IC structures from moisture, ionic contamination, and thermal and mechanical threats

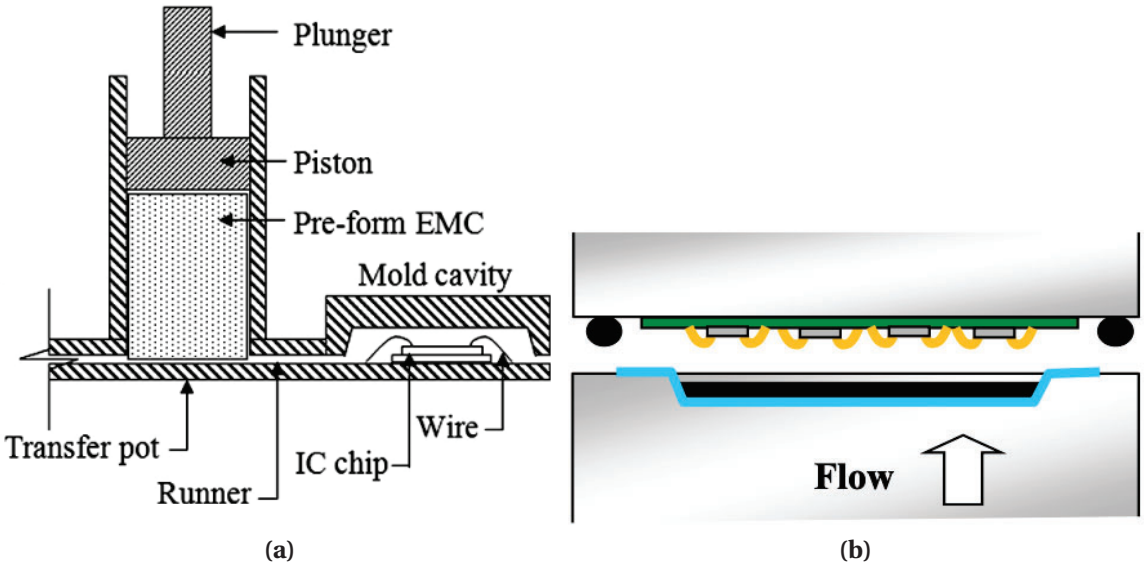


Figure 4.1: (a) An illustration of the transfer molding encapsulation setup [1]. The pre-formed EMC will be transferred from the transfer pot via the runner to the mold cavity. The IC chip will be encapsulated and protected by the EMC. (b) An illustration of compression molding [2]. The mold with compound will be closed by applying required pressure, with a vacuum to suck up air, gas and moisture coming out from the compound.

[1]. There are two main methods for IC encapsulation, transfer molding and compression molding. Transfer molding, illustrated in Fig. 4.1(a) [1], is the most commonly used encapsulation method. In transfer molding, liquefied epoxy resin runs through a mold cavity. In the encapsulation stage, random unevenness of the IC chip surfaces can form due to the flow mark, a rough mold cavity surface, and existing air traps [46]. Also, in the curing process, the liquefied epoxy resin will be hardened, and the surface will also become random in this process. Such unevenness can result in physical unclonable features for IC chip authentication. However, transfer molding requires high injection pressure of resin flow to fill in the cavity, and it has recently faced challenges when molding advanced packages with limited space for resin flow and longer and finer wires [2]. TOWA Corporation proposed compression molding, illustrated in Fig. 4.1(b) [2], to address such challenges. In compression molding, mold with granular compound will be completely closed by applying required pressure. When using the compression molding, a surface will become random in the curing process.

4.3 Investigating Physical Features for IC Chips

4.3.1 Feasibility of Capturing Microstructures of IC Chip Surfaces Using Consumer Grade Imaging Devices

In this subsection, we explore the possibility to derive meaningful physical features using flatbed scanners. We compare the norm maps estimated from scanner-captured images and those obtained from the confocal microscope, which can be regarded as ground truth. We used a CanoScan Lide 300 flatbed scanner to scan the images of the chip surface, and followed the procedure in [36] to obtain the norm map of the chip surface. We scanned IC chip surfaces in opposite directions, and took the differences of the images scanned in opposite directions as in 4.3 to obtain x- and y-components of norm maps. The resolution of the scanner is 600 ppi, the the pixel edge length is 42.33 μm. We used a Keyence VKx1100 confocal microscope to measure the heightmap of a chip surface, which can be used to derive the ground truth norm map. The pixel edge length is 5.37 μm. Since the confocal microscope can accurately measure the height of IC chip surface, and we regard the derived norm map as the ground truth.

We followed the procedures in previous work [6, 36] using scanners to derive the norm map of paper surfaces based on the fully diffuse model and photometric stereo setup. Under the fully diffuse model, the perceived intensity l_r at location $\mathbf{p}$ is modeled as follows [21, 14, 6]:

$$l_r(\mathbf{p}) = \lambda \cdot l(\mathbf{p}) \cdot [\mathbf{n}(\mathbf{p})^T \mathbf{v}(\mathbf{p})]^+, \tag{4.1}$$

where $x^+ = \max(0, x)$, $\mathbf{n} = (n_x, n_y, n_z)$ is the normal direction of the paper surface at the microscopic level, $\mathbf{v}(\mathbf{p})$ is the incident light direction arriving at $\mathbf{p}$, $l(\mathbf{p})$ is the strength of the light and the surface albedo λ characterizes the physical capability of reflecting the light. The pixel value at location $\mathbf{p}$ in the image of a scanner could be expressed by the integral over all light diffusely reflected off that surface point and originating from points $\mathbf{v}(\mathbf{p}) = (o_x, o_y, o_z)^T$ along the linear light in the x-direction [6]:

$$I_{0^\circ} = \int_{x_1}^{x_2} l_r \, d\, o_x = l \cdot w_d \int_{x_1}^{x_2} \mathbf{n}^T(o_x, o_y, o_z) d\, o_x \tag{4.2}$$

If the scanner scan the paper in two opposite directions and get two images $I_{0^\circ}(\mathbf{p})$ and

$I_{180^\circ}(\mathbf{p})$, $I_{0^\circ}(\mathbf{p}) - I_{180^\circ}(\mathbf{p})$ could be estimation for $n_x(\mathbf{p})$ or $n_y(\mathbf{p})$ [6]:

$$\begin{aligned} I_{0^\circ} - I_{180^\circ} &= l \cdot w_d \int_{x_1}^{x_2} \mathbf{n}^T [(x, o_y, o_z) - (x, -o_y, o_z)] d\, o_x \\ &= n_y \cdot s \end{aligned} \tag{4.3}$$

where $s = 2l\, w_d\, o_y (x_2 - x_1)$. It has been theoretically shown [36] that, under the generalized reflection model, the difference in (4.3) is still proportional to the y-component n_y.

We selected a part of scanner's norm map, with the size of 70×70 pixels, in the background of the chip surface (the part without texts), denoted as **S**. We investigated the background instead of the text areas because the same text on different chips may have similar engraved strokes on the chip surfaces, and therefore similar norm maps, which is a false positive case. To compare the norm maps obtained from scanner and confocal microscope, we upsampled the confocal height map to obtain **H**, such that the edge length of confocal heightmap is 2.65 μm, i.e., 1/16 of that of the scanner's pixel edge length. In the heightmap measured confocal microscope **H**, we applied the idea of crosscorrelation to search for an area $\mathbf{H}_0$ of the same physical size as **S**, such that the norm map **C** obtained from $\mathbf{H}_0$ matches **S** best in terms of correlation. We followed the procedure in [21] to derive the norm map **C** of size 70×70 from heightmap $\mathbf{H}_0$. When **S** and **H** are from the same chip surface, the correlation between **C** and **S** should be high. We chose three different background areas of size 70×70, and the correlations reach to 0.53, 0.53, and 0.54. An example of the estimated norm map **S**, confocal heightmap $\mathbf{H}_0$, and confocal norm map **C** is shown in Fig. 4.2. When **S** and **H** are from two different chip surfaces, the correlations between **C** and **S** are only 0.04, 0.04, and 0.03. The results show that consumer grade imaging devices can capture meaningful microstructures of the chip surface. We use scanner to acquire the images instead of using cameras because scanners can provide better-controlled experimental setup and therefore a larger signal-to-noise ratio.

4.3.2 Ubiquitous Capturing of IC Surface PUF Using Mobile Cameras

Scanners can be bulky and not available to everyone, while cameras are much more widely available. In this work, we investigate the potential of using a mobile camera to capture the physical features. Since flatbed scanners have been shown to be able to capture microstructures of IC chip surfaces, we use a camera and mimic the scanner case when we capture the images of IC chip surfaces, as shown in Fig. 4.3. We set a camera of iPad Pro

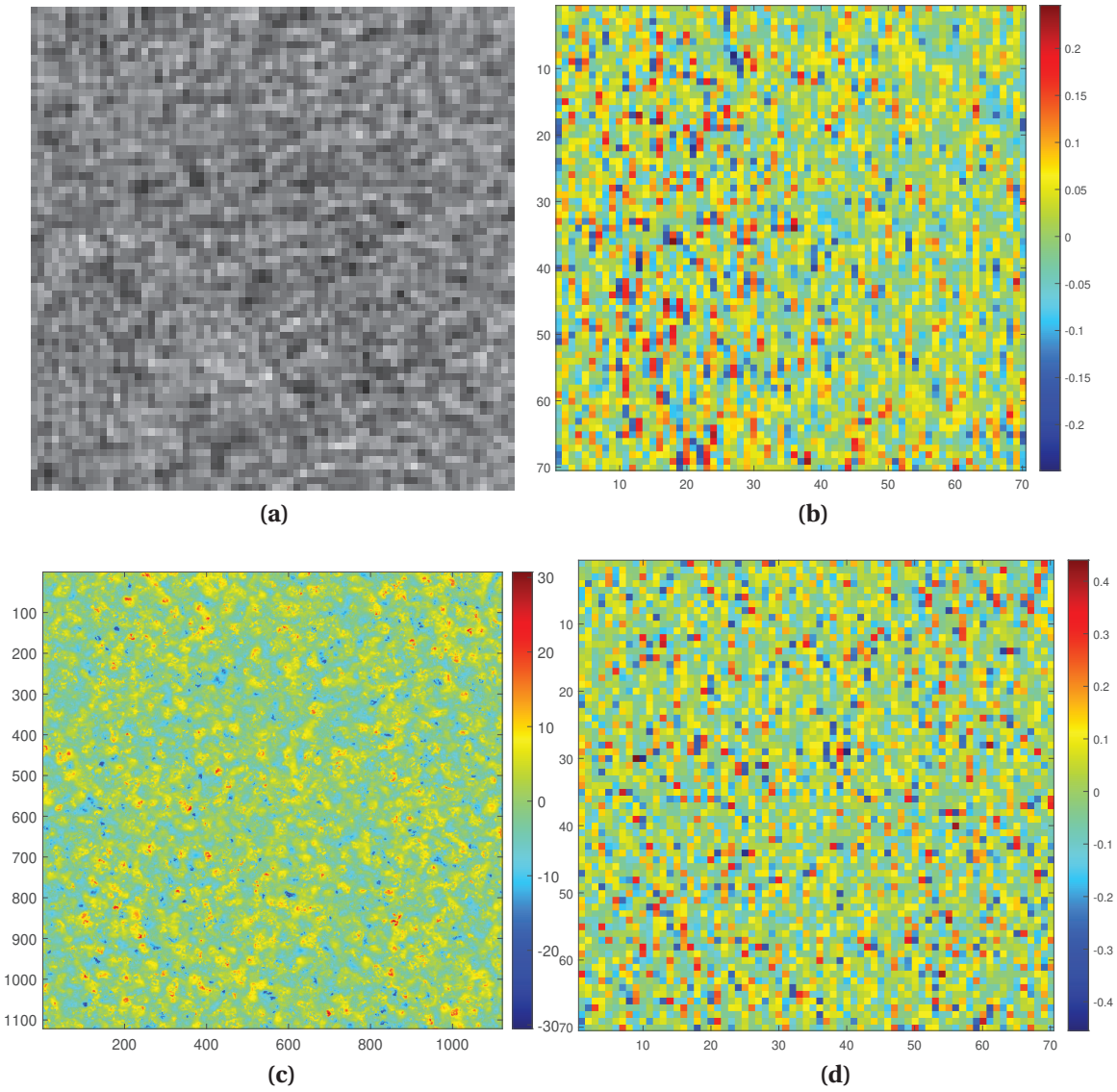


Figure 4.2: (a) A scanner-captured image (after contrast enhancement) of an area in the background of IC chip surface. (b) The estimated x-component of norm map $\mathbf{S}$ from the scanner. (c) The height map $\mathbf{H}_0$ measured by confocal microscope and (d) the derived norm map $\mathbf{C}$ from $\mathbf{H}_0$.

2018 directly above the IC chip surface. Denote the center of IC chip surface to be the origin point, we took images when a lamp light was rotated to be placed at 0°, 90°, 180°, or 270°. The lamp was placed such that the incident light direction is about 45° from the z-axis. At each direction three images were taken, and we captured images for four IC chips.

When capturing images with a camera, the light source should be close to the IC chip to ensure a strong enough reflected light. However, the incident light is not parallel light

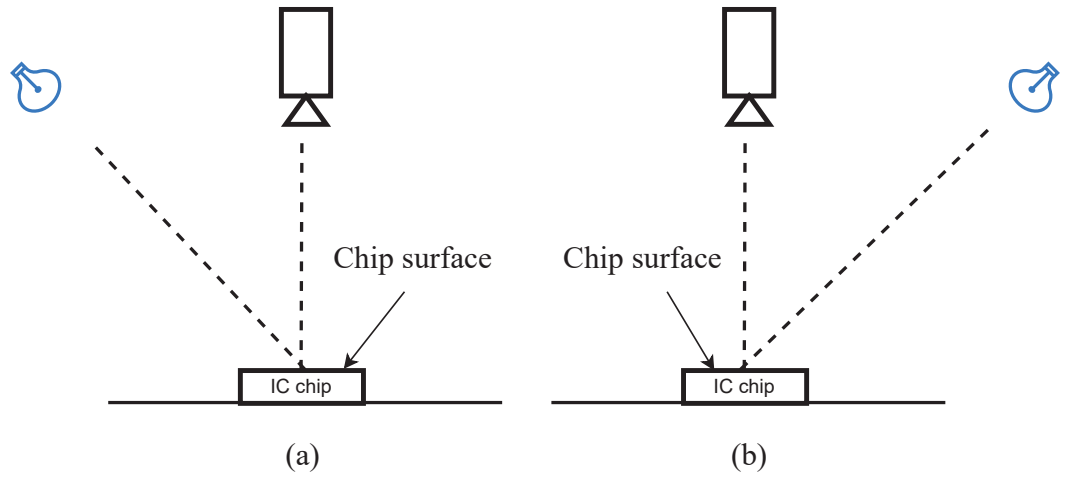


Figure 4.3: A front view when using a camera to capture images of IC chip surface with light source placed at (a) 0° and (b) 180°. Taking the difference of the two captured images results in a scaled version of x-component of norm map.

and the light sources are not exactly at opposite directions for each pixel on the IC chip surface. Taking the difference of images captured in opposite directions will result in an inaccurate norm map estimation. We studied the effect of distance between IC chip and light source in 4.6, and found that a distance of 40 cm can produce negligible error.

We take the difference of images captured in opposite directions to derive the norm map of chip surfaces. Before taking difference of images, we first need to register the captured images of IC chip surfaces to estimate the norm maps. To register an image, an intuitive way is to compare the pixel values between the image and the template. In the chip surface case, it is possible to use the printed letters on the chip surfaces. However, under the resolution of a scanner, the scale of the microstructures are much smaller than the pixel size, and such methods are sensitive to the noise as well as scaling of images. To show that misalignment of a single pixel in scanner's image will cause the authentication to fail, we examine the spatial autocorrelation of IC chip surface by shifting a heightmap of size 8.78 mm^2 on the background of chip surface measured by the confocal microscope. We then calculated the autocorrelation between the heightmaps before and after shifting, as shown in Fig. 4.4. A pixel length in scanner equals about eight confocal pixels. The correlation drops faster for the IC chip surface than the paper surface, indicating that more accurate registration for IC chip surface is needed and previous method [21] estimating locations of key points to register paper patches may not be accurate enough for IC chip surfaces. In this work, we propose to register the captured test images of IC chips using phase correlation [47]. First,

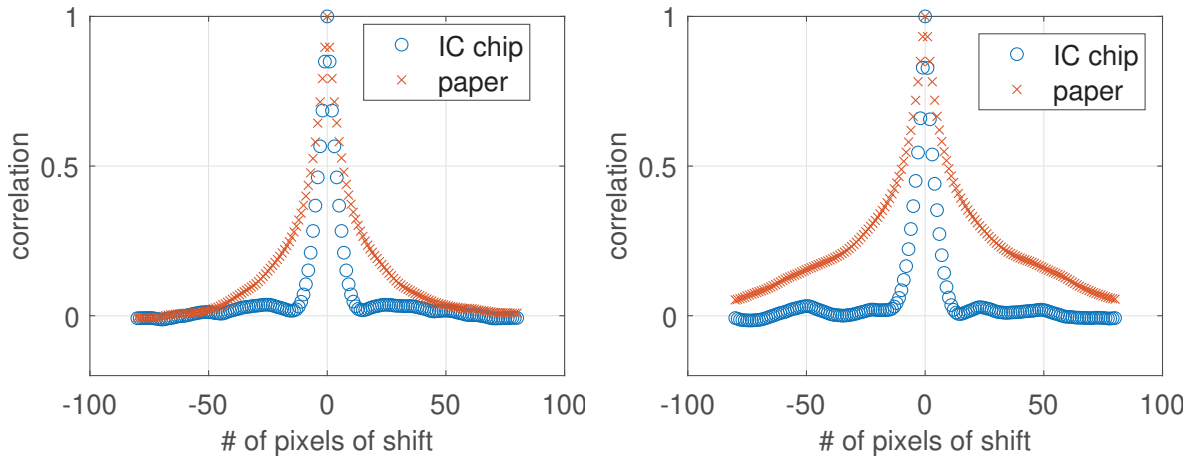


Figure 4.4: The correlation values between the heightmaps before and after shifting in the (a) $x-$ or (b) $y-$direction. The correlation drops faster for the IC chip surface when shifting.

