Detection of counterfeit coins based on 3D Height-Map Image Analysis

Saeed Khazaee

A Thesis
In the Department of
Computer Science and Software Engineering

Presented in Partial Fulfillment of the Requirements
For the Degree of
Doctor of Philosophy in Computer Science

Concordia University
Montreal, Quebec, Canada

July 2020

CONCORDIA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared

By: Saeed Khazaee

Entitled: Detection of counterfeit coins based on 3D Height-Map Image
                Analysis

and submitted in partial fulfillment of the requirements for the degree of

            Doctor Of Philosophy (Computer Science and Software Engineering)

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

_____Chair
Dr. Arash Mohammadi

_____ External Examiner
Dr. Herbert Yang

_____ External to Program
Dr. Nizar Bouguila

_____ Examiner
Dr. Sudhir Mudur

_____ Examiner
Dr. Weiyi Shang

_____Thesis Supervisor
Dr. Ching Y. Suen

Approved by   _____
                         Dr. Leila Kosseim, Graduate Program Director

October 7, 2020

                         _____
                         Dr. Mourad Debbabi, Dean
                         Gina Cody School of Engineering and Computer Science

Abstract

Detection of counterfeit coins based on 3D Height-Map Image Analysis

Saeed Khazaee, Ph.D.
Concordia University, 2020

Analyzing 3-D height-map images leads to the discovery of a new set of features that cannot be extracted or even seen in 2-D images. To the best of our knowledge, there was no research in the literature analyzing height-map images to detect counterfeit coins or to classify coins. The main goal of this thesis is to propose a new comprehensive method for analyzing 3D height-map images to detect counterfeit of any type of coins regardless of their country of origin, language, shape, and quality. Therefore, we applied a precise 3-D scanner to produce coin height-map images, since detecting a counterfeit coin using 2D image processing is nearly impossible in some cases, especially when the coin is damaged, corroded or worn out. In this research, we propose some 3-D approaches to model and analyze several large datasets. In our first and second methods, we aimed to solve the degradation problem of shiny coin images due to the scanning process. To solve this problem, first, the characters of the coin images were straightened by a proposed straightening algorithm. The height-map image, then, was decomposed row-wise to a set of 1-D signals, which were analyzed separately and restored by two different proposed methods. These approaches produced remarkable results.

We also proposed a 3-D approach to detect and analyze the precipice borders from the coin surface and extract significant features that ignored the degradation problem. To extract the features, we also proposed Binned Borders in Spherical Coordinates (BBSC) to analyze different parts of precipice borders at different polar and azimuthal angles. We also took advantage of stack generalization to classify the coins and add a reject option to increase the reliability of the system. The results illustrate that the proposed method outperforms other counterfeit coin detectors.

Since there are traces of deep learning in most recent research related to image processing, it is worthwhile to benefit from deep learning approaches in our study. In another proposed method of this thesis, we applied deep learning algorithms in two steps to detect counterfeit coins. As Generative Adversarial Network is being used for generating fake images in image processing applications, we proposed a novel method based on this network to augment our fake coin class

and compensate for the lack of fake coins for training the classifier. We also decomposed the coin height-map image into three types of Steep, Moderate, and Gentle slopes. Therefore, the grayscale height-map image is turned to the proposed SMG height-map channel. Then, we proposed a hybrid CNN-based deep neural network to train and classify these new SMG images. The results illustrated that a deep neural network trained with the proposed SMG images outperforms the system trained by the grayscale images. In this research, the proposed methods were trained and tested with four types of Danish and two types of Chinese coins with encouraging results.

I would like to express my sincere gratitude to my supervisor, Prof. Ching Yee Suen, for the valuable advice and support he has given me in this chapter of my life. I have been extremely lucky to have a supervisor who cared so much about my work, and who responded to my questions and queries so promptly. I cannot forget Dr. Suen's support and kindness, not only for providing the funding which allowed me to undertake this research or giving me the opportunity to attend places related to coin project and meet so many interesting people but also for his constant care about my personal life.

I would also like to thank all the members of staff at CENPARMI at Concordia University who have been helping me in my study. Specifically, Nicola Nobile, CENPARMI's research manager, for his great technical supports and providing very intimate and effective collaborations.

I would also like to extend my gratitude to my examination committee for taking the time to read and evaluate this thesis. Their comments and feedbacks are valuable and highly appreciated.

I would like to thank Ultra Electronics - Forensic Technology and the people there, specifically Serge Leveque for providing support and equipment for the collection of data.

I am honored to thank the Natural Sciences and Engineering Research Council for supporting this research project.

I want to express my hearty gratitude to my love, Maryam, whose love and supports are with me in whole chapters of my life.

I am truly grateful to my parents for their immeasurable love and care. They have always encouraged me to explore my potential and pursue my dreams.

Declaration

The work described in this thesis has not been previously submitted for a degree in this or any other university, and unless otherwise referenced, it is the author's own work. No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institution of learning.

Saeed Khazaee

Concordia University

July 2020

Publications produced towards Ph.D. candidature:

The following papers have been published for publication or under review as a direct result of the research discussed in this thesis.

The papers that I contributed as the first author extracted from my thesis:

1. **S. Khazaee,** M. Sharifi Rad, C. Y. Suen, "Detecting of Counterfeit Coins Based on Modeling and Restoration of 3D Images", In: Barneva R., Brimkov V., Tavares J. (eds) Computational Modeling of Objects Presented in Images. Fundamentals, Methods, and Applications. Lecture Notes in Computer Science, Springer, Volume 10149, 2017, pp. 178-193.

2. **Khazaee, S.**, Sharifi M., Suen, C., "Restoring height-map images of shiny coins using spline approximation to detect counterfeit coins", In proceeding of ICPRAI 2018, Montreal, Canada, Center for Pattern Recognition and Machine Intelligence, 2018, p. 383-387.

3. **Saeed Khazaee**, Maryam Sharifi Rad, Ching. Y. Suen, "Detection of counterfeit coins based on 3D Height-Map Image Analysis ", Expert System with Applications, (Under review)

4. **Saeed Khazaee**, Maryam Sharifi Rad, Ching. Y. Suen, "A novel deep learning-based method to detect counterfeit coins"

The following papers have been published for publication or under review as an indirect result of the research discussed in this thesis.

Other articles related to counterfeit coin detection as the second author:

5. Sharifi M., **Khazaee, S.,** Suen, C., "Counterfeit Coin Detection Based on Image Content by Fuzzy Association Rules Mining", In proceeding of ICPRAI 2018, Montreal, Canada, Center for Pattern Recognition and Machine Intelligence, 2018, p. 285-289.

6. M. Sharifi Rad, **S. Khazaee**, L. Liu, and C.Y. Suen, "A Blob Detector Images-Based Method for Counterfeit Coin Detection by Fuzzy Association Rules Mining", Second International Conference on Pattern Recognition and Artificial Intelligence, ICPRAI 2020, Zhongshan City, China, In press.

7. M. Sharifi Rad, **S. Khazaee**, L. Liu, and C.Y. Suen, "A Pruned Fuzzy Associative Classifier to Detect Image-Based Counterfeit Coins", Pattern Analysis and Applications, (Under review)

The papers related to deep learning:

8. **S. Khazaee**, A. Tourani, S. Soroori, A. Shahbahrami, C. Y. Suen, "A real-time License-Plate Detection Method using a Deep Learning Approach", ICPRAI2020, Zhongshan, China. (Accepted)

9. Tourani, S. Soroori, **S. Khazaee**, Ching Y. Suen, "A robust Deep Learning Approach for Automatic Iranian Vehicle License Plate Detection and Recognition for Surveillance System", IEEE Transaction on Intelligent Transportation Systems, (Under review).

10. Joe Afshany, A. Tourani, A. Shahbahrami, **S. Khazaee**, and A. Akoushideh, "Parallel Implementation of a Video-based Vehicle Speed Measurement System for Municipal Roadways", International Journal of Intelligent Systems and Applications, vol. 11, no. 11, pp. 25-37, 2019.

11. Tourani, A. Shahbahrami, A. Akoushideh, **S. Khazaee**, C. Y Suen, " Motion-based Vehicle Speed Measurement for Intelligent Transportation Systems", International Journal of Image, Graphics and Signal Processing(IJIGSP), Vol.11, No.4, pp. 42-54, 2019.DOI: 10.5815/ijigsp.2019.04.04

Table of Contents

Introduction

This thesis proposes several new methods for counterfeit coin detection. Counterfeit coin detection is a classic research topic, and it has attracted much interest in fields such as numismatics and forensic investigation. In this research, instead of conventional 2-D methods for counterfeit coin detection, we applied several 3-D approaches to model and analyze six different coin datasets. In this chapter, the problems of counterfeit coin detection, motivation, dataset preparation, challenges, and their solutions, and related research problems will be discussed.

1-1 Problem Statement

A counterfeit coin is an imitation of a genuine coin with very high quality so that it can deceive many ordinary people and even experts. There are presently a lot of companies, museums, and government agencies in the world that have increased the demand for automatic systems to classify precious, historical, and common coins. Thanks to the increased necessity on intelligent approaches to recognize counterfeit coins, coin detection has continuously been evolving in recent years, and image-based coin recognition has become an integral part of this research area. An image-based coin recognition system takes digital images of coins as input and classifies them according to their denomination, country, and year of production. According to previous works, coin detection is mostly based on 2-Dimensional image processing techniques. Many researchers usually have extracted texture features from coin images and recognized them. In particular, edge information had been frequently adopted as features of coin images with Fourier transform. The difficulty level of coin detection heavily depends on the backgrounds of the coin datasets used in the experiments. Fortunately, all the datasets used for coin detection are captured with plain backgrounds, such as pure white or conveyor belt with extremely deficient gray levels [1]. Counterfeit coin detection can be more challenging than coin recognition because of the very high similarity between fake and genuine coins, in most cases, needing more sophisticated features to distinguish between them. Some of the standard features in coin recognition like diameter, thickness, weight, or shape may be used to detect counterfeit coins. We should note that these basic attributes are easy to copy, and the fake coins made nowadays are of high quality; therefore, these systems cannot distinguish counterfeit coins from genuine ones when their physical properties are

necessarily the same. Recently, several methods based on image processing techniques and classification algorithms have been proposed to improve the performance of counterfeit detection systems, and many lectures and tutorials have been devoted to them. Some of these methods are not very sophisticated and use coin colors and radius-based features to detect counterfeit coins. Unfortunately, these approaches are incapable of distinguishing coins having the same values of the limited set of features. They are also not able to overcome the problems related to rusty, dusty, or poor-quality coins.

In this research, we applied 3-D scanning to obtain a height-map image-dataset. In the height-map images, most of the quality problems of the coins are compensated. We take advantage of 3-D approaches to extract effective features that can ignore many of the coin's quality problems and prove the feasibility of 3-D approaches for counterfeit coin detection.

In our first and second papers of this research, the ring part of coin images were straightened by a proposed straightening algorithm. Then the height-map image was decomposed row-wise to a set of separate 1-D signals, which were analyzed separately and restored by two different proposed methods. After estimating degradation function and signal restoration, we performed feature extraction by various methods. The system trained by several state-of-the-art classifiers, and the results were compared.

We also proposed a new method using 3-D approaches to detect and analyze the precipice borders of the coin's surface and extract significant features to recognize counterfeit coins. Therefore, proposing a method that can extract more useful features from the precipice border of the coin images, can be remarkable. Our first contribution in this method was to propose an approach based on fuzzy C-Means clustering to detect precipice borders. Our next contribution was to propose Binned Borders in the Spherical Coordinate system. This system can be used to analyze the detected precipice borders from different directions in a polar and azimuthal angle. Besides, for extracting feature, a method using BBSC and sub-division structure is proposed.

In another proposed method of this thesis, we applied deep learning algorithms in different steps to detect counterfeit coin. As Generative Adversarial Network is being used for generative fake images in image processing applications, we proposed a method based on this network to augment our fake coin class and solved the lack of fake coins for training the classifier. We also decomposed the coin height-map image into three proposed channels of Steep, Moderate, and Gentle slopes. Therefore, the grayscale height-map image is turned to the proposed SMG height-map channel. Then, we used transfer learning to train a pre-trained neural network and classify these new SMG images.

2

1-2 MOTIVATION

Criminals have made counterfeit coins for thousands of years as an illegal industry. A lot of counterfeiting rings manufacture and sell fake coins, which have caused great loss and damage to society. As an example, in Europe, the data withdrawn from the Counterfeit Monitoring System (CMS) indicate an increase in the number of counterfeit coins seized from 2015 to 2017. Figure 1.1 shows that the total number of counterfeit euro coins removed from circulation since 2013 was equal to 837,910 pieces with a value of 1,330,401 Euros [2] while much more are still expected to remain in circulation. In non-circulating coin markets, ancient and valuable coins, counterfeiting is the most serious issue facing collectors today that could cost billions of dollars every year. In the past, rare coins were forged to deceive tourists, inexperienced people, or novices. These days, the technology of counterfeiting has grown exponentially that can even fool coin experts in some cases [3].
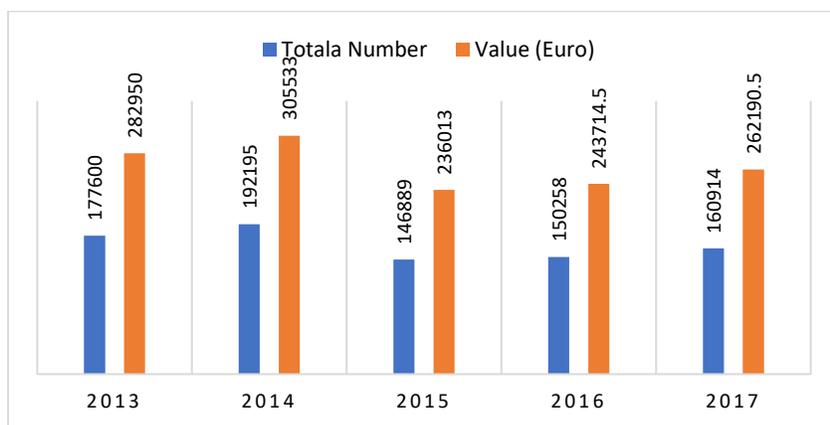
Figure 1.1. The total number and value of the counterfeit coins removed from circulation from 2013 to 2017.

Given the highly improving qualities of fake coins made nowadays, more thorough and closer inspections are necessary to detect well forged fake coins to provide substantial evidence for detection results. To get the idea to extract effective features from coins, we met several experts from several companies and coin collectors in Montreal, Canada, to find the features with which they used to concentrate, for counterfeit coin detection. Consequently, we found that all experts rotate the coins and focus on *precipice borders* of the characters and shapes on the coin's surface when the weights of the coins were mostly correct. Therefore, we decided to propose a method that can detect and analyze precipice borders on the coin surface as an expert, with the difference that we take advantage of machine intelligence, which has been a remarkable and very effective alternative for human experts in the most similar applications.

1-3 DATASET PREPARATION

1-3-1 Source of Coins and Labeling

The coins in this research were provided by the Law enforcement office and Danish police, including fake and genuine coins. In collaboration with *Ultra Electronics Forensic Technology Ltd. Co* and coin experts in Montreal, we labelled Danish coins available for this research. We also applied the majority votes for Chinese coin labelling by at least five coin-experts.