Figure 4.5: An example of a (a) camera-captured image, (b) the template image used, and (c) the registered camera image. All images have undergone contrast enhancement for better visualization

we obtain template image for each IC chip. We scan the surface of IC chip using a flatbed scanner, which will generate a better-quality image with less noise than using a mobile phone. We then crop the area of interest of the captured image as the template image. Next, we register the captured test image. Given the test and template images, we first use phase correlation and Fourier properties [47] to estimate the translational, rotational, and scale movement to determine a geometric transform T. Finally, we use T to warp the test image with the template image as reference to obtain the final registered image. An example of a camera-captured image before and after registration is shown in Fig. 4.5.

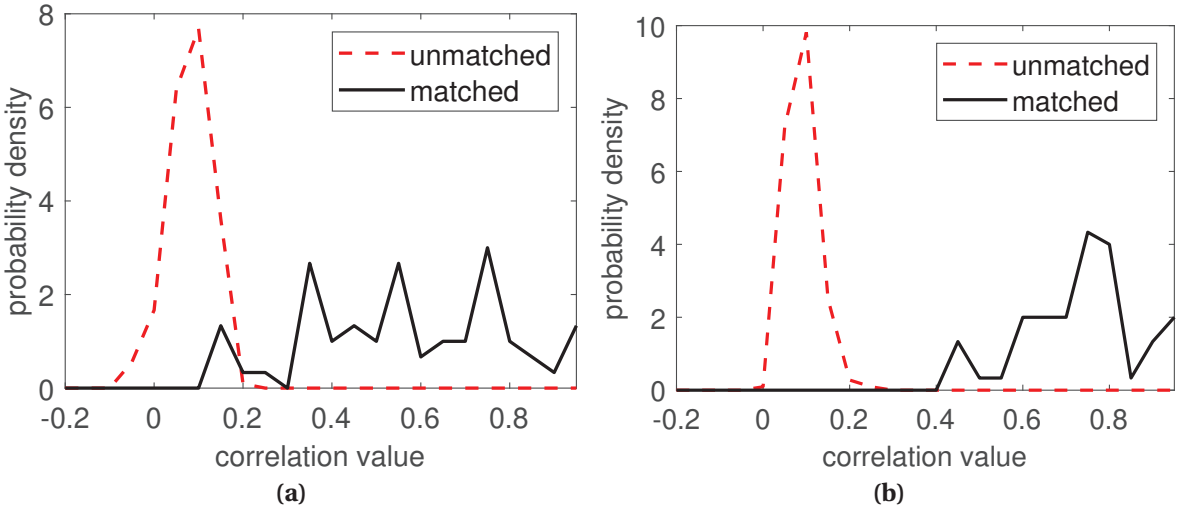


Figure 4.6: Histograms of correlation values between (a) x- or (b) y-component of norm maps estimated from mobile camera measurements.

4.3.3 Authentication Performance Using Optical PUFs

In this subsection, we examine the authentication performance in EER when using the extracted physical features for IC chip authentication. The test images are captured by the mobile camera, which is user-friendly and ubiquitous. We also compare the performance when reference images are acquired by mobile camera or flatbed scanner. Flatbed scanners are less portable to be used in real-world scenario but they are less prone to human error and can capture images with less noise.

We calculated the correlation values of norm maps between the test and reference data. Since scanner has been shown to be able to capture meaningful physical features, we used norm maps estimated from the scanner as the reference data and the norm maps estimated from the camera as the test data, are shown in Fig. 4.7. The correlation values are higher under the matched case, indicating that norm maps estimated from the camera can be used for IC authentication. When the test and reference data were both obtained from the mobile camera, the empirical distribution of correlation values are shown in Fig. 4.6.

We also followed the procedure in [36] to obtain different spatial-frequency subbands of the IC chip surfaces using the capture images. High spatial-frequency subbands can capture the micro-structures of the IC chip surfaces. For each subband, we calculate the correlations between the test and reference data and then calculated the EER. The authentication results using subbands are shown in Fig. 4.8 when test and reference data are acquired by the

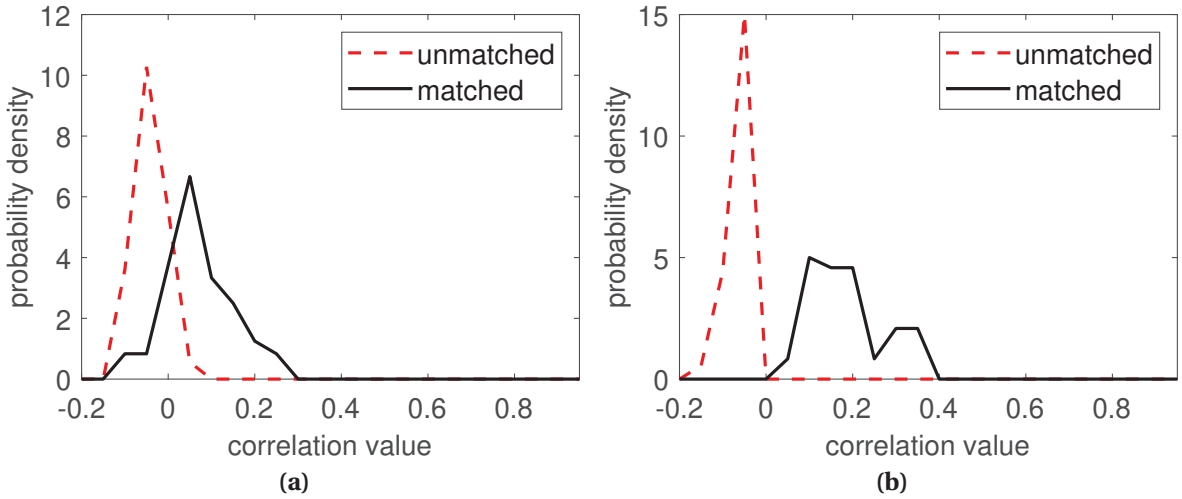


Figure 4.7: Histograms of correlation values between (a) x- or (b) y-component of norm maps estimated from mobile camera and flatbed scanner measurements.

mobile camera and the scanner, respectively. The sixth subband is the most discriminative physical feature. When test and reference data are both acquired with the camera, the subbands are not more discriminative than the norm map.

Besides of using the physical features norm map, which is estimated by taking the difference of raw images captured in the opposite directions, we also used the raw image captured in the same direction for authentication. The performance is worse than using the norm map. Compared to using norm map, using a single image is a fast verification method, which will further be explored in the next section.

4.4 Fast Verification for IC Chips

4.4.1 Dataset Collection

In the real world, it is inconvenient to capture images of IC chips in the opposite directions. It is more practical to capture a video of an IC chip from one direction. Also, all the frames in the captured video can be used for authentication, which can improve the confidence in the authentication results. In this proof-of-concept work, we use camera to capture videos of eight IC chip surfaces instead of images to mimic the real-world scenario. Similar to the setup when taking images using the camera, we fixed the camera directly above the IC chip surface to take videos. In the beginning of the video, we placed a lamp in the front

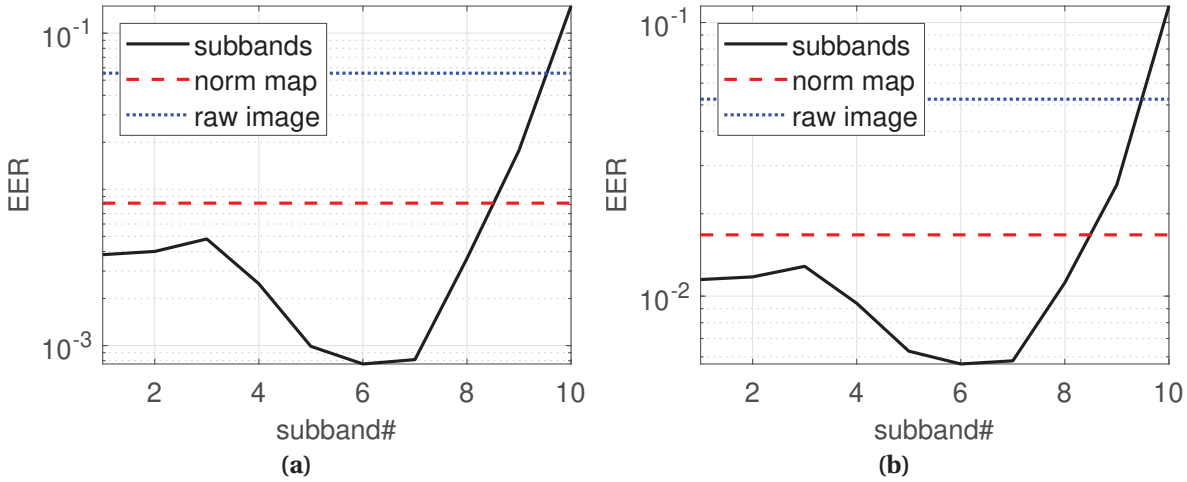


Figure 4.8: EER calculated for different physical features when correlation values are believed to follow (a) Gaussian or (b) Laplace distributions. Test images are captured using a mobile camera and the reference images are captured using a flatbed scanner. The horizontal lines are the performances when using y-component of norm map or the raw images in the y-direction. The sixth subband is the most discriminative physical feature for IC chip authentication.

of the IC chip, illuminating the chip at about 45° between the z-axis. Then we recorded a video with the light source moving slowly towards the IC chip, where the incident light direction changed from about 45° to about 30°. For each of the eight IC chip, we took three videos. Each video lasted about three to four seconds, resulting in about 100 frames. For the matched case, two videos of the same IC chip will be used as the test and reference video pair. For the unmatched case, two videos of different IC chips will be used as the test and reference video pair.

4.4.2 Authentication with Specular Points

The specular reflection on the IC chip surfaces causes pixel values very large in the captured images. The locations of specular points on the IC chip surface are determined by the microstructures and easy to spot, therefore the locations of specular points can also be used for fast authentication. When using the locations of specular points for authentication, we do not need to send the whole image, which will improve the communication efficiency.

Definition of Robust Specular Points

In a working pixel where specular reflection is observed, only a small part may cause the specular reflection. Since the chip surface is continuous, with small perturbation of camera or light source location, other parts in the working pixel or the neighboring pixels may also cause specular reflection. We call such working pixels "robust" specular pixels and exploit them for IC chip authentication.

We quantify the number of the "robust" specular points in a test and reference image pair. Denote the registered test image to be $\mathbf{X}^{\mathrm{t}}$, and the registered reference image to be $\mathbf{X}^{\mathrm{r}}$. We define the N pixels with the largest pixel values in the background of registered image as the observed specular points. Define $n(\mathbf{X}^{\mathrm{t}}, \mathbf{X}^{\mathrm{r}})$ to be number of robust specular point. A robust specular point is a pixel (i, j), where $\mathbf{X}^{\mathrm{t}}(i, j)$ is an observed specular point, and $\mathbf{X}^{\mathrm{r}}(i, j)$ with its 8-adjacent neighboring pixels have at least one observed specular point. We define the robust matching score S^{rm} to be

$$S^{\mathrm{rm}} = (n(\mathbf{X}^{\mathrm{t}}, \mathbf{X}^{\mathrm{r}}) + n(\mathbf{X}^{\mathrm{r}}, \mathbf{X}^{\mathrm{t}}))/2. \tag{4.4}$$

The robust matching score should be large when the test and reference images are from the same IC chip surface.

Design of Test Statistic

For each test video and reference video pair, we randomly sample ten frames from test and reference video to obtain the test and reference frames. The robust matching score between each test and reference frames are calculated, resulting in a total of 100 scores, denoted as $\{S_i^{\mathrm{rm}}\}_{i=1}^{100}$. For the test and reference frames of the same IC chip, the robust matching score S_i^{rm} will be larger if the incident light directions are more consistent. When the test and reference frames are from different IC chips, all S_i^{rm} should be small because the specular points appear at different locations in two different chips. Therefore, we use the maximum of the robust matching scores, i.e., $T^{\max} = \max(\{S_i^{\mathrm{rm}}\}_{i=1}^{100})$, as the test statistic for IC chip authentication. We apply the idea of bootstrapping [17] to obtain more scores by repeating the process of randomly sampling frames from the videos 50 times. We compare the estimated PDFs of raw score S_i^{rm} and max of scores $\max(\{S_i^{\mathrm{rm}}\}_{i=1}^{100})$ in Fig. 4.9. When using the raw scores as the test statistic, the EER is 5.5×10^{-2}, and EER improves to 2.4×10^{-3} when using max of scores as test statistic because of the smaller overlap of the two distributions.

The maximum of scores $\max(\{S_i^{\mathrm{rm}}\}_{i=1}^{100})$ under unmatched case may also be large, which

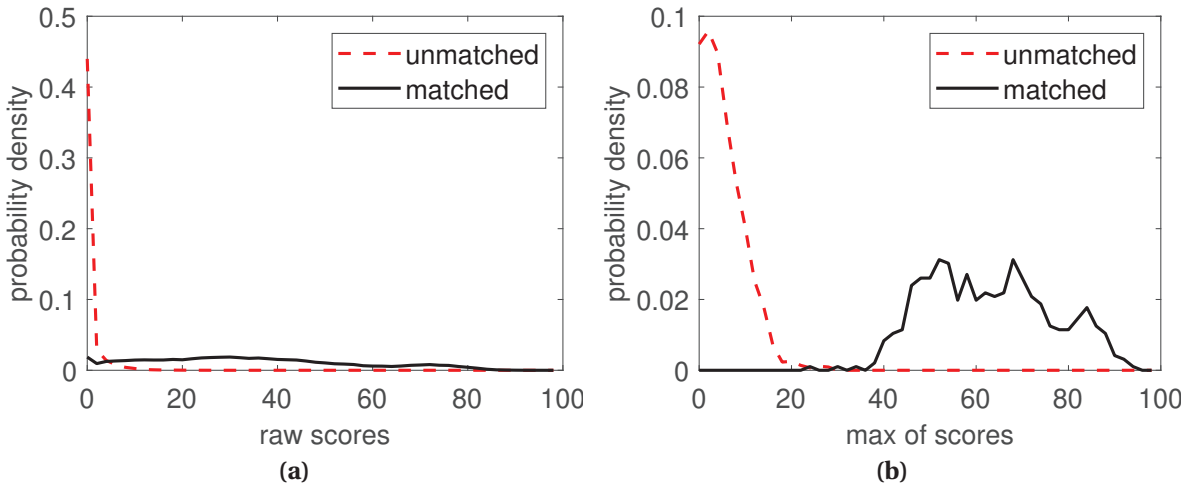


Figure 4.9: The estimated PDFs for (a) raw scores and (b) max of scores under the matched and unmatched cases. Under the matched case, the raw scores have a large spread, which will result in a bad authentication performance. The max of scores under the matched case are more concentrated at larger values.

may cause false positives. Therefore, we design a customized score to make the $\max(\{S_i^{\mathrm{rm}}\}_{i=1}^{100})$ to have a concentrated probability density under the unmatched case. We exploited the fact that, for different chips and perturbation of incident light angles, many of the scores in $\{S_i^{\mathrm{rm}}\}_{i=1}^{100}$ will be zero under the unmatched case, as Fig. 4.10(a) shows. We define a customized score T^{c} as test statistic:

$$T^{\mathrm{c}} = \max(\{S_i^{\mathrm{rm}}\}_{i=1}^{100})\mathbb{1}(r < t), \tag{4.5}$$

where $\mathbb{1}$ is the indicator function, $r = \sum_{i=0}^{100} \mathbb{1}(S_i^{\mathrm{rm}} = 0)/100$ is the ratio of scores in $\{S_i^{\mathrm{rm}}\}_{i=1}^{100}$ being zero, and t is the threshold. We set t to be 0.25, and estimated PDFs for the customized scores under the matched and unmatched cases, as shown in Fig. 4.10(b). There are no overlap of customized scores between the matched and unmatched case, and the distributions are far apart, indicating a good authentication performance.

4.4.3 Will More Frames or Observed Specular Points Help?

In previous subsections, we sample ten frames from each video. We study the effect of the number of frames sampled from each video by varying the number. We plot the empirical EER at different number of frames in Fig. 4.11(a). A few number of frames used will cause overlap between unmatched and matched customized scores, resulting in EER larger than

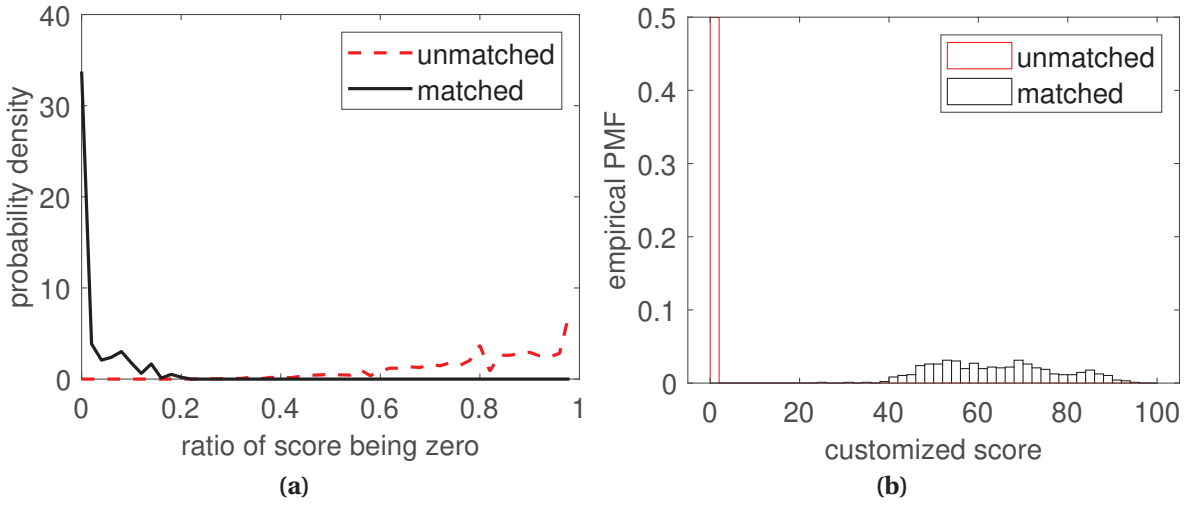


Figure 4.10: (a) The ratio of zeros in $\{S_i^{\text{rm}}\}_{i=1}^{100}$ under the matched and unmatched cases. The ratio of zeros in $\{S_i^{\text{rm}}\}_{i=1}^{100}$ is much larger under the unmatched case. (b) The estimated PMFs of customized scores under the matched and unmatched cases. The scores under the unmatched case concentrate at zero, indicating a good authentication performance.

zero. More frames will improve the authentication performance in terms of EER, but the communication cost will linearly increase with the number of frames used. The number of the observed specular points was set to be 100 in the experiments. We also studied the effect of the number of observed specular points used on the customized scores, as shown in Fig. 4.11(b). As the number of observed specular points increase, the mean and standard deviation of the customized scores will increase for both matched and unmatched cases. Large number of observed specular points may cause false positives, i.e., bright pixel caused by fully diffuse component may falsely be regarded as a specular point. The number of 100 is a reasonable choice because the scores for the matched case are large while the scores for the unmatched case are small with small standard deviation.