1-3-2 The 3-D scanner

Coin image were acquired by a very precise 3-D scanner in the name of *IBIS TRAX*. The scanner was patented by *Ultra Electronics Forensic Technology* company. Inside the machine, they have a built-in microscope. With a very high resolution of the height-map images in the order of 6 microns and lateral resolution in sub-micron, users can extract useful information from tiny topographical features. The machine uses five groups of adjustable LEDs which allow users to acquire the coin image from different angles. The machine can produce both 2-D and height-map images as a result of scanning [4].

(a)	(b)	(c)
(d)	(e)	(f)
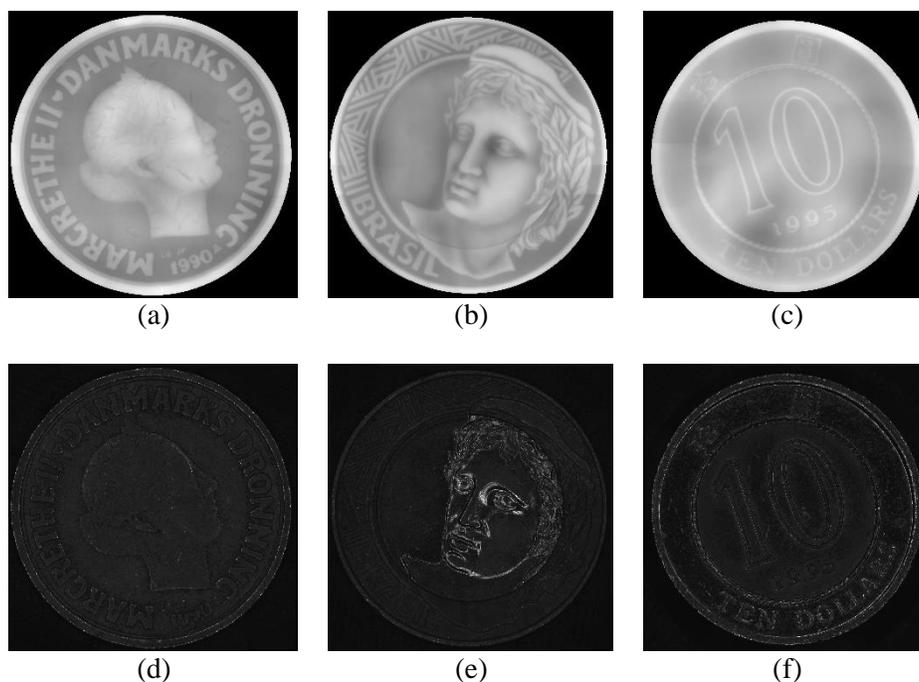
Figure 1.2. Height-map and 2-D images produced by IBIS TRAX machine.

<div align="center">(a)   (b)   (c)</div>

<div align="center">(d)   (e)   (f)</div>
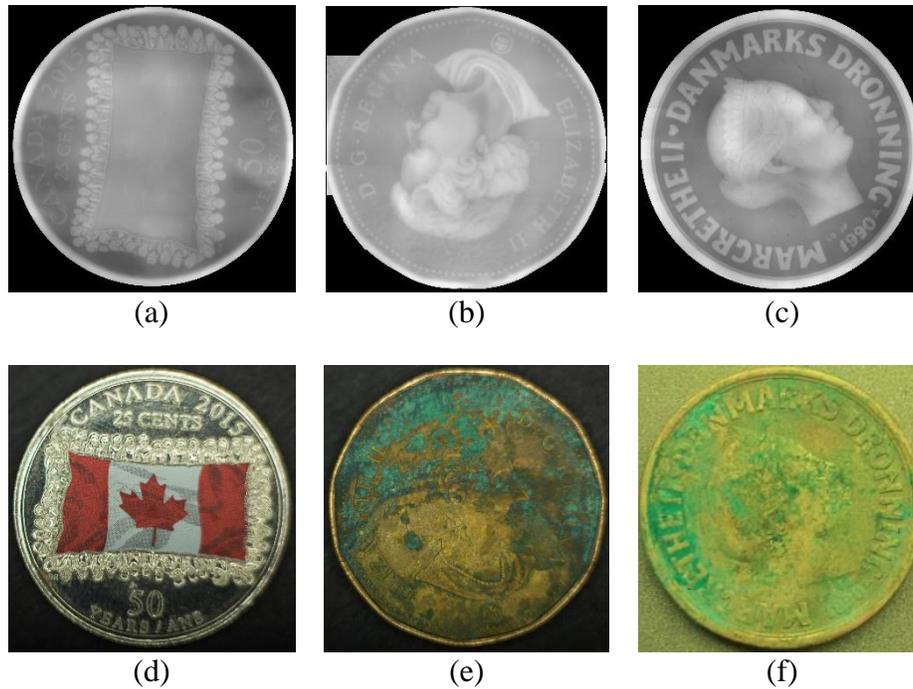
Figure 1.3. Comparing the results from the 3-D scanner and 2-D camera.

Figure 1.2 illustrates the scanning of three coins, for example. Figure 1.2. (a), (b), and (c) are the height-map images for Danish, Brazilian, and Hong Kong coins, and Figure 1.2. (d), (e), and (f) are their 2-D images resulted by *IBIS TRAX* machine respectively. Figure 1.3. (a), (b), and (c) demonstrate the height-map image for coins in Figure 1.3. (d), (e), and (f) respectively. It is clear from the images that the scanner can perfectly collect the height/depth information on the coin surface.

1-4 CHALLENGES AND THEIR SOLUTIONS

In this research, during the study on 3-D approaches to extract effective features, we have faced many challenges related to shiny coins and tried to find proper solutions for them.

1. Well-forged Coins

The preliminary goal of this research is to demonstrate the feasibility of the use of 3-D image processing to detect counterfeit coins which are very well-forged and similar to genuine counterparts. In some cases, even experts are not able to recognize them. As mentioned earlier, we employed a powerful 3-D scanner (depth resolution in the order of 6 microns, lateral resolution in sub-micro [4]) to scan six types of coins with different qualities. The first contribution of this research is exploiting the significant advantages of the 3-D approach in coin recognition, which had not been explored in the previous works. Considering the nature of the data captured by 3-D scanning, it was expected to extract effectively features related to height or depth instead of colors

related to the coin luminance. Besides, the most important benefit of 3-D scanning is its robustness against the quality of the coin surface in many cases. This capability contributed to purifying images captured from damaged coins.

2. Degradation problem due to poor scanning shiny coins

Despite the advantages of 3-D scanning [5], [6], there is a severe challenge when shiny coins are processed [6]. Although 3-D scanning is indeed independent of the lighting condition of the environment, we had a lot of unexpected degradations and shadowing on shiny coin images. Therefore, we faced unreal values of height or depth.

There are many different technologies used in 3-D scanners; each technology has its advantages, restrictions, and cost. For example, optical technologies face many difficulties when processing, reflecting, transparent, or shiny objects [6]. Figure 1.4 illustrates how a 3-D scanner captures height and depth: it is robust regarding the quality of the coin surface, while the 2-D image of this coin is entirely unreadable. In Figure 1.5, we can see the degradation of a shiny coin image captured by the 3-D scanner, which uses optical technology. As can be seen from this figure, despite the remarkable advantages of 3-D scanning for poor quality coins, there may be some abnormal and invalid results while scanning shiny and high-quality coins. However, the validity of the data captured by scanning is crucial for this research as the invalid values related to height and depth will adversely affect the pattern recognition process. To face the problems about scanning the shiny coins, we have two options:

a) Restoring the degradation problem; which we provided two different proposed methods for restoring the degraded image in chapters 3 and 4.

b) Ignoring the degradation problem, which in this research, we proposed a new approach to analyze the precipice borders in chapter 5 and a method based on deep neural networks, which are not affected by the degradation problem in chapter 6. Robustness against this problem is one of the advantages of these proposed methods.

(a)   (b)
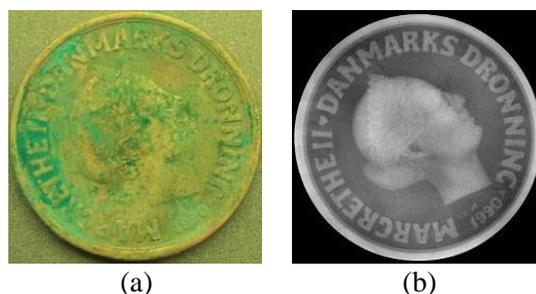
Figure 1.4. (a) Twenty Kroner 1990 coin captured by a normal camera, and (b) the same coin captured by a 3-D scanner.
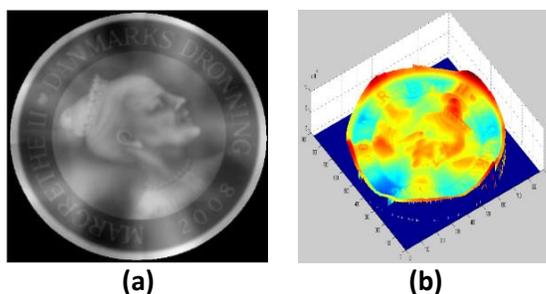
Figure 1.5. (a) Degraded image of a shiny coin, and (b) mesh plot of the same image.

3. Physically degraded or worn-out coins

Apart from the degradation problem in shiny coin's image, classifying the damaged coins is also a very challenging problem in this research. Here, we apply a dataset of damaged and suspected coins for training and testing the proposed model. There are two primary purposes of the use of the damaged coins for testing the model. Firstly, in many cases, counterfeit coin makers slightly damage the fake coins to conceal the critical details which are made roughly. Therefore, damaged counterfeit coins are more similar to damaged genuine coins. Secondly, to generalize the system to use for ancient and old coins that are precious even with low quality, we need to have the damaged coins dataset to evaluate the system. Figure 1.6 illustrates that recognizing a damaged coin can be very difficult or even impossible in many cases. According to Figure 1.6. (a), it is clear that a 2-D image of the damaged coin does not give us any useful information, and it cannot even be recognized as a Danish coin. Figure 1.6. (b) is the result of 3-D scanning having more details. However, it can be said with certainty that neither information comes from the height-map image enough to distinguish fake from genuine coins and mislead the classifier in the training process. To increase the reliability of the method, we designed a reject option for the counterfeit coin detection system. In this step, if a training sample is not good enough to lead the classifier, it is labelled rejected. The reject option will be described in Chapter 5.

(a)   (b)
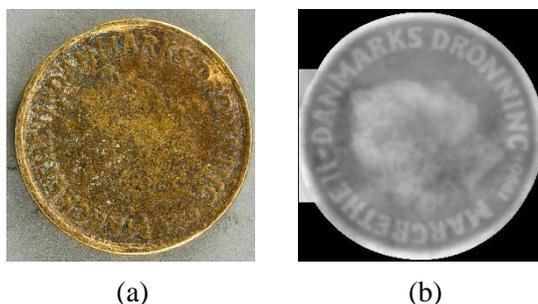
Figure 1.6. The difference between the results 2-D and 3-D scanning for a damaged coin.

4. Lack of fake coins

The limited number of counterfeit coins available in the literature was always a challenging problem. As mentioned earlier, the coins used in this research were provided by the Law Enforcement Office, and we used all of them. Yet, we should note that access to more fake coins is restricted and has been an enormous challenge for this research. To compensate for the adverse impact of this problem, in this study, we used a common augmentation technique and proposed a new method based-on Deep Convolutional Generative Adversarial Networks (DCGAN) to increase the size of the dataset for both the genuine and fake coins, which will be described in Chapter 6.

1-5 LITERATURE REVIEW

There have been a lot of studies in the literature for coin recognition and counterfeit coin detection, illustrating how to use coin features to classify the coin. In imaged-based coin recognition, for example, methods have been proposed to prepare and pre-process coin images indicating what features from the images are extracted and used in the classification step. In this section, an overview of coin recognition and counterfeit coin in literature is briefly summarized.

1-5-1 Image-based counterfeit coin detection

Recently, a few studies based on pattern recognition and image processing techniques have attracted interest as a means to detect counterfeit coins. These studies proposed different methods and were divided into three steps: finding the Region of Interest (ROI) of the coin image with coin segmentation, feature extraction, and training a classifier. Coin segmentation is the first step where the actual coin is located and segmented from the whole image [7]. Feature extraction is the most important step, which extracts effective features capable of discriminating the high-similarity objects. Several feature extraction methods have been used on coins in the literature using Gabor filter, Hough Transform, Heuristics, and Artificial Neural Networks. Eventually, in the classification step, different machine learning methods have been used to distinguish fake coins from genuine ones. In particular, edge detection has been widely used in the feature extraction process. In [8], an image-based approach to detect the fake coins has been proposed, which exploits the structure of the coin by using spatially local coding to extract effective features from the surface of the coin image. In several previous works, an edge map was extracted and segmented into several parts. After that, the authors proposed a method using histogram analysis and Fourier

transformation to handle and recognize rotated coin images. For example, in [9], the letters were extracted from the coin image, and a feature extraction method was applied to the letters and characters. The accuracy of fake coin detection was around 100%. Although the results mentioned in the paper were significant, the method was not applicable for any type of coin as it was limited to letters or shapes. However, the extracted features were not very useful for noisy and degraded images such as corroded or worn out coin images. In reference [10], the authors proposed a method to extract local texture features for coin recognition. They used Gabor wavelets and local binary pattern (LBP) to detect texture information. The coin was divided into several small sections by a concentric ring structure. Then using Gabor coefficients or LBP values, the feature vectors were produced. In the proposed method, a circular shift operator was used to provide the robustness of Gabor features against rotation variance. For matching between two-coin images, a distance measurement was used. Despite the adequate feature extraction time, the experiments in this paper illustrate that the method was not tested for poor quality coins and might be sensitive during matching coins with high similarity. In reference [11], the authors proposed a region binary patterns method which was rotation-invariant. They applied gradient magnitudes instead of histogram analysis and extracted rotation-and-flipping-robust features using local difference magnitude transform. Although the result is acceptable from the feature extraction point of view, the performance of the model is not satisfactory for fake coin detection. In reference [12], the authors proposed a multistage procedure that could recognize hundreds of coin types. Despite the novelties and results, this approach was not able to distinguish fake from genuine as it did not concentrate on the similarity of counterfeit and genuine coins. The authors in reference [1] proposed a new image-based approach to detect counterfeit coins. There, the local descriptors were employed to generate image representations. However, instead of resorting to the BOVW (Bag of Visual Words) model for a generation of vectorial representations, the authors represented a coin image in the dissimilarity space. They conducted one-class learning for fake coin detection. For this purpose, one-class SVM was employed. However, the method was susceptible to any variation and not suitable for poor quality coins. In [13], a transfer learning approach by fine-tuning a pre-trained network to analyze the features of characters on the coin surface has been used. The authors used an ensemble method to combine results of two classifiers trained by deep features from convolutional layers as well as a third classifier which was trained on distance-based feature of characters. The proposed method resulted in a precision rate of 85.1%. Albeit the method, in our knowledge, was the first use of deep learning to counterfeit coin detection, yet the results illustrated that the method did not have superiority rather than the recent methods. Recently, in [14], a

combination of a blob detector system and a method based on fuzzy association rules mining was proposed to detect counterfeit coins. The first step was about to preprocess the original image dataset by a blob detector to generate the frequent features. In the second stage, fuzzy association rules mining extracts the effective fuzzy rules and classifies automatically the coin image data. This method could handle better the sensitivity rather than the reference [1]. Since there was no option for rejecting damaged coins, in many cases, the system classified damaged coins, almost randomly.

1-5-2 Non-image-based methods

There have been several nonimage-based methods for counterfeit coin detection besides image-based methods. The authors in [15] proposed a method using an optical mouse to detect fake coins. They applied this method to the two-Euro coins and compared partial images of the coin under analysis with some partial reference coin images for matching in the detection process. The method was better in comparison with the untrained user and was almost the same as coin experts. Results illustrate that valid coins can be rejected if they are worn out or have physical damage, while poor copies and different types of coins are 100% rejected. The report in [15] also shows that the system cannot identify most of the well-forged coins as counterfeit. In [16] a method based on acoustic signal spectrum analysis for authentic and fake coins, has been proposed. In this reference, the authors optimized the parameters for time-spectrum co-analysis, and a very small dataset of Euro coins was used to validate the method. However, low costs and high accuracy were reported as the strength of the method.

1-5-3 Three-Dimensional image analysis

Albeit there have not been any studies on height-map image analysis for counterfeit coin detection before our research, it is worth mentioning some research in the literature review, which are conceptually close to our research methodology. Recently, interest in using 3-D approaches for image processing applications. Relevant studies illustrate that since 3-D approaches discover highly confident features, they can overcome some constraints introduced by other 2-D image classification methods. A plane detection algorithm in height-map 3-D images plays a vital role in 3-D object recognition. Much research is conducted to detect and analyze the planes for recognizing specific objects in a 3-D scene. For example, the authors in [17] proposed a depth image-based plane detection algorithm. As stated in this paper, the algorithm could be used as preprocessing for planar object recognition that has a fast run-time compared with other methods.

In [18], the authors proposed a method for depth image segmentation. The efficiency and accuracy of several methods for depth map segmentation are explored. They modified several state-of-the-art segmentation methods and proposed new segmentation approaches with the aim of optimizing performance characteristics. In [19], a method based on disparity transformation and 3-D road surface modelling was proposed for pothole detection. They used two stereo cameras for road data acquisition and finally extracted the point clouds of the detected potholes from the reconstructed 3-D road surface and got remarkable results rather than 2-D based images. In [20], a method of edge plane detection in three-dimensional images has been introduced by utilizing features of edge vector and edge reliability. This method can detect only reliable edge planes against noise influences. In [21], a method based on Hough transform for plane detection was proposed. The method could identify the clusters of coplanar points in the 2.5-D space using an implicit quadtree. They also applied a Hough-transform voting structure for the detection process to model the uncertainty associated with the best-fitting plane considering each cluster as a trivariate-Gaussian distribution. As concluded in this paper, the method was a real-time technique for plane detection. Figures 1.7. (a) and (b) indicate the depth-map image as input and the result of plane detection. Although the method can perfectly detect non-planar objects, we cannot extract any features from the objects, but contour-based features which are not very effective in counterfeit coin detection. Therefore, in a special task like recognizing a coin that has a very complex geometry, plane detection would not be very useful for feature extraction.
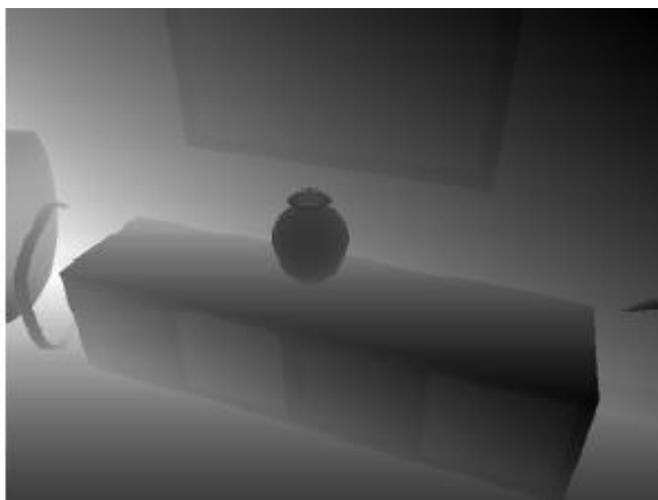
(a)
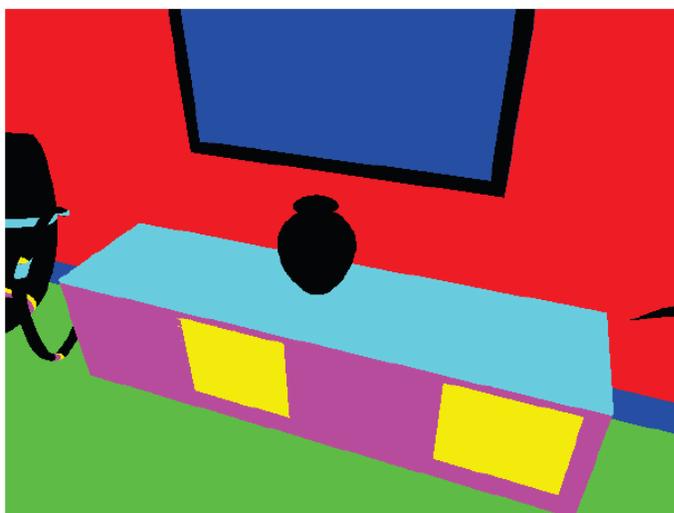
(b)

Figure 1.7. Result of plane detection produced by the method proposed in [18]: (a) a depth image as input, and (b) detected planes, visualized by colors (black color illustrates non-planar objects).

To get better use of depth images, we propose an ensemble method based on precipice border analysis in height-map images to detect counterfeit coins. In this method, the height-map image of the coins will be triangulated, and a clustering process based on Fuzzy C-Means algorithm will cluster the triangles to detect precipice borders of the coin's surface. Then, we propose a method to analyze the precipice borders and extract the valuable features for training a stacking classifier. The method will precisely investigate and analyze the precipice borders of the coin's surface to extract features and build a model sensitive to any little dissimilarity with genuine coins, which is indispensable for modern counterfeit coin detection. In the edge plane detection, we find a set of planes that are very useful to recognize and distinguish different objects in a scene. In contrast, in the precipice border detection, we find curved surfaces and analyze complicated details on non-

planar objects. By analyzing the precipice borders, we extract features capable of discriminating the high-similarity objects.

1-6 OUTLINE

The remainder of this thesis is organized as follows:

Chapter 2 defines the fundamental and preliminary concepts. In Chapters 3 and 4, we present two papers entitled "Detecting of Counterfeit Coins Based on Modeling and Restoration of 3-D Images" and "Restoring height-map images of shiny coins using spline approximation to detect counterfeit coins". These papers concentrated on restoring degraded images caused by poor-scanning of shiny coins. In chapter 5, the paper entitled "Detection of counterfeit coins based on 3-D Height-Map Image Analysis" will be introduced. In Chapter 6, a proposed method using deep learning approaches will be submitted. In this chapter, a new channel for height-map images will be proposed before using any deep learning method for counterfeit coin detection, and interesting results will be discussed. Finally, this study is concluded with a summary of the primary contributions and an outline of future work in Chapter 7.

Preliminary Concepts

In this chapter, the preliminary concepts and the overall objective to investigate the counterfeit coin detection will be briefly introduced.

2-1 FUZZY CLUSTERING

In classic clustering, each input sample belongs to one and only one cluster, and it cannot be a member of two or more clusters. When a sample is close to two or more clusters, the classic clustering for determining that the sample to which clusters belongs will have trouble. In fuzzy clustering, an instance may belong to more than one cluster and, in this situation, can solve the problem of sharp boundaries. The membership degrees in the range [0,1] are used in fuzzy clustering instead of assigning a sample to only one specific cluster. The clustering algorithms are divided into hard clustering techniques and soft clustering techniques [22]. In this paper, we use one of the most popular soft clustering techniques, fuzzy c-means (FCM) clustering, in our proposed precipice border detection of the coin's image. FCM is based on fuzzy approach and minimization of the following objective function [22], [23]:

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^m \left\| x_i - c_j \right\|^2 \qquad , 1 \leq m < \infty \qquad (2.1)$$

where m is a real number greater than 1, u_{ij} is the membership degree of x_i in the cluster j, x_i is the ith d-dimensional measured data, c_j is the d-dimension center of the cluster, and $\|*\|$ is the norm indicating the similarity between any measured data and the center of the cluster [22], [23].

2-2 PRECIPICE BORDERS DETECTION

In digital image processing, one of the most effective methods for extracting features from the important parts of objects is edge detection. Edge detection leads to a set of connected curves that indicate the contours of objects, the borders of the surface markings and curves that correspond to variations in surface orientation. Hence, an edge detection algorithm can reduce the volume of data to remove useless information by retaining the significant characteristics of an image.

In three-dimensional image processing also, edge detection can be used to identify the points placed on 3-D boundaries, which provides more effective and discriminative features. Instead of the standard edge detection in 2-D or 3-D approaches, we detect *precipice borders* for the height-map images. In edge detection, typically, we have a line separating two segments or regions while in three-dimensional *precipice border* detection, we have a plane separating two segments leading to a set of attributes. It helps us to extract useful features based on the normal vectors and the areas of the planes around the letters, numbers and other objects on the coin surface. Figure 2.1 shows the objects on the part of the coin image surrounded by precipice borders, which are highlighted in red color. Details about the proposed method for precipice border detection will be described in chapter 5.
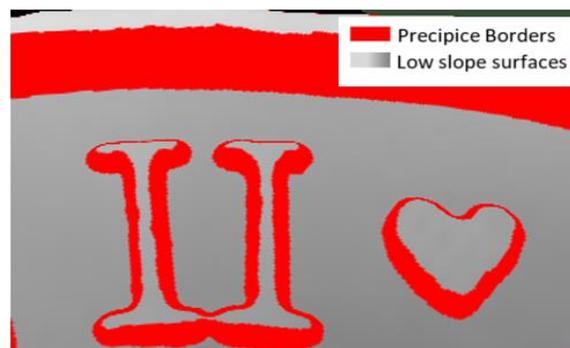
Figure 2.1. Objects are surrounded by precipice borders on a 20 Kroner 1990 coin.

2-3 STACKED GENERALIZATION

Stacking (Stacked generalization) is an ensemble learning technique that combines multiple classifications or regression models via a meta-classifier. Stacked generalization in classification tasks trains a meta-classifier by the predictions of the first-level classifiers. First, all the classifiers are trained by the train set. Then the meta-classifier is trained to predict the class using all the results of the first-level classifiers as its input. Generally, stacking performs better than any

individual trained models. It has also been successfully used on both supervised and unsupervised learning. Figure 2.2 shows a pseudo-code for stacking from [24].

Input: Training data $D_k = \{x_i, c_i\}_{i=1}^m$ ($x_i \in \mathbb{R}^n, c_i \in C$)
Output: An ensemble classifier H
   1: Step 1: Learn first-level classifiers
   2: **for $k \leftarrow 1$ to T**
   3:    Learn a base classifier h_t using D
   4: **end**
   5: Step 2: Construct new datasets from D
   6: **for $i \leftarrow 1$ to m**
   7:    Construct a new dataset that contains $\{x_i', c_i\}$, where $x_i' = \{h_1(x_i), h_2(x_i), \ldots, h_T(x_i)\}$
   8: **end**
   9: Step 3: Learn a second-level classifier
 10: Learn a new classifier h' based on the newly constructed dataset
 11: **return $H(x) = h'(h_1(x), h_2(x), \ldots, h_T(x))$**

Figure 2.2. Stacked generalization in the classification task algorithm [24].

Regarding reference [25], for successfully stacked generalization, it is necessary to use output class probabilities rather than class predictions for the first-level classifiers. To this end, we may use several classifiers that are naturally probabilistic or can be turned into probabilistic. In probabilistic classifiers, conditional distributions are considered as an alternative to functions. $P(X|Y)$, meaning that for a given $x \in X$, they assign probabilities to all $y \in Y$. The final decision for classifying x is done by this formula:

$$\hat{y} = \arg max_y P(Y = y|X). \tag{2.2}$$

In General, the selected class is the class with the highest probability. As mentioned earlier, some classification models, such as naive Bayes, logistic regression, and Multilayer perceptron are naturally probabilistic. However, there are several models, such as support vector machines that are not probabilistic initially. Therefore, we can use the methods to turn them into probabilistic classifiers [26].

2-4 DEEP NEURAL NETWORK

A deep neural network is a neural network with a high level of complexity and more than two layers. The term *deep learning* is also used to define these deep neural networks. Deep learning is

a subset of machine learning based on artificial neural networks, in which the learning process needs a large amount of data. Deep learning or *hierarchical learning* can be supervised, semi-supervised, or unsupervised. It uses multiple layers to extract higher-level features from the input data iteratively.

2-4-1 CONVOLUTIONAL NEURAL NETWORK (CNN)

Convolutional Neural Network (CNN), also known as a ConvNet is one of the most common deep neural networks in the field of image processing. A Multilayer perceptron usually has fully connected networks in all layers, also called dense layers, where each neuron of a layer is connected to all other neurons in the next layer. This characteristic of the dense networks leads them to overfit in many cases. Without concentrating on any specific segments of images, dense layers learn global patterns in input images. In CNN, the network is not fully connected in all layers. With *convolution* operation, convolution layers learn the local pattern. As can be seen in Figure 2.3, convolution operations are used to find patterns in small 2D windows of the input image. Another essential action in CNNs is *pooling,* also known as *subsampling* or *down-sampling*. The main reason to use pooling is to reduce the dimensionality and complexity. Pooling layers are usually placed after convolution layers. By reducing the dimensionality as well as the complexity of the input and parameters, the overfitting phenomenon would be minimized.

It should be noted that *max-pooling* outperforms alternative solutions like using strides in the prior convolution layer or average pooling. As is clear in the figure, the fully connected layer is typically put after the convolutional and pooling layers meaning that the output of the last layers is the input to the fully connected layers. To assign each node to a specific class (the type of virus in this example), an activation function naming *softmax* is used to produce the membership degree of the input to each class, where the sum of all probabilities is 1.
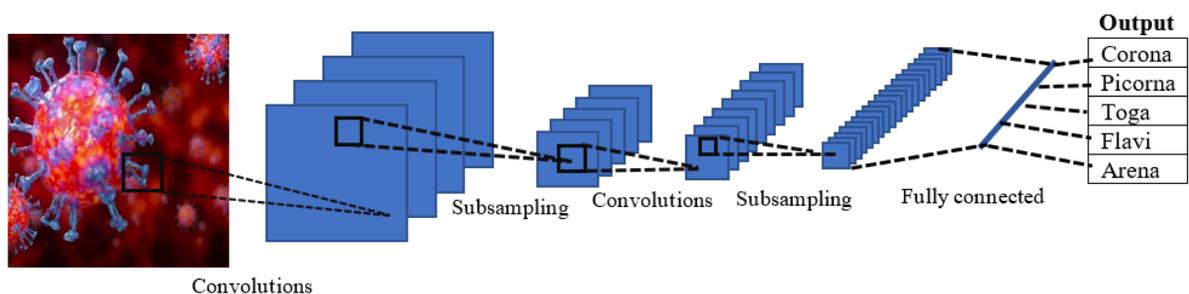
Figure 2.3. Structure of Convolutional Neural Networks.