4.4.4 Verification with Images Taken in the Same Direction

To compare with authentication performance using specular points, we use the frames in videos captured in the same direction for IC chip authentication. For each test and reference frame pair, we sampled ten frames from each video to obtain ten test frames and reference frames. We calculated the correlations between the background of each test frame and the reference frames, resulting in a total of 100 correlation values, denoted as $\{c_i\}_{i=1}^{100}$. We used only the background of the chip surface because the attackers may print

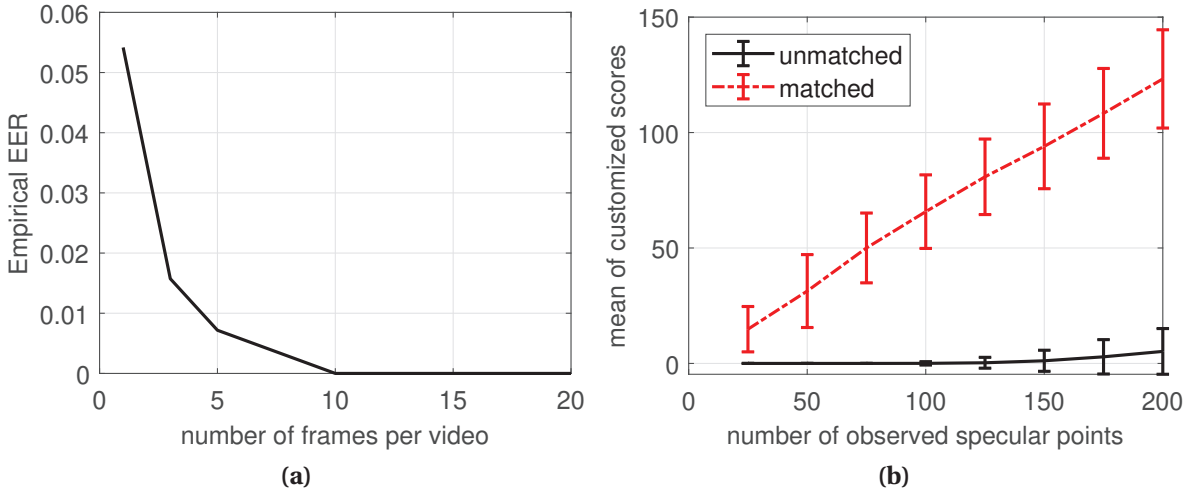


Figure 4.11: (a) The impact of number of frames sampled from each video on the authentication performance in terms of EER. Larger number of frames used will lead to better performance. (b) The impact of number of frames sampled from each video on the authentication performance. The error bars indicate one sample standard deviation above and below the averaged customized scores. As the number of observed specular points increase, scores for the matched case will increase fast.

the same text on the chip surface, which will cause high correlations for two images from different chips. We repeat the process of randomly sampling frames 20 times to obtain more correlation values. We plot the correlation values and max of the correlation values from a test-reference video pair $\max(\{c_i\}_{i=1}^{100})$ in Fig. 4.12. When using $\max(\{c_i\}_{i=1}^{100})$ as the test statistic, the distributions under the matched and unmatched cases are farther apart, indicating a better authentication performance. Instead of sampling ten frames from each video, we also sampled other numbers of frames and then used the max of correlation values as the test statistic. We assumed that the correlation values are Gaussian or Laplacian distributed to extrapolate the tail information, and then calculate the EER, as shown in Fig. 4.13. Larger number of frames will result in a better authentication performance in terms of EER. Transferring images from a user to a server may be expensive, in the next subsection we investigate using the locations of specular points to save communication cost.

4.5 Discussion

We discuss the IC chip authentication using the proposed method in the real-world application scenario. After encapsulating the IC chips, the manufacturer can capture videos of

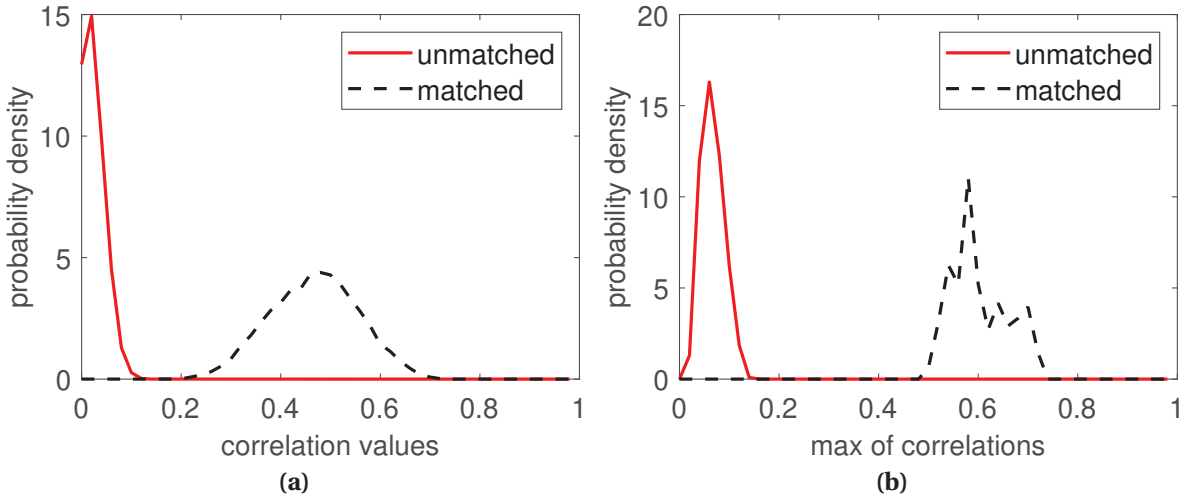


Figure 4.12: The estimated PDFs for (a) raw correlation values and (b) max of correlation values under the matched and unmatched cases. Under the matched case, the raw correlation values have a larger spread, which will result in a worse authentication performance. The max of correlation values under the matched case are more concentrated at larger values.

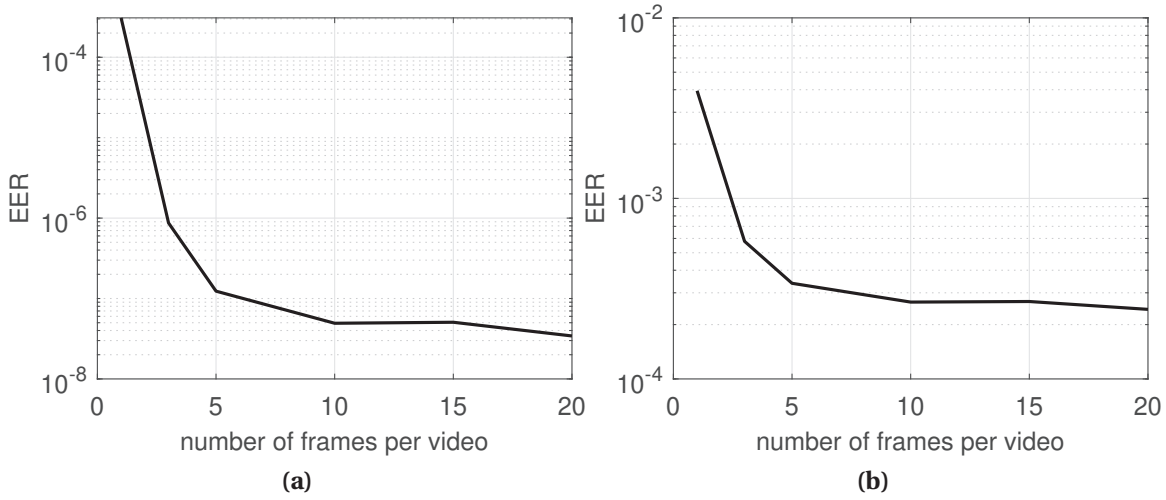


Figure 4.13: The impact of number of frames sampled per video to calculate correlations on the authentication performance: EER against the number of frames when assuming (a) Gaussian and (b) Laplace distributions. More sampled frames will result in a smaller EER.

the IC chip surfaces as reference videos. Then the reference videos are saved for authentication in the future. The IC chip company that manufactured the IC chips will provide authentication service by comparing the test videos from customers with the pre-captured reference videos. A customer who purchased IC chips can capture the test videos of the

chip surfaces, and send the videos to the manufacturer. Cryptographic protocols, such as the transport layer security (TLS), can be used to protect the communication channel between the customer and the manufacturer of the IC chips. Users can use an assembly line to capture test videos of a large number of IC chips efficiently. The authentication results will be sent back to the customer.

4.6 Appendix: Theoretical Analysis on the Effect of Distance Between IC Chip and Light Source

Denote the center of the IC chip to be at the origin, i.e., $(0,0,0)$. Denote the distance between the light source and the chip center to be L, and the incident light directions taken at the opposite directions to be $\mathbf{o}_0 = (\sqrt{2}/2, 0, \sqrt{2}/2)L$ and $\mathbf{o}_{180} = (-\sqrt{2}/2, 0, \sqrt{2}/2)L$. For the convenience of analysis, we studies the pixels on the diagonal of the IC chip, i.e., $\mathbf{p} = (1,1,0)\frac{A}{2}$. The incident light direction is $\mathbf{v} = \frac{\mathbf{o}-\mathbf{p}}{\|\mathbf{o}-\mathbf{p}\|}$, and the perceived intensity $I = \frac{l \cdot w_d}{\|\mathbf{o}-\mathbf{p}\|^2}(\mathbf{n}^T\mathbf{v})$. The estimation of x-component of norm map can be obtained by:

$$I_{0^\circ} - I_{180^\circ} = l \cdot w_d \left[\frac{n_x\left(\frac{\sqrt{2}}{2} - \frac{\eta}{2}\right) - n_y\frac{\eta}{2} + \eta_z\frac{\sqrt{2}}{2}}{L^2\left(1 + \frac{\eta^2}{2} - \frac{\sqrt{2}}{2}\eta\right)^{\frac{3}{2}}} - \frac{n_x\left(-\frac{\sqrt{2}}{2} - \frac{\eta}{2}\right) - n_y\frac{\eta}{2} + n_z\frac{\sqrt{2}}{2}}{L^2\left(1 + \frac{\eta^2}{2} + \frac{\sqrt{2}}{2}\eta\right)^{\frac{3}{2}}} \right] \triangleq l \cdot w_d \cdot d_1, \tag{4.6}$$

where $\eta = A/L$. As $\eta \longrightarrow 0$, i.e., the light source is infinitely far away, $I_{0^\circ} - I_{180^\circ} = l \cdot w_d \frac{\sqrt{2}n_x}{L^2} \triangleq l \cdot w_d \cdot d_0$. We denote $e = |d_0 - d_1|$ as the absolute error. We studied the effect of distance L on the absolute error e when fixing $A = 0.92$, as shown in Fig. 4.14(a). A distance of 40 cm can produce negligible error, when the absolute error is around 0.298. The smaller standard deviation indicates that the result is stable for different norm map data. We also fixed the distance L to be 40 cm, and varied the distance from pixel of interest to chip center A, as shown in Fig. 4.14(b). A larger distance to chip center will increase the absolute error e.

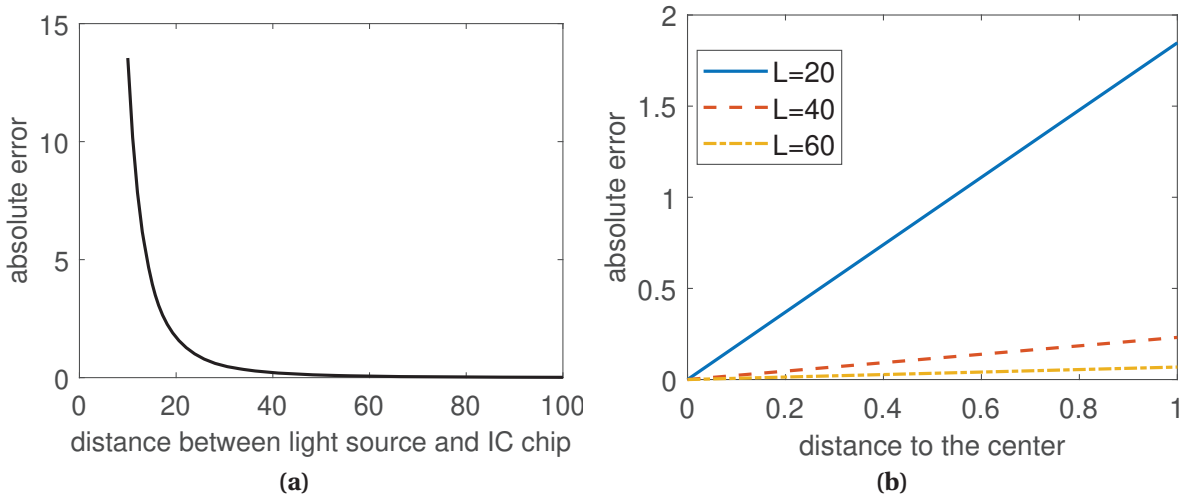


Figure 4.14: (a) The effect of distance between light source and IC chip on the absolute error of estimation of norm map. 40 cm will ensure both strong reflected light and small error. (b) The effect of the distance between pixel of interest and the chip center. A smaller distance results in more accurate norm map estimation.

CHAPTER

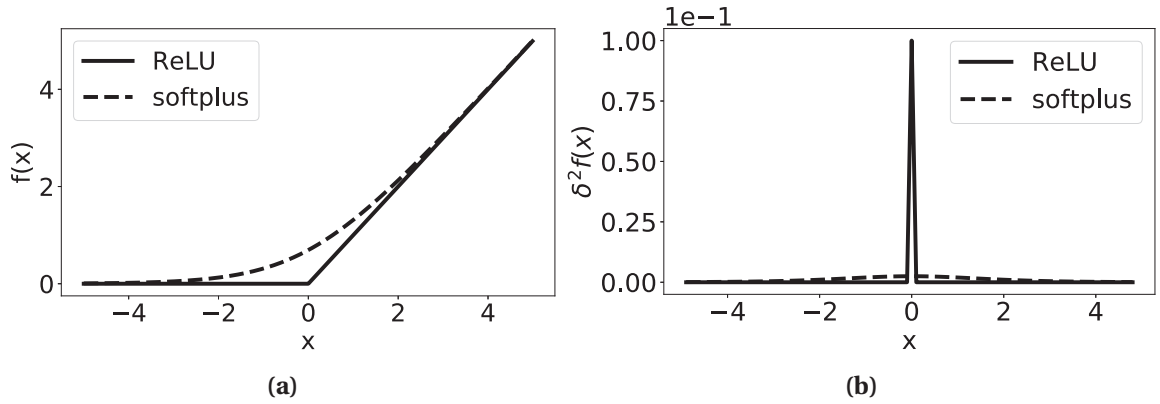
5

MODELING THE NONSMOOTHNESS OF MODERN NEURAL NETWORKS

5.1 Introduction

In the past few years, the fast development of convolutional neural networks (CNN) has led to many successful regression-based applications such as face recognition [48], facial landmark detection [49], and 3D pose estimation [50]. CNN has also been used in many other fields such as 2D/3D registration of medical images [51] and stock price prediction [52].

Another successful use of neural networks is face synthesizing. Deepfake videos synthesized by neural networks nowadays can achieve high quality and look authentic to human eyes [53]. It has raised public concerns because such synthesized images may have detrimental usages. For example, deepfake has been used to swap two person's faces in videos, which has posed serious challenges to privacy, civic engagement, and governance. Detecting the images and videos synthesized by neural networks has become an emerging research topic and researchers have been exploring different methods to detect these



(a) **(b)**

Figure 5.1: (a) ReLU and softplus activation functions near $x = 0$. ReLU is nonsmooth at $x = 0$. (b) Second-order difference of ReLU and softplus when the input x is sampled at a step size of 0.1.

images and videos. Some approaches exploited biological features [54] for deepfake detection. Other methods used artifacts/processing traces left by neural networks for detection [55, 56, 57].

In this work, we investigate an intuitive but understudied characteristic, the *nonsmoothness*, of modern neural networks comprising convolutional layers, rectified linear unit (ReLU) activation functions, and max pooling layers. Our investigation was inspired by the exploration of deepfake generation and detection. The nonsmoothness can be considered as an artifact of the neural network when it is treated as a processing unit, and we believe that it can be potentially used as a generic forensic tool in many regression-based applications. We will mathematically define the nonsmoothness and use synthetic data to confirm its existence. We will study the statistical properties of the nonsmoothness and model the events of nonsmoothness.

The rest of the chapter is organized as follows. In Section 5.2, we review key building blocks of modern neural networks. We define the concept of nonsmoothness for neural networks in Section 5.3, and confirm its existence via experiments in Section 5.4. In Section 5.5, we model the nonsmoothness events. In Section 5.8, we conclude the chapter.

5.2 Background and Preliminaries

Activation Functions Different activation functions have been used in neural networks to introduce nonlinearity. Prior to 2011, researchers had mostly used logistic sigmoid or hyperbolic tangent as activation functions. Glorot et al. [58] argued that the ReLU can better model biological neurons and showed that it can lead to sparse networks. ReLU is currently

widely used for its simple implementation and fast convergence [59]. ReLU clips negative x to zero while keeping positive x untouched, namely, $g_{\text{ReLU}}(x) = \max(0, x)$. The softplus activation [60], $g_{\text{softplus}}(x) = \ln(1 + e^x)$, can be considered as a smooth approximation of ReLU. Fig. 5.1(a) compares ReLU and softplus functions. ReLU activation has a change of slope at $x = 0$, while softplus is smooth everywhere.