A common and highly effective approach to deep learning when image datasets are very small is to use a *pre-trained* network. A pre-trained CNN is a saved network that was trained previously by a large dataset [27]. CNN models are getting deeper and improving by new structures that are being proposed frequently. Several structures of state-of-the-art pre-trained CNNs like LetNet-50 [28], AlexNet [29], VGGNet [30], ResNet [31], MobileNet [32], and Google's Inception v3 [33] have been proposed in the literature.

A pre-trained network can be used in feature extraction and fine-tuning in two different approaches. Here, feature extraction is to take the convolutional base layers of a previously trained network, and run the new data through it, and train a new classifier on top of the output layer. Fine-tuning changes and optimizes the filter weights of newly added blocks of the network iteratively. It freezes the first two convolution base layers. While those frozen layers already extracted generic types of features like edges, fine-tuning will update all intermediate layers with new data [27].

2-4-2 Deep Convolutional Generative Adversarial Networks (DCGAN)

Generative Adversarial Networks (GANs) belong to one of the most popular deep neural networks which are used to generate fake images from a set of original images [34]. Here, two models are trained at the same time by the adversarial learning process. A generator model learns to create fake images with high similarity to its counterpart, while a discriminator model learns to distinguish real images apart from fakes. Here, both generator and discriminator models are fine-tuned Convolutional Neural Networks. The generator tries to fool the discriminator network during the training process. By generating realistic images, discriminator learns to find dissimilarity. The result of this competition is to generate artificial images that are not easily distinguishable from genuine ones (Figure 2.4).

One of the most critical challenges in this study is the lack of fake coins. Since counterfeiting a coin for the research is impossible and the fake samples provided by the Law Enforcement Office is not enough for both training and test processes, we consider GANs to produce fake samples for training our counterfeit detector.
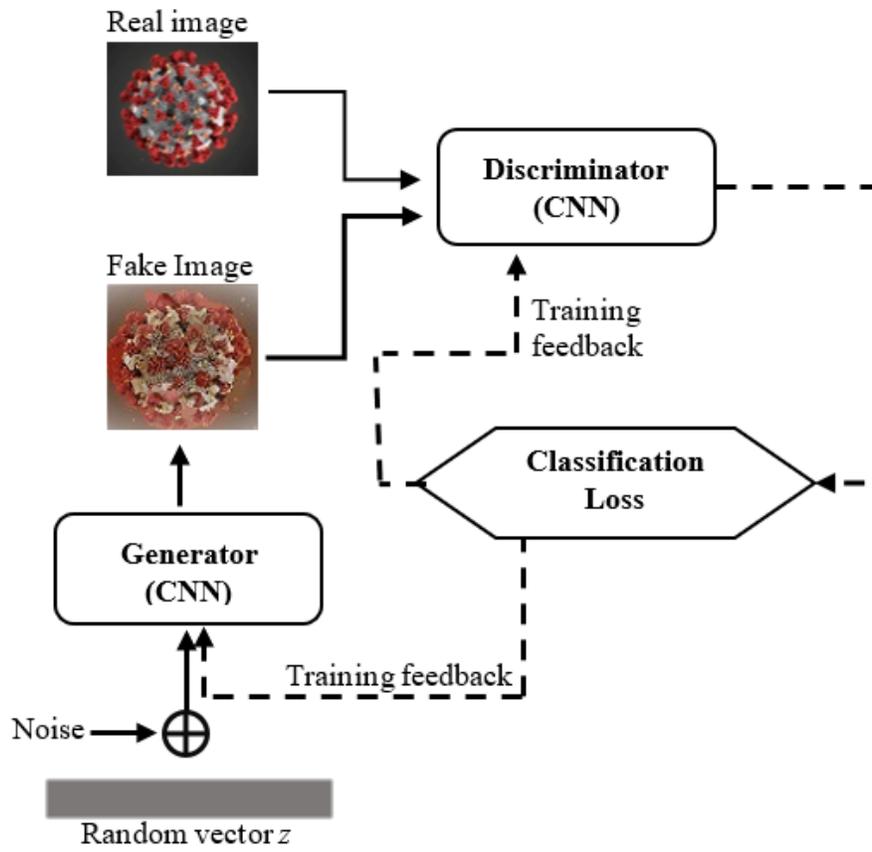
Figure 2.4. Generative Adversarial Network structure

Article 1

DETECTION OF COUNTERFEIT COINS BASED ON MODELING AND RESTORATION OF 3-D IMAGES

Saeed Khazaee, Maryam Sharifi Rad, Ching Y.Suen

CENPARMI, Concordia University, Montreal, Canada
{s_khaza, sey_shar, suen}@encs.concordia.ca

Published in:

Abstract. In image-based coin detection, making the image readable is an indispensable part of the feature extraction. However, using a 2-D image processing approach for detecting a counterfeit coin is nearly impossible in case of destroyed coins whose textures are severely burnt, sulfated, rusted, or colored. In this research, we used a 3-D scanner to scan and model an acceptable number of coins capturing height and depth instead of levels of color. The most important advantage of 3-D scanning is to compensate for the destruction mentioned above of the coin surface. Despite this advantage, we had several unexpected degradations due to shiny coin images. To solve this problem, the 3-D image was decomposed row-wise to a number of separate 1-D signals, which were analyzed separately and restored by the proposed method. This approach gave remarkable results when used to extract valuable features.

Keywords. counterfeit detection, coin recognition, 3-D-images, restoration

3-1 INTRODUCTION

A counterfeit coin is an imitation of a genuine coin with very high quality so that it can deceive many ordinary people and even experts. Nowadays, a lot of companies, museums, and government

agencies have increased the demand for automatic systems to classify precious, historical, and common coins. Thanks to the increased demand for automated approaches for detecting counterfeit coins, coin recognition has been continuously improving in classification performance, and image-based coin recognition has become a crucial part of coin detection.

Many automatic counterfeit coin detectors employ a primitive technology based on low-cost sensors to measure the weight, thickness, radius, conductivity, magnetic, or acoustic features of the coins [15]. However, these systems cannot distinguish fake coins from genuine ones when their physical properties are necessarily the same. In order to increase the potential of fake coin detectors and image-based coin detection, several methods based on image processing techniques and classification algorithms were proposed [10], [15], [35], and many lectures and tutorials were devoted to them [10], [36], [37]. Some of these methods are not very sophisticated and use coin color and radius-based features to detect counterfeit coins. Unfortunately, these approaches are incapable of distinguishing coins having the same values of the limited set of features.

In recent years, image-based counterfeit detection has expanded, and many researchers have applied image processing techniques to extract effective features from the texture of the coin images [9], [12]. In particular, edge detection information has been widely used in the feature extraction process. In references [12], [38], an edge map was extracted and segmented into several parts. After that, the authors proposed a method using histogram analysis and Fourier transformation to handle and recognize rotated coin images. However, the extracted features were not very useful for noisy and degraded images such as rust, dust, or sulfated coin images. In [9], the authors extracted the letters on the coin surface and tried to recognize them. In spite of the mentioned novelties of the proposed method, it is clear that the method was not robust enough to distinguish the counterfeit from genuine coins whose images have weak or smooth edges. In reference [11] a new method using rotation-invariant region binary patterns based on gradient magnitudes was proposed. To increase the accuracy of coin recognition, it computes gradient magnitudes in a coin image and extracts rotation-and-flipping-robust features using local difference magnitude transform. Although the result is acceptable from the feature extraction point of view, the time-performance of the model is not satisfactory.

Paper [15] proposes to use the sensor of an optical mouse as a counterfeit coin detector based on the image acquisition capabilities of the optical mouse sensor in short distances. The results are compared with the ones of trained and untrained users.

In this paper, a new method is proposed based on analyzing and restoration of 3-D images to detect counterfeit coins. We employed a 3-D scanner to scan a large number of coins with different

qualities. The first contribution of this research is exploiting the significant advantages of the 3-D approach in coin recognition, which had not been discussed in the previous works. Considering the nature of the data captured by 3-D scanning, it was expected to extract effectively features related to height or depth instead of colors related to coin luminance. In addition, the most important benefit of 3-D scanning is the robustness of the quality of the coin surface. This capability contributed to purifying images captured from sulfated, rusted, or colored coins.

Despite the advantages of 3-D scanning, there is a severe challenge when shiny coins are processed. Although 3-D scanning is indeed independent of the lighting condition of the environment, we had a lot of unexpected degradations and shadowing on shiny coin images. Therefore, we faced unreal values of height or depth. The next contribution of the proposed method is to eliminate this degradation. In order to restore an image, it was decomposed to many separated signals. To extract significant features from the signals, the part of the coin with the letters was converted into a long rectangle. Then, the signal of each row was analyzed and restored by the proposed method separately. The signal restoration process was notably improved, and we could use each newly restored signal to extract new effective features. The experimental results on various classifiers showed that the proposed method has an outstanding performance in terms of true positive (detection rate), false positive, precision, recall, f-measure, and feature extraction time.

The rest of this paper is organized as follows: Section 3-2 explains the 3-D scanning and its advantages and challenges briefly. Section 3-3 describes the proposed method to detect counterfeit coins and contains three essential subsections: straightening the coin, signal restoration, and feature extraction. In Section 3-4, we present the experimental results and analyze them. Finally, we conclude in Section 3-5 and comment on some future works.

3-2 DEGRADATION PROBLEM IN THE 3-D IMAGES

There are many different technologies used in 3-D scanners; each technology has its advantages, restrictions, and cost. For example, optical technologies face many difficulties when processing, reflecting, transparent, or shiny objects. Figure 3.1 illustrates how a 3-D scanner captures height and depth: it is robust regarding the quality of the coin surface, while the 2-D image of this coin is completely unreadable. In Figure 3.2, we can see the degradation of a shiny coin image captured by the 3-D scanner, which uses optical technology. As can be seen from this figure, despite the remarkable advantages of 3-D scanning for poor quality coins, there may be some abnormal and

invalid results while scanning shiny and high-quality coins. However, the validity of the data captured by scanning is crucial for this research as an invalid value related to height and depth will adversely affect the pattern recognition process. Therefore, our proposed restoration module should enhance not only the quality of the images as a whole but also each small part of the image must be restored as precisely as possible.

(a)   (b)
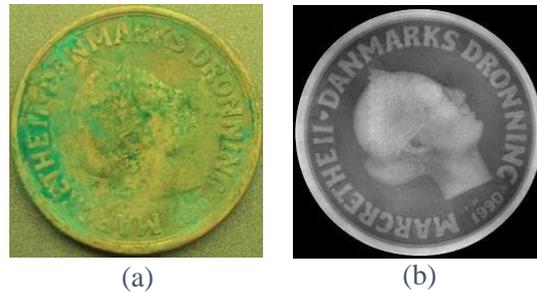
Figure 3.1. (a) Twenty Kroners 1990 coin captured by a normal camera, and (b) the same coin captured by a 3-D scanner.

(a)   (b)
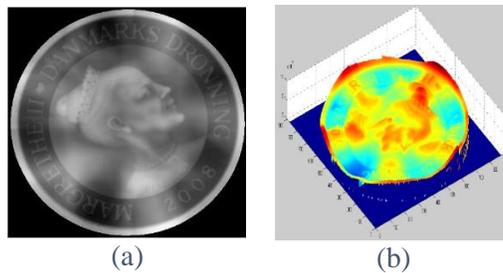
Figure 3.2. (a) Degraded image of shiny coin and (b) mesh plot of the same image.

3-3 PROPOSED COUNTERFEIT DETECTION METHOD

In this section, we present a new method based on signal restoration using 3-D images robust enough against rust, dust, sulfate, or any other factors, which may affect the coins' quality. In addition, the proposed method eliminates the problem of the degradation related to the shiny coin images, which occurs during the 3-D scanning.

3-3-1 Straightening the Coin Image

In this section, we present the first innovation of our proposed method, which is very important for performing the next steps. Having in mind that our final goal is to extract features from the

height and depth information, we assume that the part of the coin where the letters are placed in is more useful. Because the remaining part of the coin such as head profile or shapes placed in the center of the coin, most likely would have greater height and will be scratched earlier than the letters. Therefore, the straightening coin image is proposed to reduce the complexity of processing. This algorithm can also be beneficial for the segmentation of the letters in 2-D approaches. For implementing this step, we proposed an algorithm mapping the ring part of the coin into a rectangle using the equation of a circle in polar coordinates. Algorithm 3.1 demonstrates the straightening process of the coin where $C2 - C1$ is the radius of the part we want to process. It means that the matrix of the linear image is created column by column. Each column is composed of the values of the pixels from the circumference toward the center as much as $C2 - C1$ pixels. Below, lr is the growth of r for reading the next pixel's values of the current column diagonally, and lt adjusts the increase of the angle for reading the next column of the linear image from the original image circularly. These parameters should be adjusted for various images in different sizes on the circular images with center coordinates (a, b). Figure 3.3 (b) shows the result of the algorithm for Figure 3.3 (a). As can be seen from this figure, all letters are placed next to each other and can easily be processed more simply than in a circular coin. This shape is convenient before the segmentation of the 2-D image.

In the next section, the key role of the straightening process for the proposed method will be presented. Figure 3.4 presents the steps of the proposed method schematically to create a record-dataset, which will be used for training and testing a counterfeit coin detector.

Algorithm 3.1. Pseudocode for the main part of the proposed straightening algorithm

```
Straightening algorithm:

Input: cropped circular coin image
Output: straightened coin image (linear image)
C₁ ←(the number of column)/2
C₂ ←(the number pixels which should be considered as the
ring part of the coin's image)
For each t between π and −π step lt=-0.001
    k←(k+1)
    i←0
    For each r in the ring part of coin image from C₁ to
    C₂ step lr=0.01
        i←(i+1)
        x←round(r.*sin(t)+a)
        y←round(r.*cos(t)+b)
        if x>0 and y>0
            Linear_image(i,k)←Circular_image(x,y)
```


Figure 3.3. Straightening algorithm on a Danish 1990 coin(a) original image, and (b) rectangular image after executing the straightening algorithm.

3-3-2 Image and Signal Restoration

In digital image processing, restoration means recovering an image that has been degraded by using a priori knowledge of the degradation phenomenon. Therefore, restoration techniques are oriented toward modelling the degradation function in order to infer the inverse process and recover the original image [39].

After the straightening process, we can see an adverse effect of poor scanning on shiny coin images. For some kinds of coins, for example, Danish 20 kroner 2008, all the images are totally degraded. In Figure 3.5, the differences between two examples of the 1-D signal of a specific row

related to a normal and a degraded image have been shown. In both images, a long rectangular image containing the letters can be seen. The signals below these images have numerous peaks showing the heights of the letters or numbers in the lines marked. Clearly, the figure shows that shininess has adversely affected the quality of the image and signal.

Our initial experiments showed that the degradation problem is not random noise. We scanned three coins, four times each one of them, and we obtained the same poor results. Since our proposed counterfeit coin detector is based on height and depth information, signal restoration from this digital data is an inevitable process. For the given digital data set obtained from scanning, we needed to recover the original signal, which includes more precise data, since the degraded signals related to fake and genuine coins are very similar. Referring to Figure 3.5, the signal demonstrates that the important information of letters and numbers corresponds to a low-frequency signal. Therefore, we have to estimate the low-frequency signal and subtract it from the original signal. Thus, we have $x = A + D$; where x is the original and degraded signal, A is an approximation of x, and D contains the details of x. In other words, the useful information comes from the details of the signal. In our first attempt, we used wavelet transform to find A or D. For restoring all signals and the image, and several well-known methods were performed. In the first implementation, we used various kinds of wavelet transform on the degraded image and obtained interesting results by subtracting the approximation A from the original signal x. However, the obtained signals were not precise enough to allow us to distinguish fake from genuine coins through height and depth information.
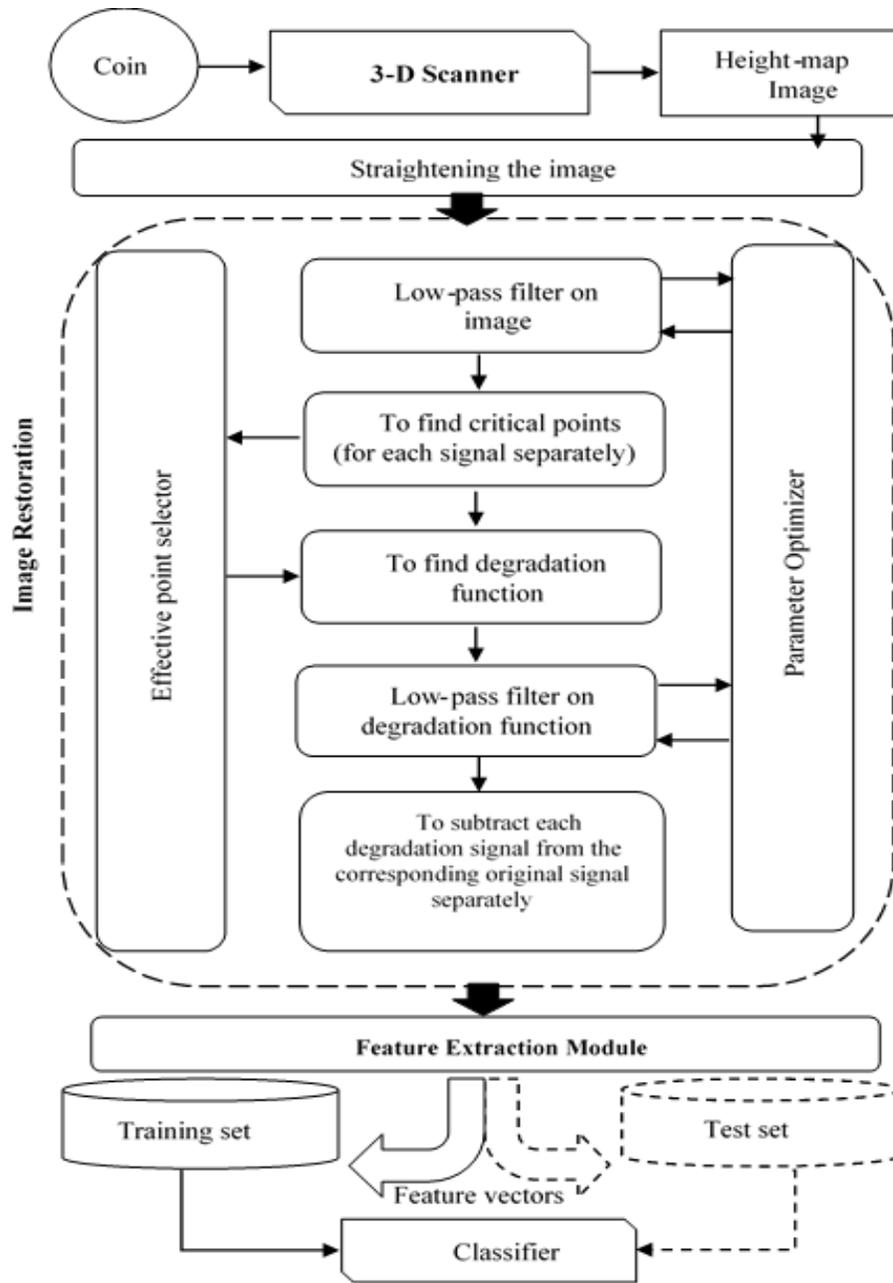
Figure 3.4. Structure of the proposed method.

Our second attempt for estimating A was to apply a Gaussian low-pass filter using Fourier transform. To find A, we can consider $A_T = X * H$, where A_T, X, and H are Fourier transforms of A, x, and h, respectively. Although this method was better than the wavelet transforms in restoring the image and produced a better signal, it could not approximate A as well as we needed.

27
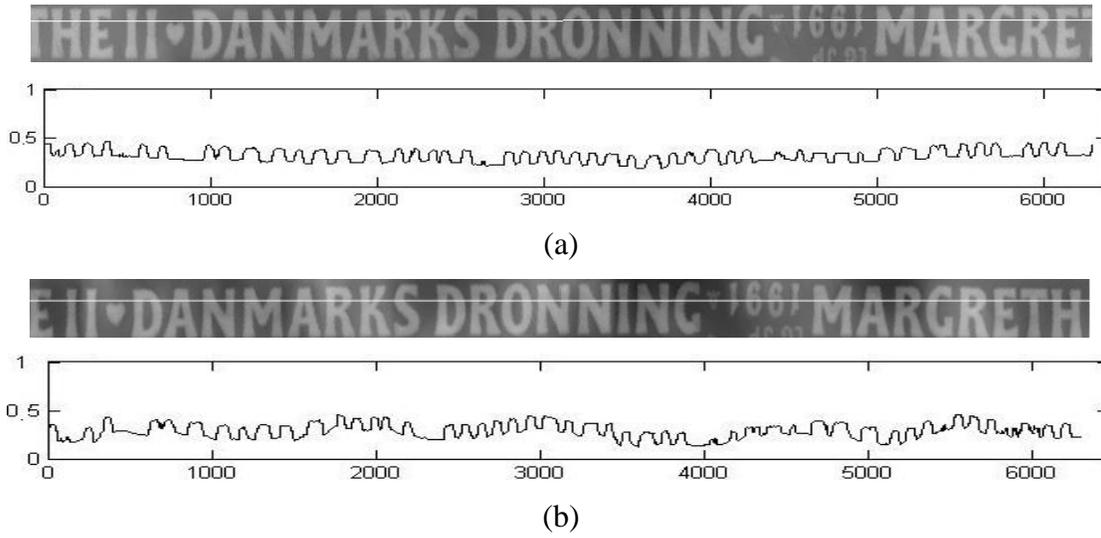
Figure 3.5. The difference between degraded and normal signals and images: (a) Normal image and its corresponding signal, and (b) degraded image and its corresponding signal.

After applying the Gaussian low-pass filter and wavelet transform, to find an approximation of the degradation function and to subtract it from the degraded signal, we could find that the restored image was still degraded. Although we had obtained an acceptable image after using the low-pass filters, there had not been the best-estimated values of height or depth. As mentioned before, the foundation of the proposed method is based on the height of the letters or numbers. Therefore, we need to perform a restoration method, which can be applied to each signal (row) of the image separately because the degradation function is completely abnormal and has to be considered as a 1-D signal restoration. For this purpose, we propose using a restoration module, which contains several steps, as described below.

Estimating Degradation Function and Signal Restoration. For estimating more accurately the degradation function for each 1-D signal, we proposed an estimator, which uses selected points determined by the discrete derivative. Since the signals, on which the process should be performed, are not smooth enough to perform mathematic operations, we used a low-pass filter to make them smoother. For estimating the approximation of degradation function related to the original signal g, we use the formula:

$$f(x) = \frac{g_i(x_i) - g_{i+1}(x_{i+1})}{x_i - x_{i+1}}(x - x_i) + g_i(x_i) \tag{3.1}$$

where $g_i(x_i)$ is the closest critical selected point to x and $x_i < x < x_{i+1}$. The critical selected point is routinely an extremum point selected by the effective point selector function. This function

28

uses momentum to avoid local minima that the signal has a lot of them. In addition, if the distance between two selected points is greater than a specific threshold, the point selector will consider several points on this part of the signal instead of the critical points. Figure 3.6 illustrates a small part of the original signal and the approximation of the degradation signal. As shown in the figure, the approximated signal is too sharp and needed to be smoothed. Consequently, we used a low-pass filter again to obtain the final signal, and we have $A = inverse(F * H)$ where F and H are Fourier transforms of f, and a 1-D Gaussian filter in the order. Also, the $inverse$ function returns the inverse discrete Fourier transform of the resulted vector. In addition, there is a simple parameter optimizer which regulates the value of σ in Gaussian filter, intending to minimize the Euclidean distance between A and f. Figure 3.7 demonstrates that the original image has been restored successfully. The signal related to the marked row of the restored image shows that the heights of the letters are more precise than the original rectangular image. Figure 3.7 also shows that the proposed restoration produces more balanced height information for the surface of the coin.
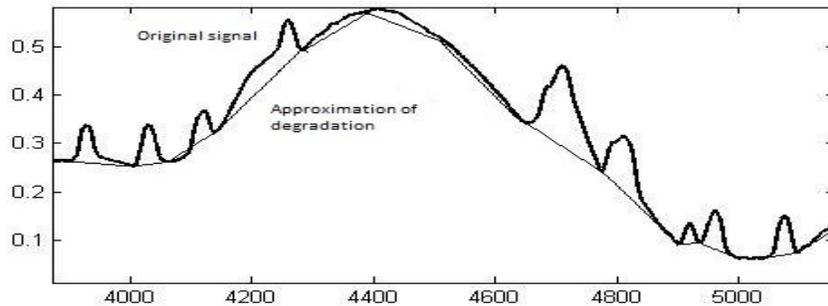
Figure 3.6. Estimating the approximation of degradation function.
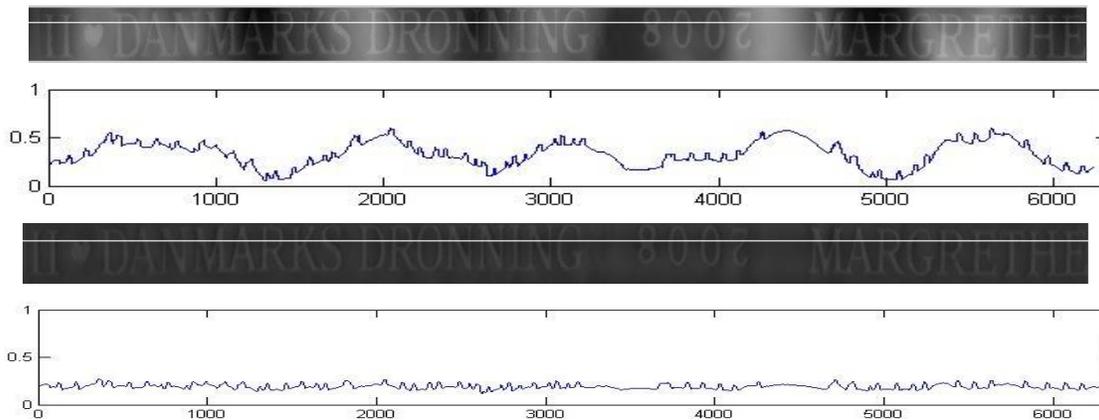
Figure 3.7. Result of the proposed restoration method on the images and their signals: (a) degraded image and its corresponding signal, and (b) restored image and its corresponding signal.

29

3-3-3 Feature Extraction and Creation of a Record Dataset

After performing the straightening process and restoring the images, we noticed that the signals corresponding to the genuine coin (in the same rows) are identical, while there is a significant difference between the genuine and the counterfeit coins. In Figure 3.8, the 1-D signals of the same row of six genuine and fake coins are shown for comparison. Since we can easily distinguish the counterfeit coin from the genuine with a height-map signal, we concentrated on the simple features of the signals. For each rectangular image, we have 311 rows, whose signals cover all the letters and the numbers. If we consider each signal as a feature, all these features would form a substantial set of features to distinguish fake from genuine coins, whereupon we would need to consider one or more features for each signal instead of the whole signal. In this paper, we use two simple features for each signal. The first feature is the energy (L2-Norm) of each decomposed signal, which can easily be calculated using the formula [40]:

$$E_i = <x_i(n).x_i(n)> = \sqrt{\sum_{n=1}^{c}|x_i(n)|^2} \qquad (3.2)$$

where x_i is the signal of row i and c is the number of columns in the matrix related to the rectangular image of the coin. An additional feature for each signal is the percentage of the energy of wavelet details in level two, whose calculation method was mentioned before; we denote it with PE_i. In addition, each coin has two general features: *year and coin number* and a target label *class* (*Fake* or *Genuine*). Accordingly, each rectangular image is converted to a record-data, and thus we have a total of 624 features from 311 signals, and a coin ID is assigned for the training process. After performing the feature extraction method on all existing images, a record-dataset is created.

3-4 EXPERIMENTAL RESULTS ON COUNTERFEIT DETECTION

In this section, the efficiency of the proposed system is discussed. The hardware of the test environment consisted of an i7-4500U 4.2 GHz CPU (only one core was used), DDR3 6 GB RAM, the operating system used was Windows 8.1- 64 bit, and the programming environment was MATLAB 2015.
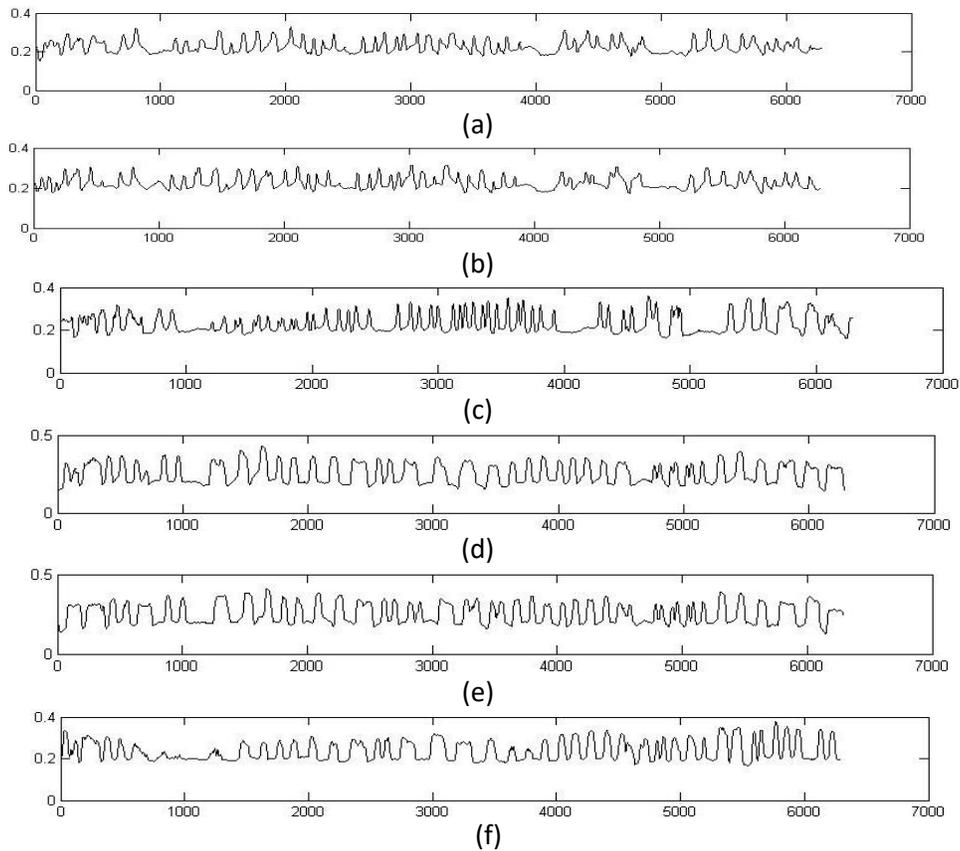
Figure 3.8. Examples of 1-D signals related to the rectangular image of coins for the same row: (a, b) two genuine Danish coins from 2008 and (c) a fake Danish coin from 2008; (d, e) two genuine Danish coins from 1996 and (f) a fake Danish coin from 1996.