Pooling Methods In convolutional neural networks, pooling layers are used to summarize data via downsampling. The max pooling outputs the largest element within a predefined range, whereas the average pooling outputs the average of all elements. Compared with average pooling, max pooling may lead to problems in deep networks because the gradients flow through only the node with maximum value [61]. Also, Boureau et al. [62] showed that max pooling is well suited to sparse features. Zhang [63] discussed the aliasing effect of max pooling and applied anti-aliasing by integrating lowpass filtering to make convolutional networks shift-invariant. In this work, we will show that ReLU and max pooling introduces nonsmoothness to neural networks, and build mathematical models for the nonsmoothness events.

5.3 Nonsmoothness in Modern Neural Networks

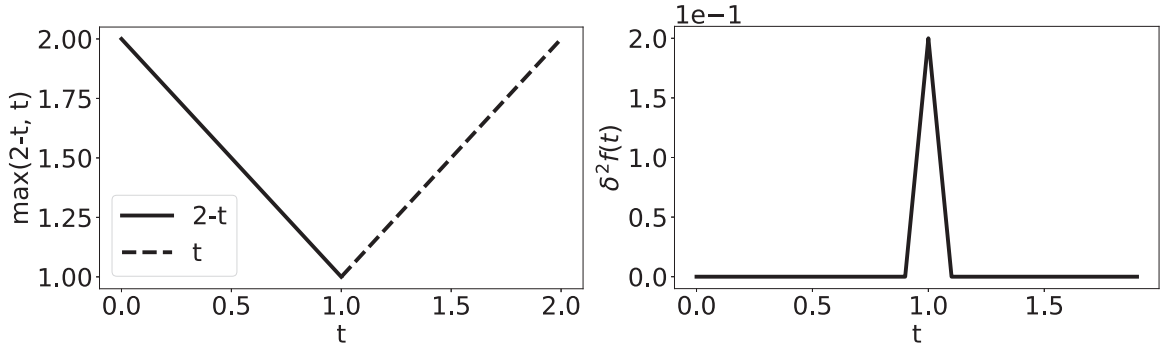
5.3.1 Definition of Nonsmoothness

Given a continuous function $f : \mathbb{R} \to \mathbb{R}$ and a parameter $t \in \mathbb{R}$, we define that function f is smooth if the first-order derivative, $f'(t) = \lim_{h \to 0} \frac{f(t+h)-f(t)}{h}$, exists for all t. For example, f is nonsmooth at t_0 when the left-hand and right-hand limits of the first-order derivative do not equal: $\lim_{t \nearrow t_0} f'(t) \neq \lim_{t \searrow t_0} f'(t)$. For simplicity, we denote the left- and right-hand limits at t_0 as $f'(t_0^-)$ and $f'(t_0^+)$, respectively. For the continuous case, a nonsmooth point t_0 can be detected if $f'(t_0^+) - f'(t_0^-) \neq 0$. If $f(t)$ also contains noise, then by allowing some false positive and false negative, the following detector may be used to detect a nonsmooth point t_0:

$$|f'(t_0^+) - f'(t_0^-)| > \tau_c, \tag{5.1}$$

where τ_c is a detection threshold.

In the digital world, any input to f is discrete in time, and one intuitive choice for detecting nonsmoothness is to adapt the nonsmoothness detector by replacing the derivative in (5.1) with the difference operation. When input t is discrete with a uniform sampling period Δ, i.e., $t \in \Delta\mathbb{Z}$, the first-order difference is $\delta f(t) = f(t + \Delta) - f(t)$, and (5.1) may be



(a) **(b)**

Figure 5.2: (a) The output of the max pooling function of a toy example. The curve is nonsmooth at $t = 0$. (b) Second-order difference of the output when t is sampled at a step size of 0.1.

adapted to:

$$|\delta^2_\Delta f(t)| = |f(t+\Delta) + f(t-\Delta) - 2f(t)| > \tau_d, \tag{5.2}$$

where $\delta^2_\Delta f(t)$ is the second-order difference and τ_d is a detection threshold for the discrete case. Similar to (5.1), there can be false positive and false negative caused by the detector (5.2), but we have to skip the details due to space limitations. A vector-valued function f, i.e., $f(t) = (f_1(t), \ldots, \mathcal{N}(t))$, where $f_i(t) \in \mathbb{R}$ for $i = 1, \ldots, n$, is said to be smooth if f_i is smooth for all i's.

5.3.2 Causes of Nonsmoothness in Neural Networks

Nonsmoothness Caused by Activation Functions We show the nonsmoothness caused by activation functions in neural networks. The neural network will behave nonsmoothly due to the nonsmooth activation functions. Assuming threshold $\tau_d = 0.02$, ReLU fulfills (5.2) at 0 but softplus does not, as Fig. 5.1(b) shows. Hence, the detector (5.2) believes that ReLU causes nonsmoothness but softplus does not.

Nonsmoothness Caused by Max Pooling Max pooling in a convolutional neural network also contributes to nonsmoothness due to the sudden change in the input–output routing as the input changes smoothly. For example, when a max pooling function operates on a smooth 2×2 subregion, $\big(x_1(t), x_2(t), x_3(t), x_4(t)\big) = (0, 0, 2-t, t)$, where the parameter t

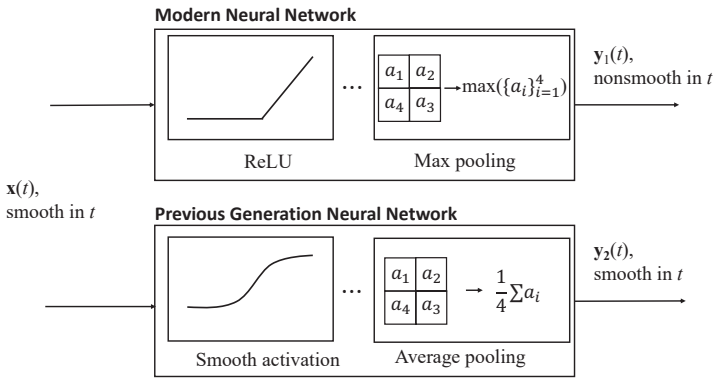


Figure 5.3: Modern neural networks prefer nonsmooth ReLU activation and max pooling, whereas previous generation neural networks prefer smooth activations and average pooling. With a smooth input function $\mathbf{x}(t)$ in t, output of modern neural network $\mathbf{y}_1(t)$ will be nonsmooth in t due to compositing with nonsmooth functions, whereas output of previous generation neural network $\mathbf{y}_2(t)$ will be smooth.

represents time, the output will be:

$$\max\big(x_1(t), x_2(t), x_3(t), x_4(t)\big)=\begin{cases} 2-t, & 0 \le t < 1; \\ t, & t \ge 1. \end{cases} \tag{5.3}$$

Note that the output of the max pooling is $2-t$ for $t \in [0,1)$ and t for $t \in [1,\infty)$, as shown in Fig. 5.2 demonstrating a sudden routing change at $t=1$ for $x_3(t)$ and $x_4(t)$. The second-order difference is $0.2 > \tau_d$ when the step size is 0.1, indicating detected nonsmoothness. For the same input, the average pooling will instead produce a constant output $\frac{1}{2}$. Compared with the average pooling, the max pooling method causes nonsmoothness in neural networks, and this will be experimentally verified in the next section.

Fig. 5.3 summarizes the behaviors of modern and previous generation neural networks. When the input to the neural network is smooth, modern neural networks with ReLU activation and max pooling will have nonsmooth output, whereas previous generation neural networks [64, 65] with smooth activation and average pooling will have smooth output.

5.4 Simulated Justification

In this section, we use synthetic data to confirm that modern neural networks will cause nonsmoothness. To illustrate the effect of the nonsmoothness along the time, the input video will be constructed as a smooth function of the time, letting the representation of an image moving smoothly on a manifold. In the experiments, we will use autoencoders

to reconstruct the input videos frame by frame, such that the collocated pixel values in an input video and its reconstructed video can be directly compared for examining the effect of nonsmoothness. We will also show another example that the representation of an image moving smoothly in the Euclidean space in Section 5.6 of the supplementary material.

5.4.1 Dataset Generation and Autoencoder Training

In the simulation, we let the image vary smoothly such that its representation in the high dimensional space forms a smooth trajectory on a manifold. We used a sphere or part of an ellipsoid to coarsely model a human face under changing illumination. We rendered a video of the ellipsoid with a moving point light source. The highlight of the ellipsoid changes smoothly as the light moves.

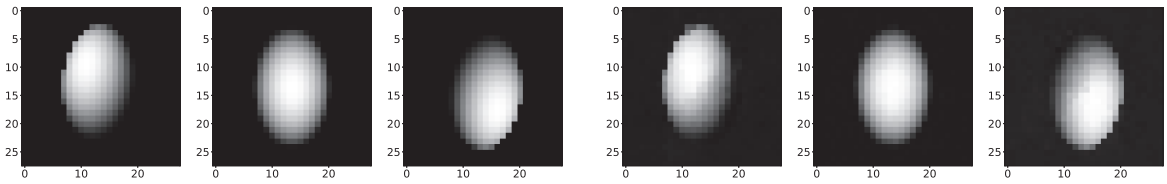
An ellipsoid is a quadric surface that follows the equation $\frac{x^2}{a^2}+\frac{y^2}{b^2}+\frac{z^2}{c^2}=1$. In this work, we used a half ellipsoid above the xy-plane, i.e., $z>0$ and fix $a=2.5$, $b=4$, $c=1$. To create a smooth video, we illuminate an ellipsoid by moving a point light source at a constant speed. We calculate the perceived intensity of a point $\mathbf{p}$ on the ellipsoid by the fully diffuse light reflection model [14], namely, $\max(\mathbf{v}_i^T\mathbf{n},\ 0)$, where $\mathbf{v}_i$ is the incident light direction at $\mathbf{p}$, and $\mathbf{n}$ is the normal vector at $\mathbf{p}$ on the ellipsoid.

To construct a training dataset, we randomly generated 10000 point light source locations $\{(x_i, y_i, 20)\}_{i=1}^{10000}$, where $x_i, y_i \overset{\text{iid}}{\sim} \mathcal{U}(-10, 10)$. Another 1000 images were generated in the same way to construct the validation dataset. Several ellipsoid images with different light source locations are shown in Fig. 5.4(a).

Two autoencoders were trained to reconstruct the ellipsoid images of different light source locations. One autoencoder follows Setup 1: ReLU activation and max pooling, and the other autoencoder follows Setup 2: softplus activation and average pooling. Adam optimizer and MSE loss were used for training. The learning rate was 10^{-3} to ensure a fast training speed and good training performance for the scale of this problem. To avoid overfitting, the model with the lowest validation error is considered as the trained model. The reconstructed frames using the trained autoencoder are shown in Fig. 5.4(b).

5.4.2 Processing Smooth Input Videos by Neural Networks

We show the nonsmoothness introduced by the autoencoder. First, we need an input video that is smooth in time. We constructed a test video where the light source moves from $(-9,-9,20)$ to $(9,9,20)$ at a speed of 2 pixels per second. We reconstructed the test



(a) **(b)**

Figure 5.4: (a) Three frames of an input video. The point light source moved from northwest to southeast. (b) The reconstructed frames of those in (a) using a trained autoencoder. The reconstructed images are slightly blurred.

video with the two trained autoencoders, RELU+MAXPOOL or SOFTPLUS+AVEPOOL. For a pixel location, the intensity curves of the input video and reconstructed videos with two autoencoders are shown in Fig. 5.5(a). The trends of the curves appear smooth because the goal of an autoencoder is to reconstruct the input video. The second-order difference curves are shown in Fig. 5.5(b). Compared to the autoencoder using SOFTPLUS+AVEPOOL, the autoencoder using RELU+MAXPOOL has introduced more nonsmoothness in terms of second-order difference.

To quantify the nonsmoothness caused by autoencoders, we define a statistic for a given video named *the average nonsmoothness* as follows:

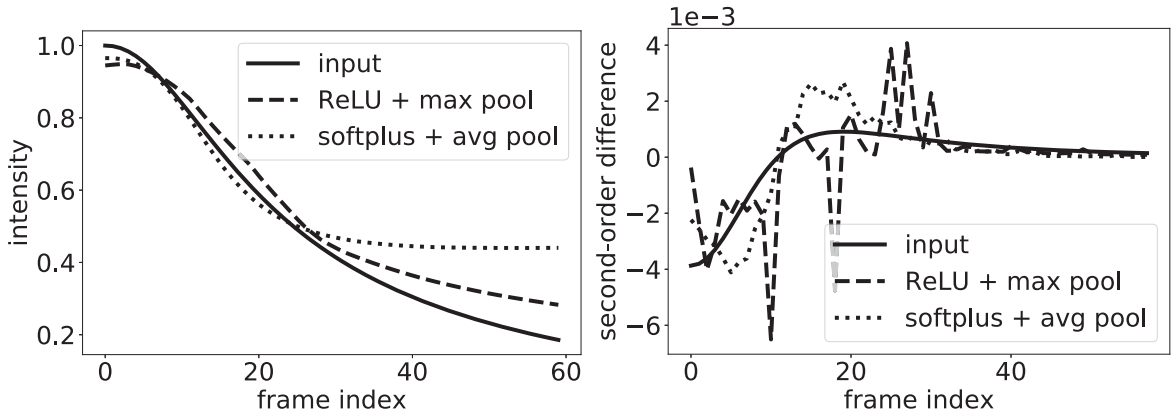
$$\text{AveNonSmooth} = \frac{1}{MNT}\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{t=1}^{T}|\delta^2 y_{i,j}(t)|, \tag{5.4}$$

where $\delta^2 y_{i,j}(t)$ is the second-order difference of pixel location (i, j) at time t, M and N are the number of rows and columns of the frame, respectively, and T is the total number of frames of the video. We examined the distribution of AveNonSmooth of 100 input videos with random moving paths of the light source reconstructed with ten trained autoencoders. Fig. 5.6 shows the histograms of the AveNonSmooth of the reconstructed videos. SOFTPLUS+AVEPOOL hardly increased AveNonSmooth but RELU+MAXPOOL increased significantly. Hence, we confirm that the autoencoder with RELU+MAXPOOL leads to nonsmoothness.

5.5 Modeling of Nonsmoothness Events

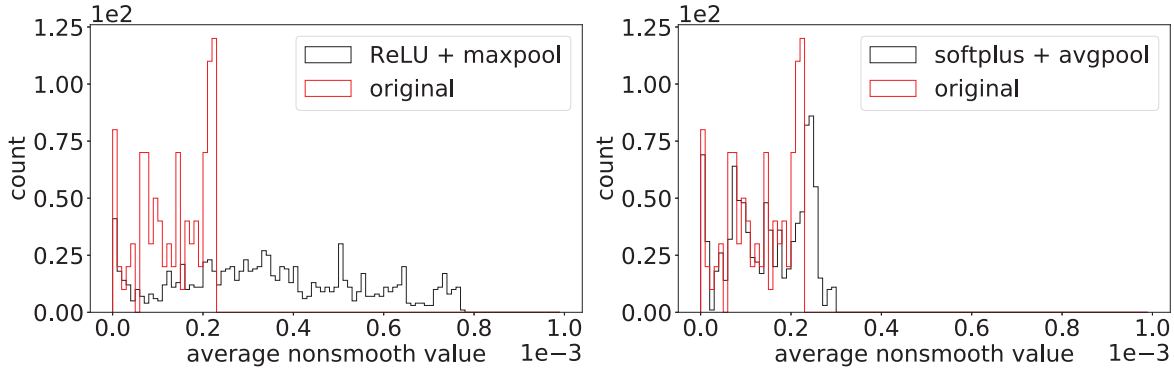
5.5.1 Motivation for Modeling

In Sections 5.3 and 5.4, we show that ReLU and max pooling are the causes of nonsmoothness in neural networks. Such nonsmoothness traces due to neural networks may be used



(a) **(b)**

Figure 5.5: For a representative pixel location, (a) intensity curves and (b) second-order difference curves of input video and reconstructed videos using the autoencoders of ReLU+MaxPool and Softplus+AvePool. The autoencoder of ReLU+MaxPool caused more nonsmoothness in terms of the second-order difference.



(a) **(b)**

Figure 5.6: The histograms of the AveNonSmooth of the original and the reconstructed videos from (a) ReLU+MaxPool, and (b) Softplus+AvePool. When using autoencoder with ReLU+MaxPool to reconstruct videos, the AveNonSmooth is larger in general.

as a forensic tool for regression-based applications. For example, when neural networks are used to generate deepfake videos, they can introduce nonsmoothness in the resulting videos, as if an extra processing unit was applied to an authentic video, as shown in the left half of Fig. 5.7. A classifier can be exploited to detect this extra processing unit by examining the statistical traces it leaves behind, effectively detecting deepfake videos. In this section, we investigate and model the statistical behaviors of nonsmoothness events by examining the propagation of nonsmoothness events through various building blocks of neural networks. In this paper, we focus on the analysis of the convolutional layer, ReLU

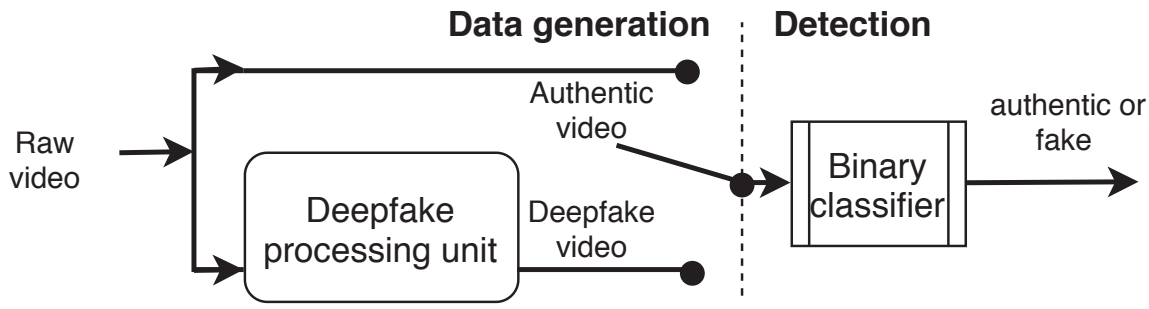


Figure 5.7: Conceptual diagram for authentic/fake videos generation and deepfake detection. The binary classifier aims at detecting the processing unit used to generate deepfake videos.

activation, max pooling layer, and the combined effect of a convolutional layer with ReLU activation and a max pooling layer. The modeling results for the transpose convolutional layer are reported in Section 5.7 of the supplementary material. We leave the combined effect of all layers of the neural network for future work.

5.5.2 Modeling of Convolutional Layers

When an input video is smooth, a large value in the second-order difference curve represents nonsmoothness caused by the neural network. In this section, we define a *peak* to be a value that is ten times larger than the sample mean and median in a time series. The occurrence of a peak in the second-order difference curve is said to be a *nonsmoothness event*. Besides the peaks, the second-order difference for a smooth function may not be zero due to the sampling period that is not infinitesimal. Using the sum of the second-order difference in a time series as in (5.4) may overestimate the nonsmoothness and therefore result in inaccuracy in modeling. To better quantify the nonsmoothness for modeling, we propose to use the *sum of the magnitude of peaks (SMP)* of the second-order difference curve.

Properties of Nonsmoothness Events To model the nonsmoothness events in neural networks, we first need to investigate their behaviors. It is important to understand the properties of the nonsmoothness events in the summation of the time series because a convolutional layer in the neural network will linearly combine coordinates of the input data, i.e., the output of convolutional layer is the weighted sum of input nodes. Given two continuous functions $f_1, f_2 : \mathbb{R} \to \mathbb{R}$, f_1 is nonsmooth at t_1, i.e., $f_1'(t_1^+) \neq f_1'(t_1^-)$, and f_2 is nonsmooth at t_2, and define $f_{\text{sum}}(t) = f_1(t) + f_2(t)$. Below, we present two properties of nonsmoothness events. First, the occurrence times of the nonsmoothness events are inherited, i.e., $f_{\text{sum}}(t)$ is nonsmooth at t_1 and t_2. Second, the change of first-order derivatives

is inherited, i.e., $f'_{\text{sum}}(t_1^+) - f'_{\text{sum}}(t_1^-) = f'_1(t_1^+) - f'_1(t_1^-), f'_{\text{sum}}(t_2^+) - f'_{\text{sum}}(t_2^-) = f'_2(t_2^+) - f'_2(t_2^-)$. For the discrete case, nonsmoothness events may happen simultaneously in different time series. Summing up the magnitude of these peaks will overestimate the nonsmoothness, and a discounting multiplicative factor can be used to compensate for this issue. In this section, we assume that the occurrence times of nonsmoothness events are different and the second-order differences will be inherited since we have observed from data that the events are rare.

Linear Model for Convolutional Layers We now model the nonsmoothness events in the convolutional neural networks. First, we investigate SMP for input nodes and output nodes of a convolutional layer. Denote the input of the convolutional layer to be $\mathbf{X} \in \mathbb{R}^{M \times M \times C_{\text{in}}}$, where C_{in} is the number of input channels and $M \times M$ is the size of the image in each input channel. Denote the output of the convolutional layer to be $\mathbf{Y} \in \mathbb{R}^{N \times N \times C_{\text{out}}}$, where C_{out} is the number of input channels and $N \times N$ is the size of the image in each output channel. For the convolutional layer, denote the size of the convolutional kernel to be $k \times k$, the stride to be s, and the padding to be p.

For pixel location (u, v) in channel c of input data, denote its SMP by X_{cuv}. Similarly, for pixel location (m, n) in channel c of output data, denote its SMP by Y_{cmn}. Due to the inheritance of the nonsmoothness events, a linear model can be established for Y_{cmn} and X_{cuv} as follows:

$$Y_{cmn} = \sum_{c=1}^{C_{\text{in}}} \sum_{u=ms-p}^{ms-p+k-1} \sum_{v=ns-p}^{ns-p+k-1} |W_{cij}| \cdot X_{cuv}, \tag{5.5}$$

where W_{cij} is the weight in the convolutional kernel for location (u, v) in channel c of input, i.e., $i = u-(ms-p)$, $j = v-(ns-p)$. Since the channels in the input or output are the same without loss of generality, SMPs in different channels $\{Y_{cmn}\}_{c=1}^{C_{\text{out}}}$ are identically distributed. Therefore, we define expectations of SMPs to be $\mu^Y_{mn} = \mathbb{E}[Y_{cmn}]$ for all c, and similarly $\mu^X_{mn} = \mathbb{E}[X_{cuv}]$. Taking the expectations of both sides of (5.5), the SMP of input and output can be related as follows:

$$\mu^Y_{mn} = w_0 \cdot C_{\text{in}} \cdot \sum_{u=ms-p}^{ms-p+k-1} \sum_{v=ns-p}^{ns-p+k-1} \mu^X_{mn}, \tag{5.6}$$

where we have invoked the assumptions that $\mathbb{E}\big[W_{cij}\big] = w_0$ for all (i, j) and c, and W_{cij} is independent of X_{cuv}. The latter assumption is justified by our analysis that the correlation between the weights and the magnitudes of peaks of the real data in Section 5.4 is small, i.e., 0.16.

We used the simulated ellipsoids data from Section 5.4 to test the effectiveness of the model for nonsmoothness events. We recorded intensity curves along the time for each node in the autoencoders of ReLU+MaxPool and then calculated SMP for the input nodes and output nodes for the second convolutional layer. We did not use the first convolutional layer because the input video is smooth and the SMP of input will be zero. The sample mean of the SMP at different channels was used to approximate the expectation of SMP, i.e., $\hat{\mu}^Y_{mn} = 1/C_{\text{out}} \sum_{c=1}^{C_{\text{out}}} y_{cmn}, \hat{\mu}^X_{mn} = 1/C_{\text{in}} \sum_{c=1}^{C_{\text{in}}} x_{cuv}$, where y_{cmn} is the SMP at location (m,n) in channel c at the output data of the convolutional layer, and x_{cuv} is the SMP at location (u,v) in channel c in the input data. C_{in} and C_{out} are the number of channels in the input and output data, respectively.

For a pixel location (m,n) in the output, we used the model (5.6) to predict the SMP using the SMPs in input nodes, i.e., $\tilde{\mu}^Y_{mn} = w_0 C_{\text{in}} \sum_u \sum_v \hat{\mu}^X_{mn} = w_0 \sum_u \sum_v \sum_c x_{cuv}$. We compared predicted SMPs $\tilde{\mu}^Y$ and SMPs in real data $\hat{\mu}^Y$ for the nodes in the output, as shown in Fig. 5.8(a). We fitted a linear model using the SMPs in real data as the response and predicted SMPs as the predictor, the R-squared value was 0.842. The high R-squared value indicates the model fits the nonsmoothness events well for the convolutional layer. To further confirm the effectiveness of the model, we plugged in the actual weight parameters in the model instead of using w_0, i.e., $\tilde{\tilde{\mu}}^Y_{mn} = \sum_u \sum_v \sum_c |W_{cij}| x_{cuv}$. The comparison of predicted and real-data results are shown in Fig. 5.8(b). We fitted a linear model using the simulation results as the response and modeled results as the predictor, and the R-squared value reached 0.944. We note that using the expected weight w_0 in the model (5.6) can lower the modeling accuracy. However, in a real-world scenario, we need to use the expected weight because the individual weights of neural networks are usually unknown. The confirmed linear relationship indicates that the nonsmoothness will propagate through the convolutional layers in a linear and almost deterministic way.

5.5.3 Modeling of ReLU

ReLU activation will create or eliminate nonsmoothness events. The statistical effect of ReLU operation on the SMP is summarized in Fig. 5.8(c) in the form of a highly structured joint distribution. The SMP can be reduced by a ReLU operation due to the deactivation of the node. The ReLU operation may also introduce additional nonsmoothness and therefore increasing the SMP. To predict SMP y at the output of the ReLU activation given input SMP x, we observe from Fig. 5.8(c) the concentrated density along the x-axis, the y-axis, and the $y = x$ line and therefore propose the following Bernoulli–Gaussian channel/conditional

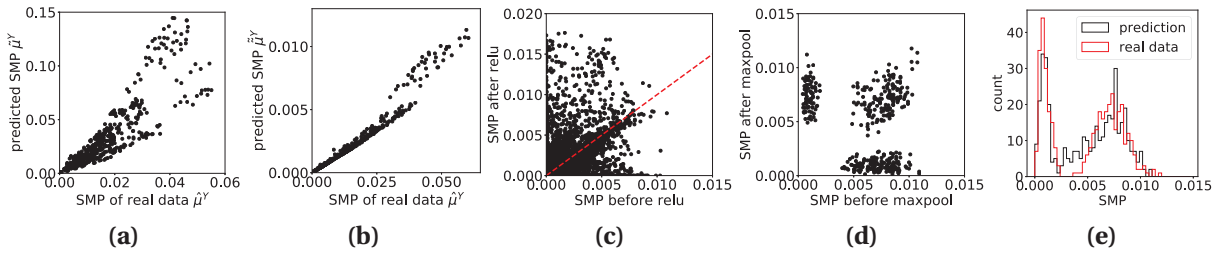


Figure 5.8: The scatter plots for real and predicted SMPs with (a) a constant w_0 as the expectation of weight parameters and (b) the actual weights W_{ij} in the trained autoencoders. The linear relationship between modeled and simulated results indicates the nonsmoothness events propagate through convolutional layers well. The scatter plots of SMP before and after (c) ReLU and (d) max pooling operations, the dashed line has a slope of one. (e) The distributions of predicted SMP and SMP in real data, the predicted distribution is consistent with the real data.

model:

$$f(y|x) = (1-\theta)\cdot\delta(y) + \theta\cdot\mathcal{N}(y;x,\sigma), \tag{5.7}$$

where θ is the parameter for Bernoulli distribution, δ is the Dirac delta function, and $\mathcal{N}(x;\mu,\sigma) = e^{-(x-\mu)^2/(2\sigma^2)}/(\sigma\sqrt{2\pi})$. The Bernoulli parameter θ and standard deviation σ of Gaussian can be estimated from paired input–output data.

5.5.4 Modeling of Max Pooling Layer

We also compared the SMPs in the input and output of the max pooling layers. For each pixel location in the output of max pooling layer, we calculated its SMP and compared it with the max value of SMPs in the corresponding locations in the input, as shown in Fig. 5.8(d). The max pooling can introduce nonsmoothness and therefore increasing SMP, whereas it can also reduce SMP in the output by choosing a channel with no nonsmoothness events in the input. To predict the SMP at the output of the max pooling layer, we built a (simplified) conditional model for the max pooling layer based on the joint distribution in Fig. 5.8(d). The output SMP y follows a Gaussian distribution when input SMP x is less than $a = 0.0025$, otherwise the output SMP y follows a mixture of two Gaussian distributions:

$$f(y|x) = \mathbb{1}(x<a)\cdot\mathcal{N}(y;\mu_0,\sigma_0) + \mathbb{1}(x\geq a)\cdot[\pi_0\mathcal{N}(y;x,\sigma_1) + (1-\pi_0)\mathcal{N}(y;\mu_2,\sigma_2)], \tag{5.8}$$

where $\mathbb{1}$ is the indicator function. The parameters of the Gaussian distributions $\mu_0, \sigma_0, \sigma_1, \mu_2$, σ_2 and the coefficient π_0 can be estimated from paired input–output data.

5.5.5 Prediction of Nonsmoothness Events

Finally, we predict the propagation of nonsmoothness events in a sequence of operations, i.e., the convolutional layer, ReLU activation, and the max pooling layer. Given the SMP to the input to the convolutional layer, we predict the SMP at the output of the max pooling layer with the linear model for the convolutional layer, the Bernoulli–Gaussian model for the ReLU, and the Gaussian mixture model for the max pooling layer. The distribution of predicted SMPs and SMPs in the simulated real data is compared in Fig. 5.8(e). It shows that the distribution of prediction matches well with the real data. Note that even though in (5.6)–(5.8) we have used parametric channel models, Monte-Carlo based methods can be used to address more generic channel models for nonsmoothness events. In future work, we plan to conduct the joint analysis of all convolutional layers, transpose convolutional layers, max pooling layers, and the ReLU activation for characterizing the nonsmoothness of the neural network as a processing unit. Such a statistical characterization of the processing unit can potentially enable forensic analysis of regression-based applications of neural networks.

5.6 Sequence of Images Forming a Smooth Trajectory in Euclidean Space

Dataset Generation and Autoencoder Training In this simulation, we let the image vary smoothly in a Euclidean space instead of on a manifold. We used digit “7" images, a total of 6265 images, from the MNIST dataset [66] to synthetically generate a smooth video dataset. To construct a training dataset, we rotated each image by 60 randomly generated angles, $\theta_i \overset{\text{iid}}{\sim} \mathcal{U}(-30°, 30°)$, where $\mathcal{U}$ denotes the uniform distribution. The total number of images in the training dataset was $6265 \times 60 = 375900$. Three typical images in the dataset are shown in Fig. 5.9(a). To construct a validation dataset, we synthetically generated another 60 angles for each of the 6265 template images.

We trained two autoencoders that can reconstruct the images in the synthetic dataset. One autoencoder follows Setup 1: ReLU activation and max pooling, and the other autoencoder follows Setup 2: softplus activation and average pooling. Since the image size is the same as the ellipsoid images, we used the same training set as in Section 5.4 of the main paper. The reconstructed images of Fig. 5.9(a) using the trained autoencoder are shown in Fig. 5.9(b).

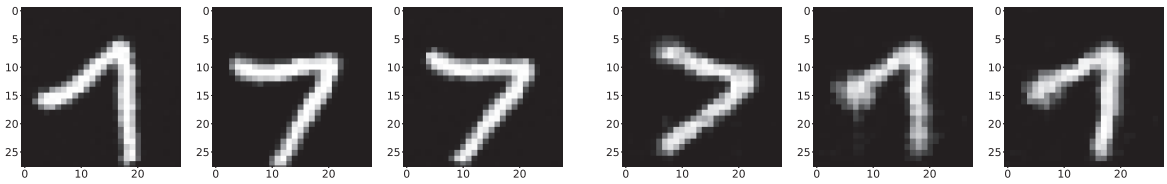
We constructed a smooth video by adding a linearly increasing Gaussian noise image to a template image. Given a template image $\mathbf{I}_0$ in the training dataset and a noise image $\mathbf{I}_e$ of the same size, where each pixel value was independently drawn from the Gaussian distribution $\mathcal{N}(0,1)$, we constructed a smooth video $\mathbf{I}(t)$ as follows:

$$\mathbf{I}(t) = \mathbf{I}_0 + \alpha(t) \cdot \mathbf{I}_e, \tag{5.9}$$

where $\alpha(t) = 10^{-2}(0.02t - 1)$, $t = 1, 2, \ldots, 100$, which is linearly increasing in time. $\mathbf{I}(t)$ is smooth because it is differentiable. This video can be regarded as image $\mathbf{I}_0$ propagating smoothly at a constant speed in an Euclidean space. To obtain statistically confident results, we additionally obtained more realizations of autoencoders. Using the same training and validation data, we repeated the training process to obtain ten autoencoders with ReLU+MaxPool, and another ten with Softplus+AvePool. We used ten template images to construct ten smooth videos based on (5.9). The ten smooth videos were reconstructed by all realizations of autoencoders, leading to a total of $10 \times 10 = 100$ reconstructed videos for ReLU+MaxPool, and another 100 for Softplus+AvePool.

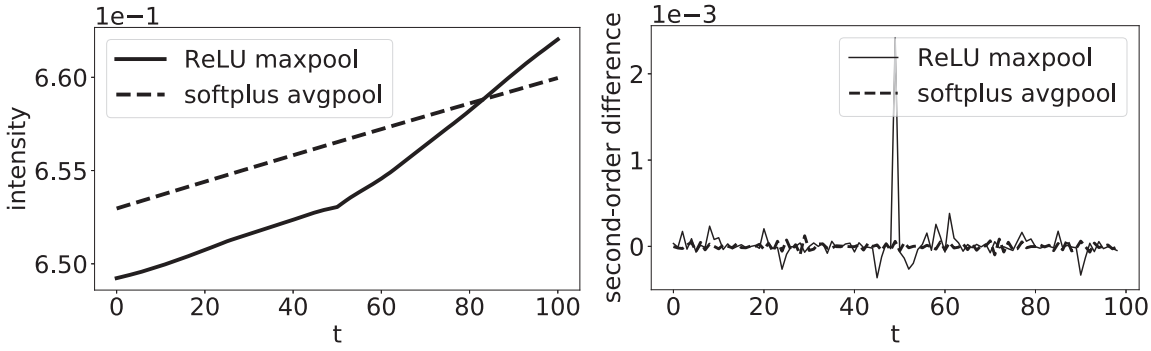
Processing Smooth Input Videos by Neural Networks For a smooth video $\mathbf{I}(t)$, we input frame by frame to a trained autoencoder to obtain the reconstructed video. For a pixel location, we compared the intensity time series of the reconstructed videos $\hat{\mathbf{I}}_{\text{ReLU+Max}}(t)$ and $\hat{\mathbf{I}}_{\text{Softplus+Ave}}(t)$ obtained using autoencoders with ReLU+MaxPool and Softplus+AvePool, respectively, as shown in Fig. 5.10(a). The second-order difference of the intensity time series were also calculated, as shown in Fig. 5.10(b). The autoencoder with ReLU+MaxPool caused larger peaks in the second-order difference curve than Softplus+AvePool given the same input video, confirming the nonsmoothness caused by ReLU+MaxPool.

We examined the distribution of AveNonSmooth of original and reconstructed videos using autoencoders with different activation functions and pooling methods, as shown in Fig. 5.11. Both autoencoders raised AveNonSmooth in the reconstructed videos, but ReLU+MaxPool raised much more than Softplus+AvePool. The slight increase of AveNonSmooth in the Softplus+AvePool setup can be attributed to the false positives due to the noisy reconstructed visual content, but it does not change the conclusion that ReLU+MaxPool leads to the nonsmoothness.