3-4-1 Dataset

In this research, we used a total of 322 genuine and 162 fake coins for training and evaluating the system, respectively. The Law Enforcement Office provided the coins used in the research, and we applied all of them. We would note that access to more fake coins was restricted. In Figure 3.9, samples of fake and genuine coins of the years 1990 and 2008 are shown. Table 3.1 indicates four types of Danish 20 Kroner scanned by the 3-D scanner and used in this model verification. In Table 3.2, the average times of scanning, restoration, and feature extraction, are listed.

(a)　　　　　(b)　　　　　(c)　　　　　(d)
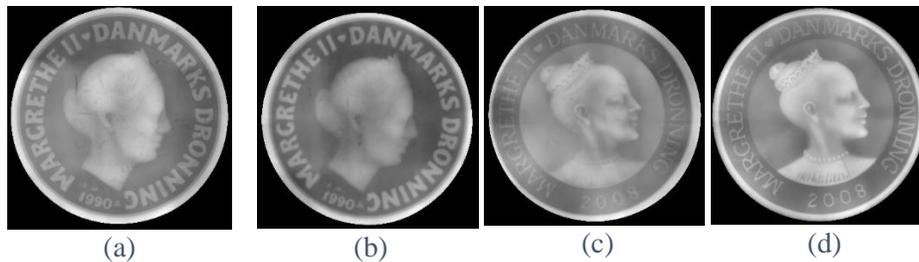
Figure 3.9. Examples of genuine and counterfeit coins: (a) genuine 1990, (b) fake 1990, (c) genuine 2008, and (d) fake 2008.

31

Table 3.1. The properties of coins used in this research.

Year of coin (Danish 20 Kroner)	Number of Genuine coins	Number of Fake coins	Training set		Test set	
			Genuine	Fake	Genuine	Fake
2008	23	113	15	75	8	38
1996	100	10	75	6	25	4
1991	100	14	75	8	25	6
1990	99	25	75	15	24	10
All coins	322	162	225	104	97	58

Table 3.2. Time spent on the steps of the proposed method.

Year of coin (Danish 20 Kroner)	Scanning time (Avg/Minutes)	Restoration time (Avg/Sec)	Feature extraction time (Avg/Sec)
2008	9.8	16.2	1.48
1996	10.2	14.4	1.46
1991	10.6	15.1	1.47
1990	10.4	13.9	1.52

As previously stated, after scanning and performing the proposed method on the coin images, each height-map image whose class (fake or genuine) was clear was converted to a record data. Therefore, the final data set contained 484 labelled records. Consequently, we used various classifiers to illustrate the fitness of the proposed method and extracted features to detect counterfeit coins.

3-4-2 Evaluation Criteria

There are several standard metrics to evaluate a classification system. *True positives* or *detection rate*, *true negatives*, *false positives*, and *false negatives* are often used to evaluate a classification system [41]. For a counterfeit detector, a *true positive* or *detection rate* indicates that the system detects precisely a counterfeit coin. A *true negative* illustrates that the counterfeit detector has correctly detected a genuine coin. The *false positive* is the case when a genuine coin is falsely classified as a fake one. The *False negative* shows that the counterfeit coin detector is unable to detect the fake coin. To improve the accuracy of the evaluations, we calculated the *precision*, *recall*, and *F-value,* which are not very dependent on the size of the train or test dataset. They can easily be computed by Eqns. (3.3), (3.4), and (3.5) where *TP*, *FP,* and *FN* are the numbers of true positives, false positives, and false negatives, respectively. Also, β is the relative importance of precision versus recall and is usually set to one [41]. Furthermore, we considered the time of building a model for classifiers to compare their training time.

$$Precision = \frac{TP}{TP + FP} \qquad\qquad (3.3)$$

$$Recall = \frac{TP}{TP + FN} \qquad\qquad (3.4)$$

$$F - value = \frac{(1 + \beta^2) * Recall * Precision}{\beta^2 * (Recall + Precision)} \qquad\qquad (3.5)$$

3-4-3 Results and Discussions

We used several well-known classifiers available in *Weka* [42] to evaluate the performance of the proposed method. By employing a feature selection algorithm and removing irrelevant and redundant, we reduced the processing complexity and improved the classification efficiency. After using Chi-squared feature evaluation on all 624 features, 36 features were selected.

Tables 3.3 and 3.4 illustrate the results of three metrics comparing the performance of the classification process for the system using 624 extracted, and 36 selected features in the order. In terms of training and test times, *NaïveBayes*, and *Lazy.Kstar* had a shorter time than other classifiers, while neural network multi-layer perceptron was the worst. *TP* (detection rate) and *FP* are weighted and averaged for both the genuine and fake classes. *Stochastic Gradient Descent* or SGD (with loss function: Hinge loss (SVM), learning rate: 0.01, epoch: 500, and lambda: 1.0E-4), and MLP (with learning rate: 0.2, momentum: 0.3, number of sigmoid nodes in hidden layer: 20) had better results compared to others. The *decision tree* was the weakest classifier in these metrics. These classifiers for both experiments (624 features, and 36 features) showed roughly the same superiority in the mentioned criteria, although the performance of the classification was dramatically improved after feature selection. SGD has correctly classified 99.3 % of the coins by 36 features; for the other classifiers, we can see acceptable results in the metrics. Taking *precision*, *recall*, and *f-value* into consideration, SGD was still the most remarkable classifier, according to Table 3.5. In spite of the relatively weaker results for *decision tree*, and *NaiveBayes* in the classification of the fake coins, the other classifiers could classify both fake and genuine coins satisfactorily. As can be seen from Table 3.5, *precision*, *recall*, and *f-value* were indispensable to measure the accuracy of the classifications. The results showed that although *NaiveBayes* and *decision tree* had an appropriate outcome in the detection of the genuine coins, they were slightly weaker in detecting the counterfeit coins. Apart from the ability of classifiers in the mentioned standard metrics, all the results are satisfactory for the proposed method and demonstrate that the extracted features are very effective for any kind of classification algorithms.

Very few researchers have used image-based coin recognition techniques to detect counterfeit or invalid coins. Therefore, the proposed method was compared with two recent counterfeit coin detection methods. Albeit the test time is not reported well in these two methods, it can be concluded that in reference [15], researchers designed a real-time system, and the method performs much better than the other methods in term of processing time. The possible metrics for a more precise comparison of these methods are classification accuracy and test data information. As Table 3.6 shows, the proposed method performs better than the reference [15] in terms of accuracy, but its accuracy is not as precise as reference [9]. However, with respect to the test process of these methods, there is another issue that could have a straightforward effect on accuracy. Although the accuracy resulted in the method of reference [9] is 100%, the size of the test set for evaluation of the system was very small. Also, it is clear that our proposed method was trained and tested by four different types of coins (years 1990, 1991, 1996, and 2008) while in [9] only two types of coins (years 1990, and 1996) were considered.

In order to indicate the impact of the restoration process proposed in Section 3.2 on this paper, we tested the coins with different qualities separately. As Table 3.7 demonstrates, the signal restoration had a significant impact on the classification of the degraded images resulted from scanning of the shiny coins. It is also worth noting that all six oxidized coins (4 fake and 2 genuine) classified correctly. It means that a 3-D scanning of an oxidized coin can produce an image as good as a normal coin.

Table 3.3. Comparing the classification results for various classifiers using 624 features.

Classifier	Training Time (sec)	Test Time (sec)	True Positive (%)	False Positive (%)
SGD	0.67	0.23	97.8	7.8
MLP	157.9	21.3	94.9	26.7
NaiveBayes	0.03	<0.01	92	12.6
Decision tree	0.11	<0.01	90.5	38.6
Lazy.Kstar	< 0.01	<0.01	83.9	83.9
Logistic	0.63	0.2	94.49	4.7

Table 3.4. Comparing the classification results for various classifiers using 36 features.

Classifier	Training Time (sec)	Test Time (sec)	True Positive (%)	False Positive (%)
SGD	0.05	<0.01	99.3	3.8
MLP	1.52	0.71	97.1	4.2
NaiveBayes	< 0.01	<0.01	92.0	12.6
Decision tree	0.02	<0.01	91.2	31.1
Lazy.Kstar	< 0.01	<0.01	95.6	15.5
Logistic	0.02	<0.01	95.6	4.5

Table 3.5. Comparing the results of classifiers in terms of precision, recall, and f-value for classifying fake and genuine coins.

Class	Metric	Classifier					
		SGD	MLP	Naïve Bayes	DT (J48)	LazyKstar	Logistic
Genuine	Precision	0.991	0.991	0.973	0.933	90.966	0.991
	Recall	1	0.974	0.93	0.965	0.983	0.957
	F-value	0.996	0.982	0.951	0.949	0.974	0.973
Fake	Precision	1	0.875	0.704	0.778	0.9	0.808
	Recall	0.955	0.955	0.864	0.636	0.818	0.955
	F-value	0.997	0.913	0.776	0.7	0.857	0.875

Table 3.6. Comparison of different methods with the proposed counterfeit detector in terms of accuracy.

Accuracy		[15]	[9]	Proposed method using SGD classifier
		97 %	100 %	99.3 %
Testset information	Genuine coins	100	3	97
	Fake coins	96	2	58
	Type of Coins	Two-Euro	20 Kroner 1990, 1996	20 Kroner 1990, 1991, 1996, and 2008

Table 3.7. Impact of the restoration process on the classification of the coins with different qualities.

Coin Quality	Number of Test samples	Correctly classified (%)	
		Before restoration	After restoration
Worn out coins	6	100	100
Degraded images	81	59.4	98.7
Normal coins	68	100	100
Overall	155	78.9	99.3

3-5 CONCLUSION AND FUTURE WORKS

In this paper, a new counterfeit coin detection method based on analyzing and restoration of 3D images was proposed. We used a 3-D scanner to scan a large number of coins and obtained the coin height-map images. To simplify and reduce the complexity of the proposed method, we suggested a straightening algorithm to convert each circular coin image to a linear rectangular image. Since the shiny coin images were abnormally degraded, it was impossible to use height and depth information to detect counterfeit coins. Because of that, the next step was to restore the images by processing the images signal-wise. After restoration, the 1-D signals of the image, the energy, and the percentage of the energy of the wavelet details in level two were extracted for all 311 signals. In sum, we had 624 features before the feature selection process. The comparison of the performance of several well-known classifiers in *Weka* illustrated that the proposed method

gives excellent results in detecting counterfeit coins. Employing multi-processing allowed easily restoring the signals and reducing the processing time. In addition, there was no need to perform any of the well-known methods for noise reduction or and contour tracking for coin recognition. Surprisingly, the proposed method had significant results on counterfeit coin detection using four kinds of coins to train and test the method. Finally, the proposed counterfeit detection method is robust against rust, dust, and sulfation, cases of which were not considered in most of the previous works.

Our plan for future work is to focus on the drawbacks of the proposed method. In spite of the substantial precision and acceptable time for feature extraction, the time of coin scanning needs improvement. To overcome this problem, we can use a 3-D scanner to capture only a small part of the coin and work on it. Enriching the feature extraction and using a hybrid classifier built especially for the created data set can improve the performance of the counterfeit detector. Another direction of our research will be towards coins with non-circular shapes and ones with no text embossing.

Article 2

RESTORING HEIGHT-MAP IMAGES OF SHINY COINS USING SPLINE APPROXIMATION TO DETECT COUNTERFEIT COINS

Saeed Khazaee, Maryam Sharifi Rad, Ching Y.Suen

CENPARMI, Concordia University, Montreal, Canada
{s_khaza, sey_shar, suen}@encs.concordia.ca

Published in:

Abstract. There are presently a lot of companies, museums, and government agencies in the world that have increased the demand for automatic systems to classify precious, historical, and common coins. Thanks to the increased exigency on the intelligent approaches to recognize counterfeit coins, coin detection has continuously been evolving in recent years, and image-based coin recognition has become an integral part of this research area. In this research, we have been studying various methods related to image-based coin detection in 2-D image processing, and we also used a 3-D scanner to scan hundreds of coins to apply 3-D approaches to detect counterfeit coins. Here, instead of conventional 2-D methods for counterfeit coin detection, we applied a 3-D approach to model and analyse a large set of different Danish coins. One of the most important advantages of 3-D approaches is the ability to extract features that cannot be found in 2-D images. Despite this advantage, we had a lot of unexpected degradations on shiny coin images. In order to restore the degraded images, we proposed a method based on signal separation in this paper. The experimental results showed that the proposed method outperformed previous methods in detecting counterfeit coins.

Keywords. counterfeit coin detection, spline approximation, 3D images

4-1 Introduction

An image-based coin recognition system takes digital images of coins as input and classifies them according to their properties. Pursuant to the previous works, coin detection is mostly based on 2-Dimensional image processing techniques. In particular, edge information had been extracted as features with Fourier transform. The difficulty level of coin detection heavily depends on the backgrounds of the coin datasets used in the experiments. Fortunately, all the datasets used for coin detection are captured with plain backgrounds, such as pure white or conveyor belt with extremely low gray-level [1].

Some of the common features in coin recognition like coin diameter, thickness, weight or shape may be used to detect counterfeit coins. We should note that these basic attributes are easy to copy, and the fake coins made nowadays are of high quality. Therefore, these systems cannot distinguish fake coins from genuine ones when their physical attributes are basically the same. To increase the potential of fake coin detectors and image-based coin detection, several methods based on image processing techniques and classification algorithms have been proposed [10], [15], [43] and many lectures and tutorials were devoted to them [37], [44]. Some of these methods are not very sophisticated and use coins' colors and radius-based features to detect counterfeit coins. Unfortunately, these approaches are incapable of distinguishing coins having the same values of the limited set of features.