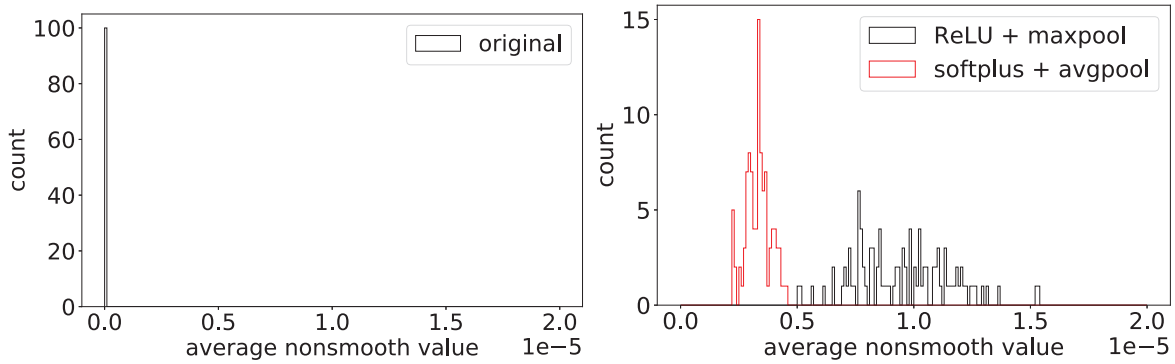
(a) (b)

Figure 5.9: (a) Three typical input images to the trained autoencoder, and (b) the corresponding reconstructed images. The reconstructed images are slightly blurred due to the encoding-decoding process.



(a) (b)

Figure 5.10: For a representative pixel location, (a) the intensity curves, and (b) the second-order difference curves of reconstructed videos. The autoencoder with ReLU+MaxPool leads to larger second-order differences than the autoencoder with Softplus+AvePool, so it should have caused more nonsmoothness.



(a) (b)

Figure 5.11: The histograms of the AveNonSmooth of the (a) original input videos and (b) reconstructed videos. The AveNonSmooth is zero for the original input videos. When using autoencoder with ReLU+MaxPool to reconstruct videos, the AveNonSmooth are larger, indicating the nonsmoothness is introduced by ReLU+MaxPool.

5.7 Modeling Transpose Convolutional Layers

We also model the nonsmoothness events in the transpose convolutional layers in the autoencoder. Similar to a convolutional layer, an SMP in the output of a transpose convolutional layer can be expressed in a weighted summation of SMPs in the input. We predicted the SMPs in the nodes of the three transpose convolutional layers of the autoencoders trained in Section 5.4 of the main paper. We compared the predicted and real data, and the R-squared values for the fitted linear models for the three transpose convolutional layers are $0.79, 0.65$, and 0.87, respectively. The large R-squared values indicate a linear relationship of SMPs between input and output of transpose convolutional layers.

5.8 Conclusion

In this work, we have shown through synthetic data that modern neural networks using ReLU and max pooling can cause nonsmoothness. We have also modeled the nonsmoothness events in the input and output of different building blocks of neural networks. To the best of our knowledge, this is the first work to investigate this understudied characteristic of the neural network, and we believe in its potential use as a generic forensic tool for regression-based neural network applications.

CHAPTER

6

INDIVIDUALIZED DEEPFAKE DETECTION USING DOUBLE OPERATIONS

6.1 Introduction

In the past few years, deepfake is one of the most rapidly developed technologies. A deepfake refers to a seemingly authentic image or video clip generated by a deep neural network. When it comes to human faces, a manipulation method may comprise reenactment, replacement, editing, and synthesis [67]. Neural network-based tools, such as FaceSwap [68] and Faceswap-GAN [69], have been made free, downloadable, and user-friendly to generate deepfake videos. Deepfakes may be put into such detrimental usages as the creation of fake news. A social science study [70] found that fake news with deepfake videos has significantly more credibility than those with only texts and photos. Due to the infodemic exacerbated by deepfake videos, developing efficient and powerful deepfake detection tools has become an increasingly important priority for the research community [67, 71, 72, 55].

Researchers have been exploring different methodologies to detect deepfakes. In the first category, the artifacts of synthetic videos are exploited for deepfake detection. Li et

al. [73] detected deepfakes using the absence of eye blinking in the synthetic videos. Yang et al. [71] found that the head poses are inconsistent in the synthetic videos. Li et al. [74] used the disparities in color components between real images and generated images for detection. In the second category, researchers investigate using end-to-end convolutional neural networks for detecting deepfake images/videos. Wang et al. [72] built a generalized neural network-based detector for images synthesized using different convolutional neural network (CNN) structures. Güera and Delp [75] used a CNN to extract frame-level features, then used a time series of features to train a recurrent neural network (RNN) that learns to classify the videos. In the third category, researchers exploit processing traces left by the neural networks for deepfake detection. Guarnera et al. [56] extracted from the spatial domain local convolutional features in modern neural networks. Durall et al. [55] built a deepfake detector using the spectral distortion caused by up-convolutions. Frank et al. [57] detected deepfake images using upsampling artifacts in the frequency domain. Our work falls under this category of exploiting processing traces for deepfake detection.

Most deepfake detectors were built to detect the whole population of deepfake videos, i.e., deepfake videos of all identities are targeted. However, victims of deepfakes are most often public figures and their deepfake videos are more detrimental due to their exposure to the public. In this work, we propose a deepfake video detecting scheme customized for an individual subject. The proposed detector can detect deepfakes of a specific individual, which is especially useful for digital journalism. For example, before reporting news based on a video of unknown authenticity about a famous person, a journalist can apply the proposed detection tool to determine its authenticity. Journalists from local newspapers, regional news stations, and university newspapers are especially in need of accessible and user-friendly deepfake detection tools.

Instead of detecting deepfake videos for the whole population, recent work also exploited characteristics of a specific person for deepfake detection. A video having inconsistent characteristics or lacking known characteristics of a target person can be regarded as a fake video. Agarwal et al. [76] targeted deepfake videos of a specific individual by capturing speaking patterns. Cozzolino et al. [77] trained using authentic videos to learn authentic temporal facial features for deepfake detection. In this work, we also target to detect deepfake videos of an individual with a focus on public figures.

Our deepfake detection method is inspired by the line of work of double JPEG compression detection [78, 79, 80, 81, 82, 83, 84, 85, 86, 87], where researchers have exploited traces in the images left by different rounds of JPEG compressions. Luo et al. [78] found that a second JPEG compression will destroy the symmetry of the matrix characterizing

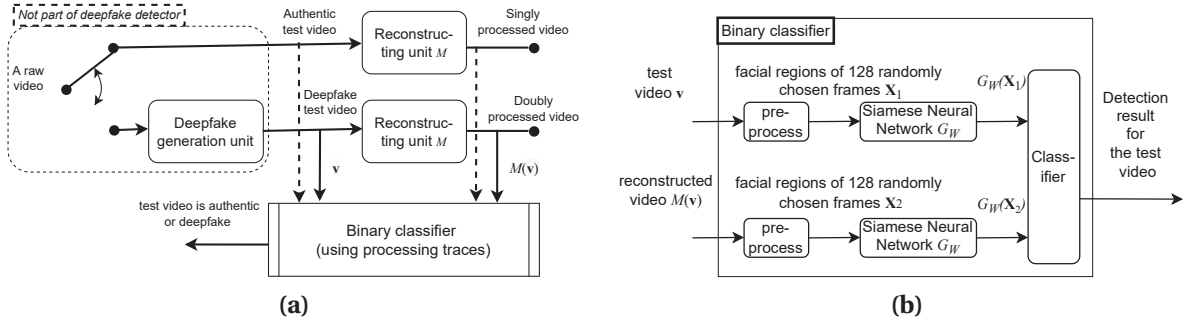


Figure 6.1: (a) The conceptual diagram of deepfake detection using double neural-network operations. For a raw authentic video, after reconstructing, the reconstructed video is singly processed; whereas for a deepfake video, the reconstructed video is doubly processed. The input to the binary classifier is a video pair, either the authentic test video and the singly processed video (dashed arrows) or the deepfake test video and the doubly processed video (solid arrows). The binary classifier will compare the videos before and after the reconstruction to decide whether the video before reconstruction is an authentic or deepfake video. (b) Details of the binary classifier. Preprocessing is first applied to obtain the facial regions of the frames. The facial regions of the frames of the input and reconstructed videos are the input to a Siamese neural network, which will learn a mapping to a manifold characterizing the processing traces. The extracted features on the processing manifold will be used to determine the deepfake detection results.

blocking artifacts. Bianchi et al. [79] studied the integer periodicity of the blockwise discrete cosine transform (DCT) coefficients for detection of double JPEG compression. Lukávs and Fridrich [80] proposed using statistical artifacts, i.e., *double peaks*, in the histogram of JPEG coefficients for detection of double JPEG compression. Huang et al. [81] proposed a random perturbation strategy to capture the difference between a singly compressed image and a doubly compressed image. Neural network-based detectors have also been introduced for the detection of aligned and non-aligned double JPEG compression. Wang et al. [86] introduced using CNNs for detection of double JPEG compression, with input to be a 1-D histogram of the DCT coefficients.

Inspired by the methodology of double JPEG compression detection, we propose to apply the idea of double operations, i.e., we use a potentially extra neural-network operation for a test video for deepfake detection, as illustrated in Fig. 6.1(a). If a test video is a deepfake video generated by a neural network such as an autoencoder, the extra neural-network operation is considered the second operation. We reconstruct the test video with the reconstruction model, which can be trained from videos of the same person as in the test video. For a deepfake input video, the frames before and after the reconstruction will both have the processing traces left by the neural networks. On the other hand, the frames in a raw video before the reconstruction do not have any processing trace of a neural network.

We detect the deepfake videos by contrasting the traces in the frames before and after the reconstruction. The contributions of this work are threefold.

- We propose to use the methodology of double operations for deepfake detection, exploiting neural-networks processing traces.
- The proposed detector empowered by manifold learning can significantly outperform the state-of-the-art end-to-end CNN classifiers.
- Individualized detectors are better suited for journalism.

6.2 Background

6.2.1 Autoencoder-based Deepfake Generating Tool

The majority of the high-quality face-swapped videos are generated by tools based on convolutional autoencoder models [88]. An autoencoder consists of two neural networks, an encoder and a decoder. The encoder f_{enc} will map the original input $\mathbf{x} \in \mathbb{R}^{d_1}$ to a lower dimensional representation $\mathbf{y} \in \mathbb{R}^{d_2}$: $\mathbf{y} = f_{\text{enc}}(\mathbf{x})$, where d_1 and d_2 are dimensions of the input and the feature space, respectively, and $d_1 > d_2$. The decoder f_{dec} will reconstruct input $\mathbf{x}$ from the lower dimensional feature $\mathbf{y}$: $\mathbf{x}' = f_{\text{dec}}(\mathbf{y})$. Denote loss function $L : \mathbb{R}^{d_1} \times \mathbb{R}^{d_1} \rightarrow \mathbb{R}$. With training data $\{\mathbf{x}_i\}_{i=1}^N$, an autoencoder can be trained by minimizing a loss, $\frac{1}{N}\sum_{i=1}^N L(\mathbf{x}_i, \mathbf{x}_i')$. The autoencoder-based deepfake generation tool consists of a shared encoder and two decoders, as shown in Fig. 6.2(a). Faceswap-GAN [89] is a representative tool based on autoencoder, publicly available, and can generate high-quality videos, therefore it can be a common tool for deepfake generation. In this work, we train Faceswap-GAN models to reconstruct videos and generate deepfake videos. The output videos of Faceswap-GAN models will have traces caused by the models.

6.2.2 Siamese Neural Networks

Fig. 6.2(b) shows the structure of Siamese neural network. Siamese neural network can learn embedding vectors that represent key points on image manifolds, and has been introduced to measure the similarity of paired input data for applications in various fields [90]. Given two input vectors $\mathbf{X}_1$ and $\mathbf{X}_2$, two neural networks with shared weights W, denoted as G_W, can be trained to learn the representations of the input, $G_W(\mathbf{X}_1)$ and $G_W(\mathbf{X}_2)$. Contrastive

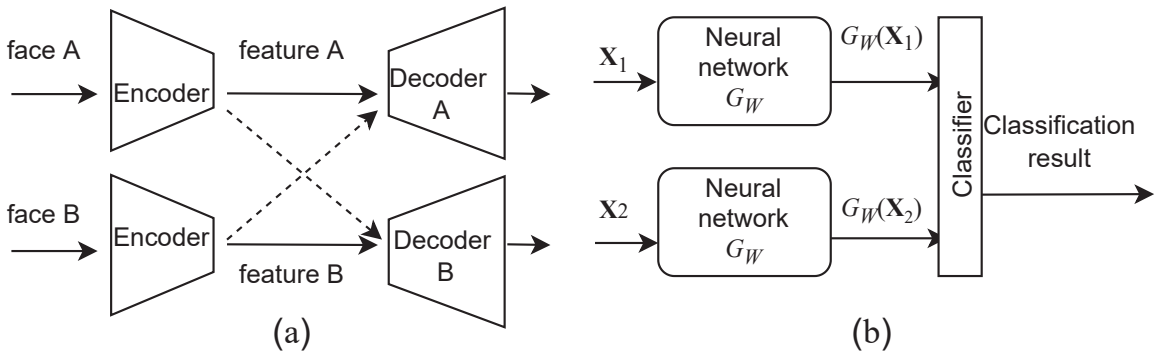


Figure 6.2: (a) A representation of autoencoder-based deepfake generator. The two encoders have shared weights and are used to extract features of the faces from both person A and B. The decoders A and B are trained to reconstruct the faces of person A and B, respectively. The dashed line represents the deepfake generation step, i.e., feature A is sent to decoder B and the reconstructed image is the face of person B with facial expressions from person A. (b) A representation of the Siamese neural network. A paired input $\mathbf{X}_1$ and $\mathbf{X}_2$ are sent to two neural networks with shared weights W. A loss function is used to compare the distance between the outputs of two networks $G_W(\mathbf{X}_1)$ and $G_W(\mathbf{X}_2)$.

loss [91] has been used as the loss function to update parameters W in the training of a Siamese neural network:

$$L(W, Y, \mathbf{X}_1, \mathbf{X}_2) = \frac{1}{2}(1-Y)D_W^2 + \frac{1}{2}Y\,[\max(0, m - D_W)]^2 \tag{6.1}$$

where $D_W(X_1, X_2) = \|G_W(X_1) - G_W(X_2)\|_2$, m is a margin, and Y is the binary label. $Y = 0$ represents that the input $\mathbf{X}_1$ and $\mathbf{X}_2$ are similar, and $Y = 1$ represents that he input $\mathbf{X}_1$ and $\mathbf{X}_2$ are different. Distance between $G_W(\mathbf{X}_1)$ and $G_W(\mathbf{X}_2)$ can be calculated to measure the similarity between input $\mathbf{X}_1$ and $\mathbf{X}_2$. In this work, we use the Siamese neural network to compute the distance between two frames before and after the reconstruction on a learned manifold.

6.3 Double Operation for Deepfake Detection

We propose to use double neural-network operations for deepfake detection and illustrate the system design in Fig. 6.1. Given a video **v** of unknown authenticity, we decide whether it is a deepfake by reconstructing it and then compare the frames before and after the reconstruction. As shown in Fig. 6.1(a), a deepfake video has already been processed by a deepfake generation neural network and reconstructing it yields a doubly processed video,

which is different from a singly processed video resulted from reconstructing a raw video. In this work, we focus on detecting deepfake videos of a specific person, such as public figures since they are common targets of deepfake attacks. We assume that it is easy to visually recognize the person in $\mathbf{v}$ and to manually collect more videos for this particular person to train the reconstruction model M. The frames before and after reconstruction will be sent to a classifier with a manifold mapping unit to decide whether the input is a deepfake.

A Siamese neural network G_W can facilitate measuring the "distance" between the frames before and after the processing/reconstruction for binary decision, as shown in Fig. 6.1(b). The Siamese neural network aims to learn a mapping function to a manifold that allows powerful discrimination between the frames with and without processing traces. Let us denote $\mathbf{X}_1$ to be frames of the facial region from the input video $\mathbf{v}$, and $\mathbf{X}_2$ to be those from the reconstructed video $M(\mathbf{v})$. Contrastive loss [91] is used as the loss function to update parameters W while training the Siamese neural network G_W:

$$L(W,Y,\mathbf{X}_1,\mathbf{X}_2) = (1-Y)D_W^2 + Y[\max(0, m-D_W)]^2 \tag{6.2}$$

where $D_W(\mathbf{X}_1,\mathbf{X}_2) = \|G_W(\mathbf{X}_1) - G_W(\mathbf{X}_2)\|_2$ is the "distance" between the frames before and after the reconstruction, $m > 0$ is a margin, and $Y \in \{0,1\}$ is a known binary label indicating whether $\mathbf{X}_1$ and $\mathbf{X}_2$ belong to the same class. The hinge loss $\max(0, m - D_W)$ is used, where it will be zero when D_W exceeds the margin, i.e., $D_W > m$. When input video is a deepfake video, $\mathbf{X}_1$ and $\mathbf{X}_2$ are both deepfake frames, i.e., $\mathbf{X}_1$ and $\mathbf{X}_2$ are in the same "deepfake" class with binary label $Y = 0$. When the network G_W is correctly trained to extract processing traces on the processing manifold left by neural networks, the distance between $G_W(\mathbf{X}_1)$ and $G_W(\mathbf{X}_2)$ is expected to be small since both $\mathbf{X}_1$ and $\mathbf{X}_2$ contain processing traces. When input video is a raw video, $\mathbf{X}_1$ contains raw frames while $\mathbf{X}_2$ contains deepfake frames, i.e., $\mathbf{X}_1$ and $\mathbf{X}_2$ are in different classes with binary label $Y = 1$. The distance between $G_W(\mathbf{X}_1)$ and $G_W(\mathbf{X}_2)$ on the processing manifold is expected to be large since only $\mathbf{X}_2$ contains traces left by neural networks but $\mathbf{X}_1$ does not.

(a) **(b)**

Figure 6.3: (a) Face regions from raw frames (first row) and the reconstructed frames (second row). The reconstructed frames are singly processed frames. (b) Face regions from deepfake frames (first row) and the reconstructed frames (second row). The reconstructed frames are doubly processed. Since the reconstructing neural networks are trained with videos from the same person, the reconstructing quality is good for both raw and deepfake frames.

6.4 Experimental Results

6.4.1 Dataset Creation and Intermediate Videos Generation

In this work, we collected from YouTube a total of 50 authentic videos of ten famous people, such as Roger Federer and Ban Ki-moon. Five authentic/raw videos were collected for each person j, denoted as $\{\mathbf{v}^{\mathrm{r}}_{ij}\}_{i=1}^{5}$, for $j = 1,...,10$. For each of the authentic video $\mathbf{v}^{\mathrm{r}}_{ij}$, we trained a Faceswap-GAN model using $\mathbf{v}^{\mathrm{r}}_{ij}$ and a video of another identity to obtain a deepfake video, denoted as $\mathbf{v}^{\mathrm{d}}_{ij}$. 40,000 iterations of training were used to ensure a good visual quality of deepfake videos. We will evaluate the proposed double-operations method on our collected dataset of public figures, instead of existing deepfake datasets that mainly comprises non-public figures. Identifying non-public figures to obtain more of their videos for training reconstruction models is impractical.

For each video $\mathbf{v}_{ij}$, we reconstruct the input video with the reconstruction model M_j to obtain an input video pair, i.e., $\{(\mathbf{v}_{ij}, M_j(\mathbf{v}_{ij})\}_{i=1}^{5}$. Examples of facial regions from the raw and deepfake frames along with the reconstructed frames are shown in Fig. 6.3. For each video pair, 128 collected frame pairs were randomly chosen as input to a pair of shared Siamese neural networks. A larger number of frames from each video is preferred, as shown in Fig. 6.6.

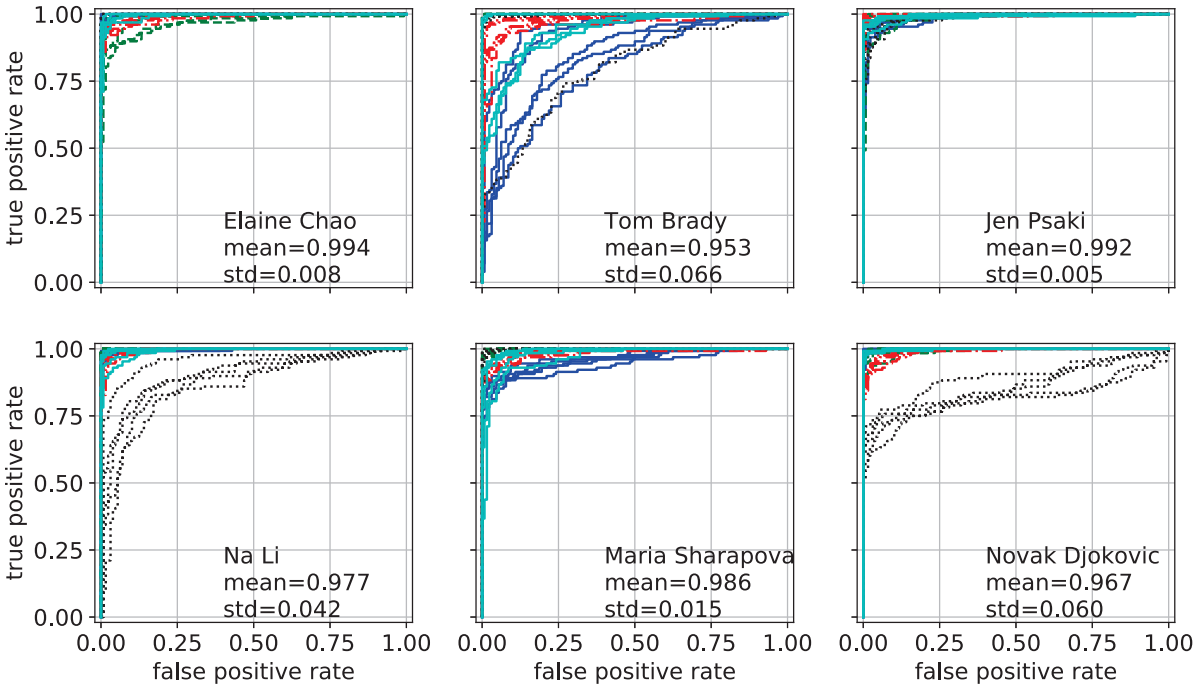


Figure 6.4: ROC curves for deepfake detection using the proposed neural-network double-operations method leveraging the Siamese neural network for manifold learning of processing traces. Each plot contains the results from a public figure and each color represents a different index of test data. The AUC values are large with small standard deviations, indicating good performance.

6.4.2 Deepfake Detection Results

In this subsection, we evaluate the double-operations method for deepfake detection. We customized a unique Siamese neural network-based detector to detect deepfake videos for each person separately. The Siamese neural network G_W has two parts: 1) the EfficientNetAutoAttB4ST [92] network pretrained on the DFDC dataset [88] except for the last layer, and 2) a fully connected layer. The DFDC contains deepfake videos generated by various methods [68, 69, 93, 94], and the first layers EfficientNetAutoAttB4ST trained on it is a powerful feature extractor [92]. The first part is used to extract features of the frames and the weights will not be updated in the training. The second part will be updated to learn the best manifold characterizing the processing traces for deepfake detection. The output $G_W(\mathbf{X})$ is a vector of length 50, which is a large enough dimension to characterize the processing traces as evidenced experimentally. Per Section 6.3, when input $\mathbf{X}_1$ is a raw or a deepfake frame, the binary label Y is set to 1 or 0, respectively. We used Adam optimizer and the learning rate was determined by the grid search.

For person j's videos, we trained a separate Siamese neural network and therefore a

Table 6.1: The deepfake detection performance of the proposed double-operations method and end-to-end CNN classifiers.

Method	Tuned with our dataset	AUC mean (std)	p-value
Xception [95]	✗	0.798 (0.33)	10^{-12}
EfficientNetAutoAttB4ST[92]	✗	0.807 (0.23)	10^{-47}
Xception[95]	✓	0.930 (0.11)	0.008
EfficientNetAutoAttB4ST[92]	✓	0.949 (0.10)	0.009
Proposed method	✓	**0.974** (**0.05**)	

different deepfake detector. During each training session, one video pair in $\{(\mathbf{v}_{ij}, M_j(\mathbf{v}_{ij})\}_{i=1}^{5}$ was chosen as the test data, and another pair was randomly chosen as the validation data, with the rest three pairs as the training data. For each test data, we repeated this process 5 times to capture the statistical behavior. In each training session, the neural network with the smallest validation loss was chosen as the final network for the test dataset. We used the $\|G_W(\mathbf{X}_1) - G_W(\mathbf{X}_2)\|_2$ as the test statistic for deepfake detection; the detection results on the test dataset for six of the ten public figures are shown in Fig. 6.4. The averaged AUC values of all public figures in different test videos is 0.974 and the sample standard deviation is 0.05.

We compared the performance of the state-of-the-art end-to-end neural network-based detectors with our proposed method. We used the Xception [95] network trained on the FaceForensics++ dataset [96] with deepfake videos generated by four methods including Faceswap [68]; and the EfficientNetAutoAttB4ST [92] network trained on the DFDC dataset [88], a dataset consisting of deepfake videos generated by various popular face-swapping methods, such as Facewap-GAN [69], StyleGAN [93], Faceswap [68], and NTH [94]. When directly applying the pretrained networks to detect deepfake videos in our dataset, the averaged of AUC is only 0.798 and the standard deviation is 0.33 when using Xception, and the average of AUC is only 0.807 and the standard deviation is 0.23 when using Efficient-NetAutoAttB4ST, as summarized in TABLE 6.1. Their weak performance may be explained by the fact that the detectors were pretrained with other datasets and do not generalize well to our dataset. When a journalist directly uses an off-the-shelf neural network detector, the detection results are untrustworthy due to the limited generalization capability of the detector.

We further fine-tuned off-the-shelf neural networks with our dataset to improve its performance, assuming that a journalist can first collect a lot of videos of different identities

and then fine-tune the off-the-shelf model with the collected videos. We tuned the last fully connected layer, with the weights in other layers fixed for feature extraction. When we tuned an off-the-shelf neural network for an individual public figure, the neural network did not generalize well to test data. We, therefore, tuned the off-the-shelf neural networks on videos of all the public figures. In the fine-tuning, we used the Adam optimizer and used grid search to determine the learning rate. The averaged AUC in different test videos has increased to 0.949 with standard deviation of 0.10 when using the EfficientNetAutoAttB4ST network, and increased to 0.930 with standard deviation of 0.11 when using the Xception network, as summarized in TABLE 6.1. The detection results for several public figures after tuning the pretrained EfficientNetAutoAttB4ST network are shown in Fig. 6.5. The larger variance of the AUC values makes the off-the-shelf neural network less attractive in the real-world applications. We also used t-tests to show that the proposed method are significantly better than those of the off-the-shelf detectors in AUC, with the p-values summarized in TABLE 6.1.

Although end-to-end CNNs are trained to detect all deepfake videos, they may not be able to generalize well to new videos, as evidenced by the larger standard deviation of the AUC values. In our proposed method, a test video is processed by a reconstruction unit. The Siamese neural network is subsequently trained to contrast the processing traces in videos before (if any) and after reconstruction via manifold learning, which yields better discriminative capability than end-to-end classifiers trained only on deepfake and raw videos.

Compared to end-to-end CNN-based classifiers, our proposed method targets deepfake detection for individuals, with main applications on public figures. Although our method needs additional training of the reconstruction models, the training can be done in advance for each famous public figure before deepfake detection is needed. For example, the journalists can train the reconstruction models for different public figures well in advance before they need to use videos for their reporting tasks. Journalists may also shared trained detectors of public figures via their professional networks. Also, using the proposed method, a journalist only needs to train the model using videos of the specific public figure. When tuning end-to-end neural network classifiers, significantly more videos of people of different looking are needed to avoid overfitting. The proposed method is therefore a more effective and efficient choice for journalists to use for deepfake detection.

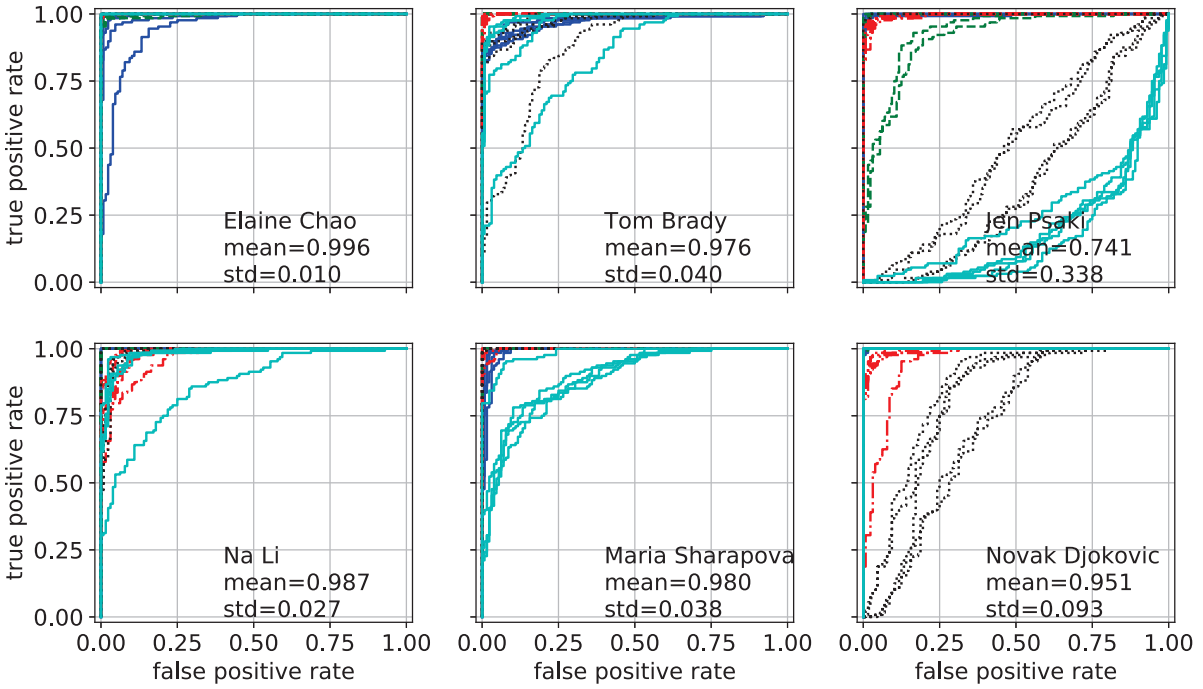


Figure 6.5: ROC curves on our dataset when detecting deepfake videos using a benchmark neural network structure. The overall results are worse than the proposed double-operations method in terms of AUC.

6.4.3 Ablation Studies

In the pipeline of Fig. 6.1 of the main body of the paper, we randomly chose 128 frames from each video to train the Siamese neural network. We varied the number of frames chosen from each video and compared the detection performance in AUC, as shown in Fig. 6.6. A larger number of frames in general results in larger AUC values with smaller standard deviations. Also, the proposed method outperforms the off-the-shelf EfficientNetAutoAttB4ST structure in terms of AUC values.

We set the dimension of output of the Siamese neural network G_W to be 50, i.e., $G_W(X)$ is a vector with length of 50. We varied the structure of G_W, and compared the effect of the dimension of the output on the detection performance in terms of AUC, as shown in Fig. 6.7. A Siamese neural network with small output dimension has worse detection performance in terms of AUC, because the dimension of the learned manifold is not large enough to characterize the processing traces.

In the proposed double-operation method, the Siamese neural network has been used to contrast the processing traces in the videos, where the contrastive loss was used as the loss function in the training. To study the effect of the contrastive learning, we replace the

Siamese neural network with a neural network based classifier H_W. Denote the first layers of the pretrained EfficientNetAutoAttB4ST neural network to be B_W, the input to the classifier H_W has two channels, $B_W(\mathbf{X}_1)$ and $B_W(\mathbf{X}_2)$, the extracted features in frames before and after reconstruction. The classifier H_W has two convolutional layers and a fully connected layer, with output size to be 1. The cross entropy loss was used as the loss function when training the classifier H_W. The averaged AUC values when detecting deepfake videos of different public figures was 0.651 with a standard deviation to be 0.23. The bad detection performance indicates that a general CNN structure cannot capture processing traces in the frames before and after reconstruction. The contrastive learning using the Siamese neural network is efficient in exploiting processing traces for deepfake detection.

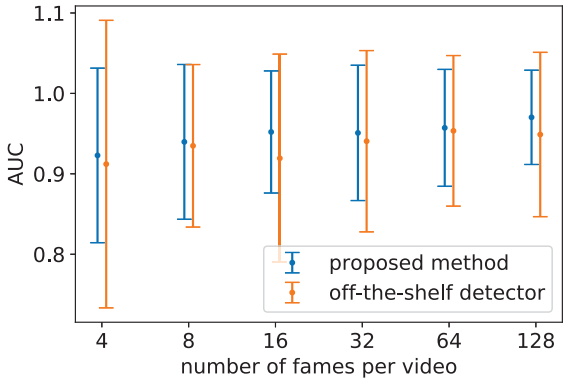


Figure 6.6: The effect of the number of frames used for detection per video on the performance in terms of AUC for the proposed method and off-the-shelf EfficientNetAutoAttB4ST neural network structure. Error bars correspond to one sample standard deviation above and below the average of the AUC values. Using more frames will result in larger AUC values and smaller variance.

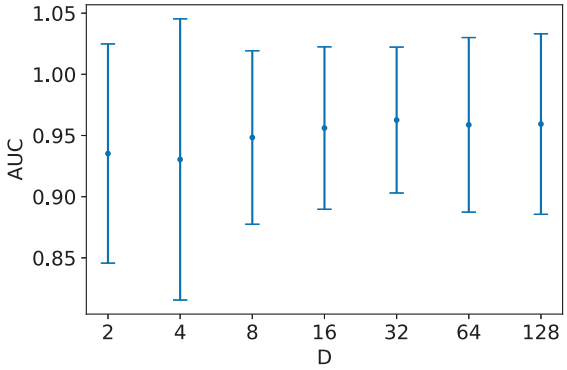


Figure 6.7: The effect of the number of dimensions of the learned manifold from the Siamese neural network G_W. Although a smaller dimension makes the model simpler, but generally a larger dimension will result in larger AUC values and smaller variance.

6.5 Discussion

In the current work, the reconstruction model is the same as the deepfake generation tool, i.e., Faceswap-GAN, one of the most popular and effective off-the-shelf tools for deepfake generation. When the reconstruction model does not match the tool used for deepfake generation, the processing traces before and after reconstruction may be different, but both traces are left by neural networks. For example, the reconstruction model is a Faceswap-GAN model while the deepfake video is generated by a Faceswap model. A more sophisticated classifier may be needed to exploit the processing traces left by different neural network structures. Under the proposed double operations framework, a more general Siamese neural network for processing-trace manifold learning can still be obtained using deepfake videos synthesized by different tools and reconstruction models of different neural network structures.

6.6 Conclusion and Future Work

In this work, we had used the method of double neural-network operations for deepfake detection of specific individuals. The proposed detector based on learning a processing-trace manifold can achieve better detection performance than end-to-end CNN-based detectors on our collected dataset of famous people. In future work, we plan to extend the double-operations detection to the scenarios with mismatched neural network architectures.

CHAPTER

7

CONCLUSION AND FUTURE WORK

With the rapid development of mobile phones and computer vision technology, there are more opportunities to use camera-captured images for authentication tasks. Also, there appears increasing threat from deepfake videos with the fast development of image synthesizing technology. This dissertation had investigated extracting micro signals from images acquired by a camera/scanner to authenticate paper patches or IC chips. Also, deepfake detection using the traces left by the neural network had been studied.

We had looked into surface-based authentication applications for paper patches and IC chip surfaces. In Chapter 2, we had proposed two enhanced geometric reflection models that take into ambient light in addition to diffuse reflection light. We had also proposed more powerful discriminative physical features, i.e., high spatial frequency subbands for authentication of paper surfaces. In Chapter 3, we had investigated more details of paper surface-based authentication systems using flatbed scanners, which had better controlled experimental environments. In Chapter 4, we had re-purposed the paper surface-based authentication system for the IC chips. We had also developed fast authentication schemes to authenticate IC chips with camera-captured videos.

In future work, it is desirable to develop algorithms to authenticate objects of a curved surface, such as the label on the bottles of fine wine. Current authentication systems focus

on flat paper patches and IC chip surfaces. When using a mobile camera to capture images, the effect of specular refection in the camera-captured images should be investigated. Also, it will be more flexible to capture videos of IC chip surfaces with hand-holding mobile cameras. Such a less-controlled scenario will require efficient algorithms to accurately register the frames in the video.

We had also investigated deepfake detection based on the traces left by neural networks. We had looked into the nonsmoothness caused by ReLU and max pooling in the neural networks. We had confirmed the existence of nonsmoothness and modeled its behavior in the propagation through layers of neural networks. Inspired by double JPEG compression detection, we had proposed a double operation method to exploit traces left by neural networks for deepfake detection. We targeted to detect deepfake videos of public figures, and the detection results had outperformed the state-of-the-art methods.