In recent years, image-based counterfeit detection has expanded, and many researchers have applied image processing techniques to extract effective features from the texture of the coin images [36]. In general, edge detection has been frequently employed in the feature extraction process. In references [9], [38], an edge map was extracted and segmented into several parts. After that, the authors proposed a method using histogram analysis and Fourier transformation to handle and recognize rotated coin images. However, the extracted features were not very useful for noisy and degraded images such as rust, dust, or sulfated coin images. In [9], the authors extracted the letters on the coin surface and tried to recognize them. In spite of the mentioned novelties of the proposed method, it is clear that the method was not robust enough to distinguish the counterfeit from genuine coins whose images have weak or smooth edges. In reference [11] a new method using rotation-invariant region binary patterns based on gradient magnitudes was proposed. To increase the accuracy of coin recognition, it computes gradient magnitudes in a coin image and extracts rotation-and-flipping-robust features using local difference magnitude transform.

Although the result is acceptable from the feature extraction point of view, the time-performance of the model is not satisfactory.

In references [8] and [12], authors proposed their methods to recognize coin images in terms of their denominations and countries of origin. However, the problem of coin recognition significantly differs from fake coin detection investigated in this study. For coin recognition, one important concern is to reduce the sensitivity to the variations among the coins from the same class. Therefore, it is highly probable that the fake coins, especially those of high quality, are classified as belonging to the same class as their genuine counterparts under the coin recognition framework.

As we discuss later, we have been using a three-dimensional scanner to create an image data set for Danish coins. In this research, during the study on 3-D approaches to extract effective features, we have faced many challenges related to shiny coins and tried to solve this problem. There are many different technologies used in 3-D scanners; each technology has its own advantages, restrictions, and cost. For example, optical technologies face many difficulties when processing, reflecting, transparent, or shiny objects. Figure 4.1 illustrates how a 3-D scanner captures height and depth: it is robust regarding the quality of the coin surface, while the 2-D image of this coin is completely unreadable. In Figure 4.2, we can see the degradation of a shiny coin image captured by the 3-D scanner, which uses optical technology. As it can clearly be seen from this figure, in spite of the remarkable advantages of 3-D scanning for poor quality coins, there may be some abnormal and invalid results while scanning shiny and high-quality coins [45]. However, the validity of the data captured by scanning is crucial for this research as an invalid value related to height and depth will adversely affect the pattern recognition process. Therefore, our proposed restoration module should enhance not only the quality of the images as a whole but also each small part of the image must be restored as precisely as possible.

The rest of this paper is organized as follows. Section 4.2 describes the proposed method to detection counterfeit coins. In this section, we will explain our proposed image restoration and feature extraction. In Section 4.3, we talk about the experimental results and analyze them, and we explain our new method to edge plane detection. Finally, we conclude in Section 4.4 and comment on some future works.

(a)   (b)
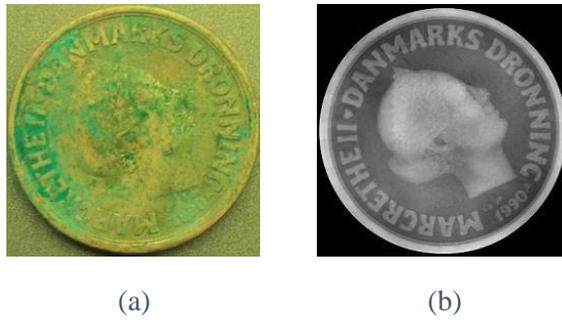
Figure 4.1. (a) Twenty Kroner 1990 coin captured by a normal camera, and (b) the same coin captured by a 3-D scanner.
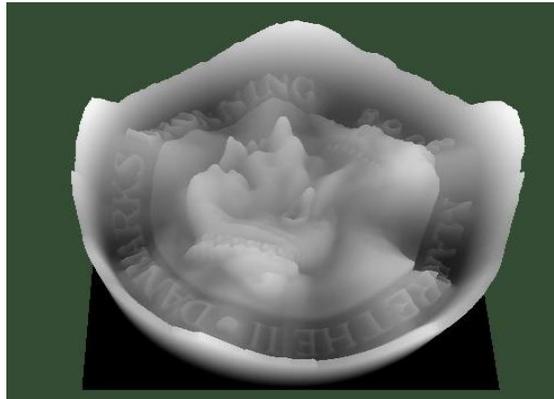
Figure 4.2. Degradation problem for a shiny coin's image in a height-map image (wrong height and depth information.

4-2 COUNTERFEIT COIN DETECTION

4-2-1 Restoring Images Using Spline Approximation

After the straightening process, which we proposed in [45], we can see an adverse effect of improper scanning on shiny coin images. For some kinds of coins, for example, in the Danish 20 kroner 2008 dataset, all the images are totally degraded. Our initial experiments showed that the degradation problem is not random noise. We scanned three coins, four times for each coin, and we obtained the same poor results [45].

Since our proposed counterfeit coin detector (Figure 4.3) is based on height and depth information, signal restoration from this digital data is an inevitable process. For a given digital data set obtained from scanning, we needed to recover the original signal, which includes more precise data, since the degraded signals related to fake and genuine coins are very similar. Here, the important information of letters and numbers corresponds to a low-frequency signal. Therefore, we must

estimate the low-frequency signal and subtract it from the original signal. Thus, we have $x=A+D$; where x is the original and degraded signal, A is an approximation of x, and D contains the details of x. In other words, the useful information comes from the details of the signal.

In our first attempt, we used wavelet transform to find A or D. For restoring all signals and the image, and several well-known methods were performed.

For estimating more accurately the degradation function for each 1-D signal, we proposed an estimator, which uses control points determined by the Catmull-Rom spline. Figure 4.4 shows the basis, control matrix used for estimating a Catmull-Rom spline in a specific period. Since the signals, on which the process should be performed, are not smooth enough to perform mathematic operations, we used a low-pass filter to smooth them. The critical selected point or control point is routinely an extremum point selected by the effective point selector function. This function uses momentum to avoid local minima, which should be ignored, and the signal has a lot of them. In addition, if the distance between two control points is greater than a specific threshold, the point selector will consider several points on this part of the signal instead of the critical points.

The principal advantage of this technique is that the points along with the original set of points also make up the control points for the spline curve. Two additional points are required on either end of the signal. Figure 4.5 illustrates a small part of the original signal and the approximation of the degradation signal.

Consequently, we used a low-pass filter again to obtain the final signal, and we have $A = inverse(F * H)$ where F and H are Fourier transforms of f, and a 1-D Gaussian filter in the order. Also, the $inverse$ function returns the inverse discrete Fourier transform of the resulted vector. In addition, there is a simple parameter optimizer which regulates the value of σ in Gaussian filter, with the aim of minimizing the Euclidean distance between A and f. In this module, a distance between each signal and a fixed counterpart signal will be calculated. The fixed signal is an average of a signal related to each row of the straightened images which are not degraded. Then, the optimizer can optimize the proper values to filter the images by Fourier transformation. Figure 4.6 demonstrates that the original image has been restored successfully. The signal related to the marked row of the restored image shows that the heights of the letters are more precise than the original rectangular image. Figure 4.6 also shows that the proposed restoration produces more balanced height information for the surface of the coin. In the top of the figure, we can see a 3-D

view of the straightened image of a shiny coin, and in the bottom, the restored image by the proposed method.
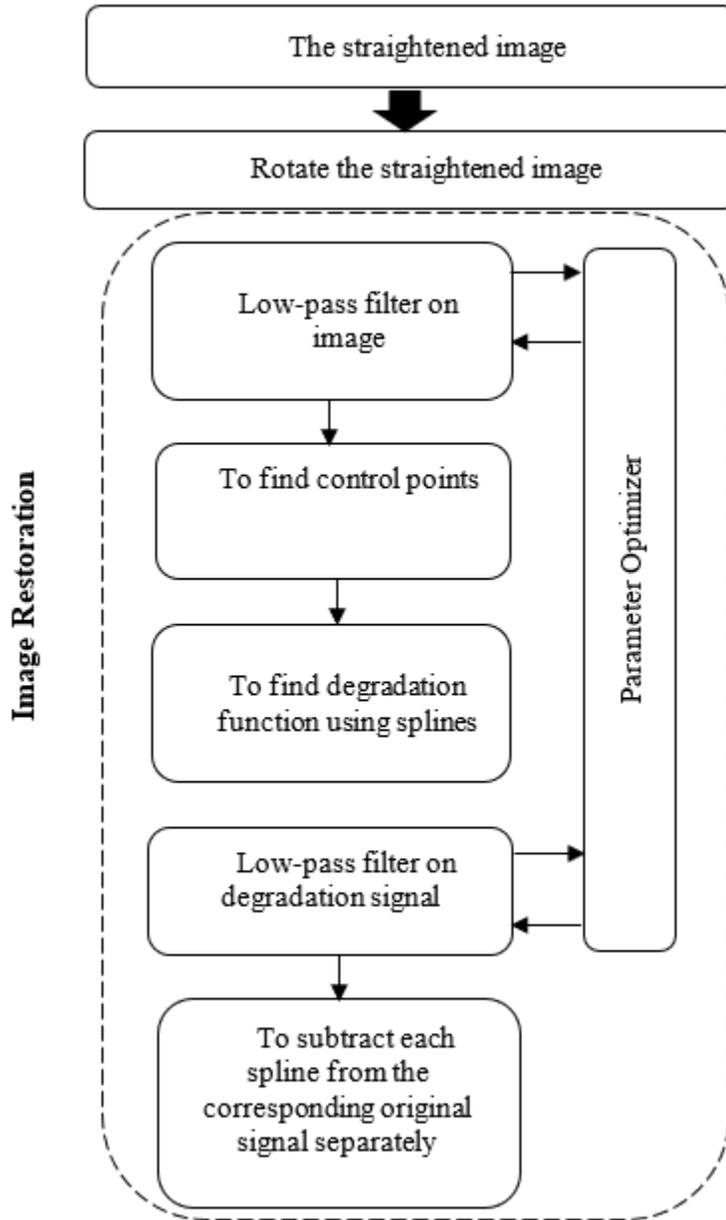
Figure 4.3. Proposed method for restoring the signals.

Catmull-Rom Spline for $t \in [-1,+1]$

$C[1]$ $C[5]$

$C[3]$

$C[4]$ $C[7]$

$C[6]$

$C[0]$

$C[2]$

$C[3]-C[1]$

0.4

$t=-1$ $\quad$ $t=-0.6$ $\quad$ $t=-0.2$ $\quad$ $t=0.2$ $\quad$ $t=0.6$ $\quad$ $t=1$

$$
\begin{bmatrix} x & y & z \end{bmatrix} = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \begin{bmatrix} -s & 2-s & s-2 & s \\ 2s & s-3 & 3-2s & -s \\ -s & 0 & s & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \\ x_4 & y_4 & z_4 \end{bmatrix}
$$

basis $\qquad\qquad$ control matrix
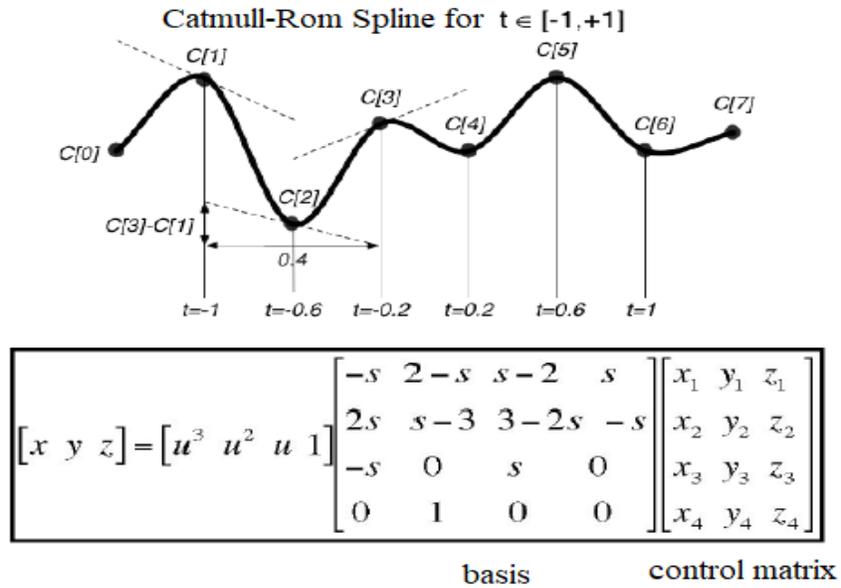
Figure 4.4. Basis and control matrix to approximate the spline [46].

■ Original (degraded) signal

■ Spline

■ Restored signal

1000 $\qquad$ 2000 $\qquad$ 3000 $\qquad$ 4000 $\qquad$ 5000 $\qquad$ 6
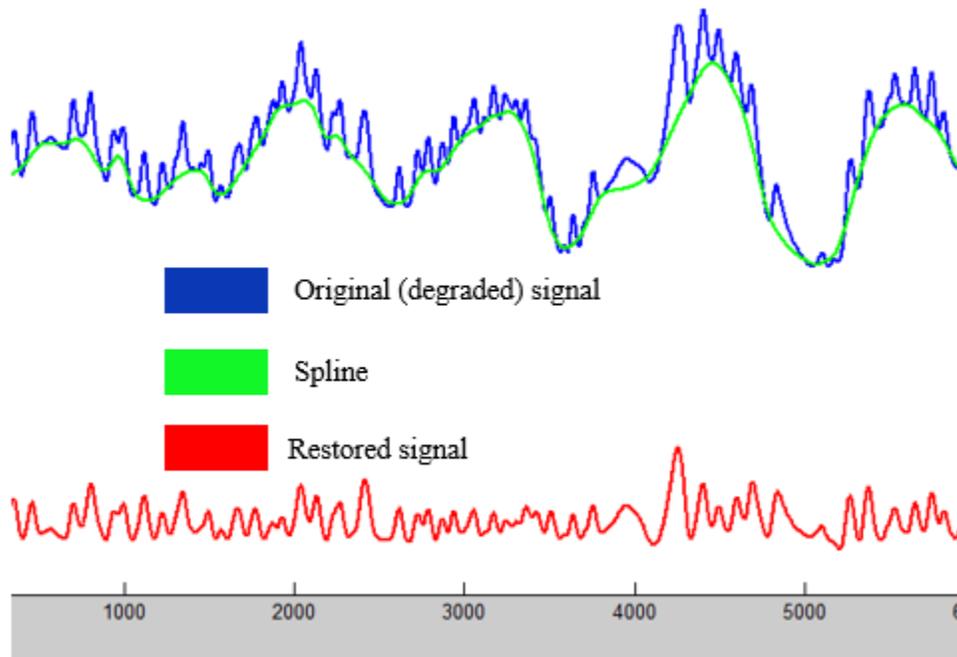
Figure 4.5. Example of a restored signal corresponds to a specific row of the straightened image obtained from the original and the spline approximation.

4-2-2 Feature Extraction

Here, we consider one 1-D signal as a representative of the straightened and restored image. To achieve this aim, we calculated the median of each column of the matrix instead of all rows and

made a new vector or signal, which we call it, the median signal. The median signal of different coins' images is shown in Figure 4.7. Figures (a) through (c) show that the median signal of original coins which are ideally similar, while in Figure 4.7 (d) it is clear that the median signal of the fake coin is totally different from the genuine coins' pattern. Therefore, the median signal is a desirable representative for the image, and we have one 1-D signal to extract features.