To better exploit the traces left by neural networks, it is desirable to collect a larger dataset with deepfake videos generated by different neural network structures. The Siamese neural network trained on the enlarged dataset may be able to extract more general traces left by the neural network. Also, more powerful classifiers are desired for deepfake detection.

REFERENCES

[1] C. Khor, M. Z. Abdullah, C.-S. Lau, and I. Azid, "Recent fluid-structure interaction modeling challenges in IC encapsulation–A review," *Microelectronics Reliability*, vol. 54, no. 8, pp. 1511–1526, 2014.

[2] M. Miura, "Compression molding solutions for various high end package and cost savings for standard package applications," in *International Conference on Electronics Packaging (ICEP)*, 2016, pp. 243–247.

[3] Y. D. Kariakin, "Authentication of articles," Oct. 7 1997, patent WO9724699A1.

[4] J. D. Buchanan, R. P. Cowburn, A.-V. Jausovec, D. Petit, P. Seem, G. Xiong, D. Atkinson, K. Fenton, D. A. Allwood, and M. T. Bryan, "Forgery: "Fingerprinting" documents and packaging," *Nature*, vol. 436, no. 7050, p. 475, 2005.

[5] F. Beekhof, S. Voloshynovskiy, O. Koval, R. Villán, and T. Pun, "Secure surface identification codes," in *Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*, vol. 6819. International Society for Optics and Photonics, 2008, p. 68190D.

[6] W. Clarkson, T. Weyrich, A. Finkelstein, N. Heninger, J. Halderman, and E. Felten, "Fingerprinting blank paper using commodity scanners," in *Proc. IEEE Symposium on Security and Privacy*, Berkeley, CA, May 2009, pp. 301–314.

[7] S. Voloshynovskiy, M. Diephuis, F. Beekhof, O. Koval, and B. Keel, "Towards reproducible results in authentication based on physical non-cloneable functions: The forensic authentication microstructure optical set (FAMOS)," in *Proc. IEEE International Workshop on Information Forensics and Security*, Tenerife, Spain, Dec. 2012, pp. 43–48.

[8] M. Diephuis and S. Voloshynovskiy, "Physical object identification based on FAMOS microstructure fingerprinting: Comparison of templates versus invariant features," in *Proc. International Symposium on Image and Signal Processing and Analysis*, Trieste, Italy, Sep. 2013, pp. 119–123.

[9] M. Diephuis, S. Voloshynovskiy, T. Holotyak, N. Stendardo, and B. Keel, "A framework for fast and secure packaging identification on mobile phones," in *Proc. SPIE, Media Watermarking, Security, and Forensics*, San Francisco, CA, Feb. 2014, p. 90280T.

[10] C.-W. Wong and M. Wu, "A study on PUF characteristics for counterfeiting detection," in *Proc. IEEE International Conference on Image Processing*, Quebec City, Canada, Sep. 2015, pp. 1643–1647.

[11] ——, "Counterfeit detection based on unclonable feature of paper using mobile camera," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 8, pp. 1885–1899, Apr. 2017.

[12] C. Kauba, L. Debiasi, R. Schraml, and A. Uhl, "Towards drug counterfeit detection using package paperboard classification," in *Pacific Rim Conference on Multimedia*. Springer, 2016, pp. 136–146.

[13] R. Schraml, L. Debiasi, C. Kauba, and A. Uhl, "On the feasibility of classification-based product package authentication," in *IEEE Workshop on Information Forensics and Security*. IEEE, 2017, pp. 1–6.

[14] R. Szeliski, *Computer Vision: Algorithms and Applications*. Springer, 2010, ch. 2.2.

[15] B. C. Levy, *Principles of Signal Detection and Parameter Estimation*. Springer, 2008.

[16] J. L. Devore, *Probability and Statistics for Engineering and the Sciences*. Cengage Learning, 2015.

[17] B. Efron and R. J. Tibshirani, *An Introduction to the Bootstrap*, ser. Monographs on Statistics & Applied Probability. Chapman & Hall/CRC, 1994.

[18] "The paper project," Retrieved Feb. 2019. [Online]. Available: http://www.paperproject.org/

[19] C.-W. Wong and M. Wu, "A study on PUF characteristics for counterfeiting detection," in *IEEE International Conference on Image Processing*, Quebec City, Canada, Sep. 2015, pp. 1643–1647.

[20] ——, "Counterfeit detection using paper PUF and mobile cameras," in *Proc. IEEE International Workshop on Information Forensics and Security*, Rome, Italy, Nov. 2015.

[21] ——, "Counterfeit detection based on unclonable feature of paper using mobile camera," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 8, pp. 1885–1899, Apr. 2017.

[22] R. Liu, C.-W. Wong, and M. Wu, "Enhanced geometric reflection models for paper surface based authentication," in *IEEE International Workshop on Information Forensics and Security*, Hong Kong, Dec. 2018.

[23] A. Sharma, L. Subramanian, and E. A. Brewer, "Paperspeckle: Microscopic fingerprinting of paper," in *ACM Conference on Computer and Communications Security*. ACM, 2011, pp. 99–110.

[24] E. Toreini, S. F. Shahandashti, and F. Hao, "Texture to the rescue: Practical paper fingerprinting based on texture patterns," *ACM Transactions on Privacy and Security (TOPS)*, vol. 20, no. 3, p. 9, 2017.

[25] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge University Press, 2003.

[26] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[27] T. Lindeberg, "Scale-space theory: A basic tool for analyzing structures at different scales," *Journal of applied statistics*, vol. 21, no. 1-2, pp. 225–270, 1994.

[28] P. Kovesi, "Shapelets correlated with surface normals produce surfaces," in *IEEE International Conference on Computer Vision*, Beijing, China, Oct. 2005, pp. 994–1001.

[29] M.-C. Beland and J. M. Bennett, "Effect of local microroughness on the gloss uniformity of printed paper surfaces," *Applied Optics*, vol. 39, no. 16, pp. 2719–2726, Jun. 2000.

[30] J. F. Kenney and E. Keeping, *Mathematics of Statistics, Vol. II.* New York: D. Van Nostrand Co. Inc, 1951.

[31] J. L. Devore, *Probability and Statistics for Engineering and the Sciences.* Cengage Learning, 2011.

[32] J. Villasenor and M. Tehranipoor, "Chop shop electronics," *IEEE Spectrum*, vol. 50, no. 10, pp. 41–45, 2013.

[33] M. Roel, "Physically unclonable functions: Constructions, properties and applications," *Katholieke Universiteit Leuven, Belgium*, 2012.

[34] Y. Gao, S. F. Al-Sarawi, and D. Abbott, "Physical unclonable functions," *Nature Electronics*, vol. 3, no. 2, pp. 81–91, 2020.

[35] R. Pappu, B. Recht, J. Taylor, and N. Gershenfeld, "Physical one-way functions," *Science*, vol. 297, no. 5589, pp. 2026–2030, 2002.

[36] R. Liu and C.-W. Wong, "On microstructure estimation using flatbed scanners for paper surface-based authentication," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 3039–3053, 2021.

[37] K. Lofstrom, W. R. Daasch, and D. Taylor, "IC identification circuit using device mismatch," in *IEEE International Solid-State Circuits Conference*, 2000, pp. 372–373.

[38] R. Helinski, D. Acharyya, and J. Plusquellic, "A physical unclonable function defined using power distribution system equivalent resistance variations," in *46th ACM/IEEE Design Automation Conference*, 2009, pp. 676–681.

[39] B. Gassend, D. Clarke, M. Van Dijk, and S. Devadas, "Silicon physical random functions," in *9th ACM Conference on Computer and Communications Security*, 2002, pp. 148–160.

[40] J. W. Lee, D. Lim, B. Gassend, G. E. Suh, M. Van Dijk, and S. Devadas, "A technique to build a secret key in integrated circuits for identification and authentication applications," in *Symposium on VLSI Circuits*, 2004, pp. 176–179.

[41] H. Patel, Y. Kim, J. T. McDonald, and L. Starman, "Increasing stability and distinguishability of the digital fingerprint in FPGAs through input word analysis," in *International Conference on Field Programmable Logic and Applications*, 2009, pp. 391–396.

[42] D. E. Holcomb, W. P. Burleson, K. Fu *et al.*, "Initial SRAM state as a fingerprint and source of true random numbers for RFID tags," in *Conference on RFID Security*, vol. 7, no. 2, 2007.

[43] D. Yamamoto, K. Sakiyama, M. Iwamoto, K. Ohta, T. Ochiai, M. Takenaka, and K. Itoh, "Uniqueness enhancement of PUF responses based on the locations of random outputting RS latches," in *International Workshop on Cryptographic Hardware and Embedded Systems*, 2011, pp. 390–406.

[44] R. Maes, P. Tuyls, and I. Verbauwhede, "Intrinsic PUFs from flip-flops on reconfigurable devices," in *3rd Benelux Workshop on Information and System Security (WISSec)*, vol. 17, 2008, p. 2008.

[45] P. Simons, E. van der Sluis, and V. van der Leest, "Buskeeper PUFs, a promising alternative to D flip-flop PUFs," in *IEEE International Symposium on Hardware-Oriented Security and Trust*, 2012, pp. 7–12.

[46] L. M. Siong and C. Y. Tat, "Effect of the laser parameters, epoxy mold compound properties and mold tool surface finishing on mark legibility of encapsulated IC package," in *IEEE 20th Electronics Packaging Technology Conference (EPTC)*, 2018, pp. 652–656.

[47] B. S. Reddy and B. N. Chatterji, "An FFT-based technique for translation, rotation, and scale-invariant image registration," *IEEE Transactions on Image Processing*, vol. 5, no. 8, pp. 1266–1271, 1996.

[48] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," 2015.

[49] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, "Facial landmark detection by deep multi-task learning," in *ECCV*, 2014, pp. 94–108.

[50] S. Mahendran, H. Ali, and R. Vidal, "3D pose regression using convolutional neural networks," in *ICCV Workshops*, 2017, pp. 2174–2182.

[51] S. Miao, Z. J. Wang, Y. Zheng, and R. Liao, "Real-time 2D/3D registration via CNN regression," in *Int. Symp. Biomed. Imag.*, 2016, pp. 1430–1434.

[52] M. U. Gudelek, S. A. Boluk, and A. M. Ozbayoglu, "A deep learning based stock trading model with 2-D CNN trend detection," in *Symp. Ser. Comp. Intell.*, 2017.

[53] Y. Mirsky and W. Lee, "The creation and detection of deepfakes: A survey," *arXiv preprint arXiv:2004.11138*, 2020.

[54] U. A. Ciftci, I. Demir, and L. Yin, "FakeCatcher: Detection of synthetic portrait videos using biological signals," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020.

[55] R. Durall, M. Keuper, and J. Keuper, "Watch your up-convolution: CNN based generative deep neural networks are failing to reproduce spectral distributions," in *IEEE/CVF Conf. Comput. Vision Pattern Recog.*, Jun. 2020, pp. 7890–7899.

[56] L. Guarnera, O. Giudice, and S. Battiato, "Deepfake detection by analyzing convolutional traces," in *IEEE/CVF Conf. Comput. Vision Pattern Recog. Workshops*, 2020, pp. 666–667.

[57] J. Frank, T. Eisenhofer, L. Schönherr, A. Fischer, D. Kolossa, and T. Holz, "Leveraging frequency analysis for deep fake image recognition," in *Int. Conf. Mach. Learn.*, Jul. 2020.

[58] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Int. Conf. Artif. Intell. Stat.*, 2011, pp. 315–323.

[59] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *NIPS*, 2012, pp. 1097–1105.

[60] C. Dugas, Y. Bengio, F. Bélisle, C. Nadeau, and R. Garcia, "Incorporating second-order functional knowledge for better option pricing," in *NIPS*, 2001, pp. 472–478.

[61] F. Saeedan, N. Weber, M. Goesele, and S. Roth, "Detail-preserving pooling in deep networks," in *CVPR*, 2018, pp. 9108–9116.

[62] Y.-L. Boureau, J. Ponce, and Y. LeCun, "A theoretical analysis of feature pooling in visual recognition," in *ICML*, 2010, pp. 111–118.

[63] R. Zhang, "Making convolutional networks shift-invariant again," in *Int. Conf. Mach. Learn.*, 2019.

[64] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural networks*, vol. 61, pp. 85–117, 2015.

[65] B. Cheng and D. M. Titterington, "Neural networks: A review from a statistical perspective," *Stat. Sci.*, pp. 2–30, 1994.

[66] Y. LeCun, C. Cortes, and C. Burges, "MNIST handwritten digit database," *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, vol. 2, 2010.

[67] Y. Mirsky and W. Lee, "The creation and detection of deepfakes: A survey," *ACM Computing Surveys*, vol. 54, no. 1, pp. 1–41, 2021.

[68] "FaceSwap," *Available: https://faceswap.dev*, Oct. 2018.

[69] "FaceSwap-GAN," *Available: https://github.com/shaoanlu/faceswap-GAN*, Aug. 2018.

[70] J. Lee and S. Y. Shin, "Something that they never said: Multimodal disinformation and source vividness in understanding the power of ai-enabled deepfake news," *Media Psychology*, pp. 1–16, 2021.

[71] X. Yang, Y. Li, and S. Lyu, "Exposing deep fakes using inconsistent head poses," in *IEEE Int. Conf. Acoust., Speech, Signal Process.*, Brighton, UK, May. 2019, pp. 8261–8265.

[72] S.-Y. Wang, O. Wang, R. Zhang, A. Owens, and A. A. Efros, "CNN-generated images are surprisingly easy to spot... for now," in *IEEE/CVF Conf. Comput. Vision Pattern Recog.*, vol. 7, 2020.

[73] Y. Li, M.-C. Chang, and S. Lyu, "In Ictu Oculi: Exposing AI created fake videos by detecting eye blinking," in *IEEE Int. Workshop Informat. Forensics Security*. Hong Kong: IEEE, Dec. 2018.

[74] H. Li, B. Li, S. Tan, and J. Huang, "Detection of deep network generated images using disparities in color components," *arXiv preprint arXiv:1808.07276*, 2018.

[75] D. Güera and E. J. Delp, "Deepfake video detection using recurrent neural networks," in *15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, Auckland, New Zealand, Nov. 2018.

[76] S. Agarwal, H. Farid, Y. Gu, M. He, K. Nagano, and H. Li, "Protecting world leaders against deep fakes." in *IEEE/CVF Conf. Comput. Vision Pattern Recog. Workshops*, Long Beach, CA, Jun. 2019.

[77] D. Cozzolino, A. Rössler, J. Thies, M. Nießner, and L. Verdoliva, "Id-reveal: Identity-aware deepfake video detection," in *IEEE/CVF Int. Conf. Comput. Vision*, Sep. 2021, pp. 15 108–15 117.

[78] W. Luo, Z. Qu, J. Huang, and G. Qiu, "A novel method for detecting cropped and recompressed image block," in *IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 2, Honolulu, Hawaii, Apr. 2007, pp. II–217.

[79] T. Bianchi and A. Piva, "Detection of nonaligned double JPEG compression based on integer periodicity maps," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 2, pp. 842–848, Apr. 2011.

[80] J. Lukavs and J. Fridrich, "Estimation of primary quantization matrix in double compressed JPEG images," in *Digital Forensic Research Workshop*, Cleveland, Ohio, Aug. 2003, pp. 5–8.

[81] F. Huang, J. Huang, and Y. Q. Shi, "Detecting double JPEG compression with the same quantization matrix," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 4, pp. 848–856, Dec. 2010.

[82] D. Fu, Y. Q. Shi, and W. Su, "A generalized benford's law for JPEG coefficients and its applications in image forensics," in *Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505. International Society for Optics and Photonics, 2007, p. 65051L.

[83] C. Chen, Y. Q. Shi, and W. Su, "A machine learning based scheme for double JPEG compression detection," in *19th International Conference on Pattern Recognition*. IEEE, 2008, pp. 1–4.

[84] S. Lai and R. Böhme, "Block convergence in repeated transform coding: JPEG-100 forensics, carbon dating, and tamper detection," in *IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2013, pp. 3028–3032.

[85] J. Yang, J. Xie, G. Zhu, S. Kwong, and Y.-Q. Shi, "An effective method for detecting double JPEG compression with the same quantization matrix," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 11, pp. 1933–1942, Nov. 2014.

[86] Q. Wang and R. Zhang, "Double JPEG compression forensics based on a convolutional neural network," *EURASIP Journal on Information Security*, vol. 2016, no. 1, pp. 1–12, 2016.

[87] J. Park, D. Cho, W. Ahn, and H.-K. Lee, "Double JPEG detection in mixed JPEG quality factors using deep convolutional neural network," in *European Conference on Computer Vision (ECCV)*, 2018, pp. 636–652.

[88] B. Dolhansky, J. Bitton, B. Pflaum, J. Lu, R. Howes, M. Wang, and C. C. Ferrer, "The deepfake detection challenge dataset," *arXiv preprint arXiv:2006.07397*, 2020.

[89] "FaceSwap-GAN," Aug. 2018. [Online]. Available: https://github.com/shaoanlu/faceswap-GAN

[90] D. Chicco, "Siamese neural networks: An overview," *Artificial Neural Networks*, pp. 73–94, 2021.

[91] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *IEEE/CVF Conf. Comput. Vision Pattern Recog.* New York, NY: IEEE, Jun. 2006, pp. 1735–1742.

[92] N. Bonettini, E. D. Cannas, S. Mandelli, L. Bondi, P. Bestagini, and S. Tubaro, "Video face manipulation detection through ensemble of CNNs," in *IEEE Int. Conf. Learn. Pattern*, Jan. 2020.

[93] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *IEEE/CVF Conf. Comput. Vision Pattern Recog.*, 2019, pp. 4401–4410.

[94] E. Zakharov, A. Shysheya, E. Burkov, and V. Lempitsky, "Few-shot adversarial learning of realistic neural talking head models," in *IEEE/CVF Int. Conf. Comput. Vision*, 2019, pp. 9459–9468.

[95] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *IEEE/CVF Conf. Comput. Vision Pattern Recog.*, 2017, pp. 1251–1258.

[96] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "Faceforensics++: Learning to detect manipulated facial images," in *IEEE/CVF Int. Conf. Comput. Vision*, 2019, pp. 1–11.