Figure 4.6. Restoration of the degraded image in 3-D view.

(a)

(b)

(c)

(d)
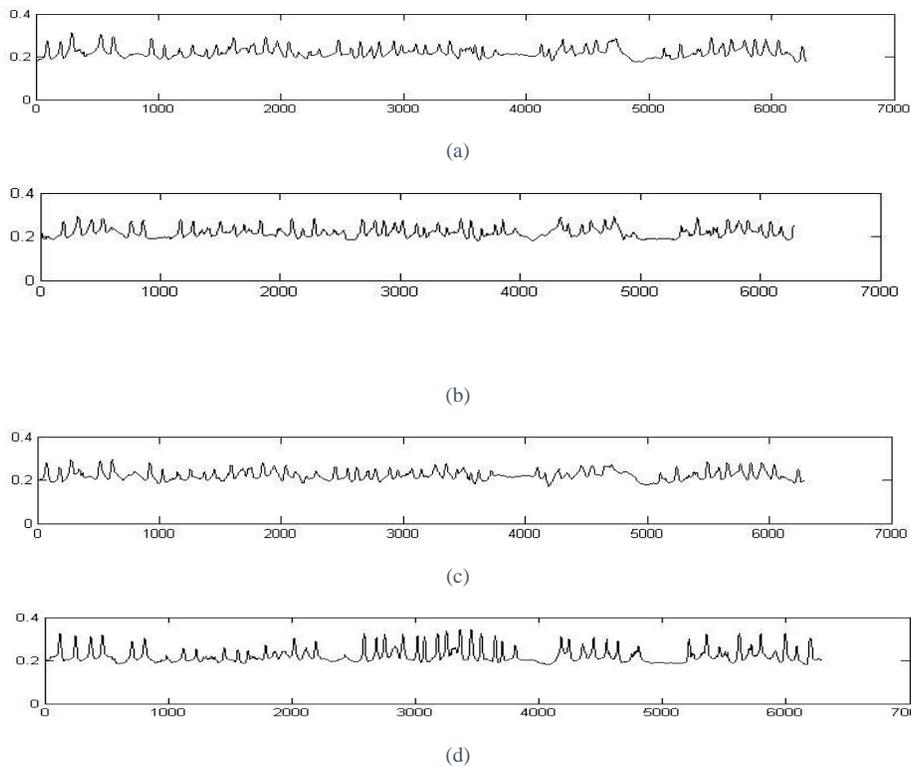
Figure 4.7. Figures (a) through (c) are the median signals of three genuine Danish 2008 coins, and (d) the median signal of a fake Danish 2008 coin.

Extracted features in the feature extraction module were quite dependent on the time (horizontal parameter of the signal) and starting point. Here, for extracting a lot of features, we must consider a base signal and shift other signals to the same position.

1) Shifting the Median Signal

To shift the median signal to the correct position, we considered a signal as a base signal and shifted the signal in the right direction until a similarity criterion was satisfied. To calculate the similarity between the signals, the cross-correlation has been used. From the cross-correlation function, we can obtain the correlation coefficient, which will give us a single value of similarity. Right from the start, the similarity of the median signal and the base signal was calculated then the median signal was shifted to the right. The best position for the median signal was marked when the similarity was the highest value. Figure 4.8 (a) shows the median signal of one coin, which we consider as a base of shifting. Figure 4.8 (b) is the shifted signal of (c). The figure shows that the signal was shifted to the best position. After shifting the median signal, we extracted two different kinds of features. The first kind of features were the statistical features, and the next ones were the information about the band power, bandwidth, and distortion. Experimentally, to extract these features, we divided the signal into four parts and extracted the features for each part separately.

(a)

(b)

(c)
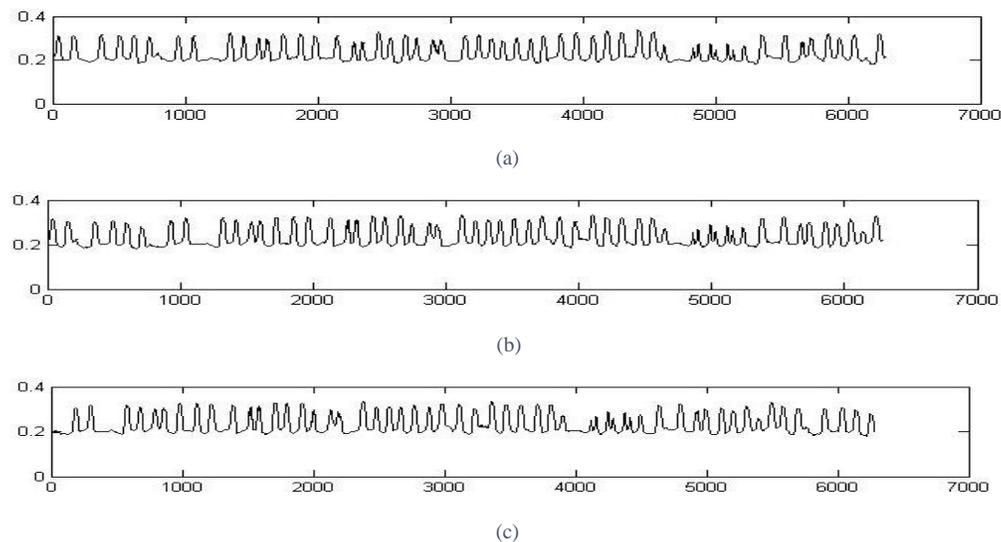
Figure 4.8. The base signal, (b) the shifted signal, (c) the original median signal.

2) Statistical Feature

The first statistical feature extracted from the signal is the period of the sequence that is computed as the minimum length of a subsequence $xi(1:p)$ of signal x that repeats itself continuously every

45

p samples in xi. The length of x does not have to be a multiple of p, so that an incomplete repetition is permitted at the end of xi. If the sequence x is not periodic, then $p = length(xi)$. To extract four next features, we estimated the mean normalized frequency of the power spectrum of a time-domain signal, xi. Features 9 through 12 are the ratio of the largest absolute value in x to the root-mean-square (RMS) value of xi, which are also called Peak-magnitude-to-RMS ratio. In all explanations, xi is one of the four parts of the original signal and $i = 1,2,3,4$.

3) Peak Analysis Features

To extract features based on peak analysis, we found the prominent peaks as far as the number of the peaks in all signals were equal. Then we estimated the average distances between peaks for each xi. Therefore, there are four features based on peak analysis. Finally, we have 16 features through the feature extraction process.

4-3 EXPERIMENTAL RESULTS

In this paper, various well-known classifiers were employed to evaluate the performance of the proposed method. We used CENPARMI Danish dataset introduced in [45]. Table 4.1 illustrates the results of three metrics comparing the performance of the classification process for the system using 16 extracted features. In terms of Detection Rate or TP (True Positive) and FP (False Positive), which are weighted and averaged for both genuine and fake classes, Stochastic Gradient Descent (SGD), and MLP had better results than others. SGD has correctly classified 99.3 % of the coins tested. For the other classifiers, we can see the acceptable results in the metrics. Turning to precision, recall, and f-value, SGD was still the most remarkable classifier in Table 4.1. Despite the weaker results for NaiveBayes and decision tree in detection rate, other classifiers could classify both fake and genuine coins satisfactorily. Table 4.2 demonstrated that although NaiveBayes and decision tree had an acceptable outcome in the detection of the genuine coins and were very fast in the building model, they were relatively mediocre in detecting the fake coins rather than the other models. Apart from the capability of classifiers in the mentioned standard criteria, all the results are satisfactory for the proposed method and illustrate that the extracted features are significant.

Table 4.1. Comparing the classification results for various classifiers using 16 features.

Classifier	Time taken to build model (sec)	TP (%)	FP (%)
SGD	< 0.01	99.3	3.8
MLP	0.6	98.4	3.6
NaiveBayes	< 0.01	92.7	10.4
Decision Tree (J48)	< 0.01	92.8	26.1
Lazy.Kstar	< 0.01	96.8	15.5
Logistic	< 0.01	96.6	4.5

Table 4.2. Comparing the results of classifiers in terms of precision, recall, and f-value for classifying fake and genuine coins by 16 features.

Class	Measure	Classifier					
		SGD	MLP	NaïveBayes	J48	Lazy.Kstar	Logistic
Genuine	Precision	0.991	0.993	0.981	0.933	0.966	0.993
	Recall	1	0.976	0.936	0.965	0.983	0.963
	F-value	0.996	0.987	0.956	0.949	0.974	0.978
Fake	Precision	1	0.90	0.712	0.778	0.90	0.812
	Recall	0.955	0.964	0.864	0.636	0.818	0.964
	F-value	0.997	0.927	0.776	0.7	0.857	0.881

4-4 CONCLUSION AND FUTURE WORK

We have been trying to find a comprehensive way to detect counterfeit coins, which can analyze the coin images as precisely as possible. In our preliminary counterfeit coin detector, the comparison of the performance of several well-known classifiers in Weka [42] illustrated that the proposed method produces excellent results in detecting counterfeit coins. Employing multi-processing will facilitate the restoration of the signals and the reduction of the processing time. Surprisingly, the proposed method had a remarkable outcome on counterfeit coin detection when we use the four types of coins for training and testing together. Finally, it was apparent that the proposed counterfeit detection method is robust against rust, dust, and sulfation, cases. However, the biggest challenge in detecting fake coins is always the lack of fake samples. In some cases, it is nearly impossible to have fake coins as many as we can build a model or test our system. Hence, investigating the feasibility of using unary (one-class) classification and PU (Positive and

Unlabeled samples) learning for improving the accuracy of the counterfeit coin detector is open for future works.

Article 3

DETECTION OF COUNTERFEIT COINS BASED ON 3D HEIGHT-MAP IMAGE ANALYSIS

Saeed Khazaee, Maryam Sharifi Rad, Ching Y.Suen
CENPARMI, Concordia University, Montreal, Canada
{s_khaza, sey_shar, suen}@encs.concordia.ca
Under review in:
Expert Systems with Applications, Elsevier

Abstract. Detecting a counterfeit coin using 2D image processing is nearly impossible in some cases, especially when the coin is damaged, corroded or worn out. Edge detection is one of the most widely used techniques to extract features from 2D images. However, in 2D images, the height information is missing with losing the hidden characteristics. In this paper, we propose a 3D approach to detect and analyze the *precipice borders* from the coin surface and extract significant features to train an ensemble classification system. In order to extract the features, we also propose Binned Borders in Spherical Coordinates (BBSC) to analyze different parts of *precipice borders* at different polar and azimuthal angles. The proposed method is robust even against degradation, which appears on shiny coins after 3D scanning. Therefore, there is no need to restore the degraded images before the feature extraction process. Here, the system has been trained and tested with four types of Danish and two types of Chinese coins. We take advantage of stack generalization to classify the coins and add the reject option to increase the reliability of the system. The results illustrate that the proposed method outperforms other counterfeit coin detectors. The accuracy obtained by testing Danish 1990, 1991, 1996, and 2008 datasets are 98.6%, 98.0%, 99.8%, and 99.9% respectively. In addition, results for half Yuan Chinese 1942 and one Yuan Chinese 1997 were 95.5% and 92.2%, respectively.

Keywords. Counterfeit coin detection, 3D *precipice borders*, ensemble classifier, height-map images.

5-1 INTRODUCTION

Over the past several decades, a lot of counterfeit money has caused significant damage to the society. They have been made by criminals or unknown coin collectors for thousands of years as an illegal industry [47]. It is worth noting that as a significant topic of security, counterfeit coin detection has become the focus of research in the field of numismatics. The data drawn from the Counterfeit Monitoring System (CMS) indicate that only in Europe, the total number of counterfeit euro coins removed from circulation from 2013 to 2017 was equal to 837,910 pieces whose value amounted to 1,330,401 Euros [2]. It is expected a very large number of counterfeit euro coins still remain in circulation. In addition, as an example of an old coin which is being traded in the market, we can mention a half Chinese Yuan 1942. The price of the best quality of this coin in the market is over $1200, which illustrates the importance of counterfeit detection for old coins.

With the continuous development of 3D technologies and the fast evolution of depth sensors attached to handheld devices, 3D approaches have become a hot topic of computer vision due to its applications in areas such as recognition, security, and biometrics. It also motivates the extension of novel image processing and computer vision techniques. Because of the progress of 3D applications for different purposes, the notion of 3D approaches has determined itself as one of the most significant alternatives for 2D approaches and has attracted considerable attention of the community in recent years. In addition, it is worth noting that 3D approaches are gradually emerging as an appealing area for the design and implementation of a classification scheme, especially by those employing image datasets. Up to now, several studies based on image processing techniques and classification algorithms have been proposed that exploit images for counterfeit coin detection. Fundamentally, fake coin detection is an effortful procedure because of widely varying input templates, cluttered images, and various rotations, which are the big challenges. Most of the existing image-based methods for counterfeit coin detection relied only on 2D images, failing to equip some statistical information about the height and depth, and often lose the hidden characteristics or suffer from low accuracy. Although researchers have recently achieved fruitful results in 2D image-based systems, these studies have an important drawback. It is not conceivable to rely only on 2D images, when the coins are corrupted, colored, or worn out, where 2D detected edges are the only source of information. As a remedy to this limitation, the development of techniques to establish a 3D structure from a coin image can discover the complex characteristics of the coin.

Based on the type of analysis carried out, the potential of a 3D image-based method has not yet been applied for counterfeit coin detection and is still an open field of research. Despite the significance of the 3D technologies, the research in this regard is relatively scant. In this paper, we propose a 3D approach to detect and analyze the coin surface and extract significant features to train an ensemble classification system. To do this, a novel method to detect specific parts of objects on the coins that we name it Precipice Border Detection Algorithm (PBDA) will be proposed, which is incorporated into the proposed framework for counterfeit coin detection. Considering the nature of the data in height-map images, we extract effective 3D features related to height or depth of the coin surface and prove the robustness of the method with cases not considered in previous works [45], [48]. In this method, the height-map image of the coins will be triangulated, and a clustering scheme based on the Fuzzy C-Means algorithm [22], [23] will cluster the triangles to detect *precipice borders* of the coin surface. Then, we propose a method to analyze the *precipice borders* and extract the valuable features for training a stacking classifier with a reject option.

The focus of this paper is to propose a method to detect counterfeit coins. However, the proposed algorithms have much broader applications in the context of 3D image classification. The major contributions of the proposed framework are summarized as the following:

a) Creating six height-map image datasets and proving the competency of 3D approaches in counterfeit coin detection: We successfully detect the *precipice border* of the surface on the coin with our proposed 3D Precipice Border Detection Algorithm (PBDA) and use it for the feature extraction process.

b) No need for image restoration or enhancement for degraded images: The *precipice borders* are not affected by the degradation problem.

c) Extracting features with a high discriminating capability: We propose a system to consider the direction of the *precipice borders* as well as their approximate areas for the feature extraction module that we name it Binned Borders in Spherical Coordinates (BBSC).

d) Feeding an ensemble classifier by feature matrices: A feature extraction methodology is proposed to extract a feature matrix instead of a feature vector in which each row of the matrix is used separately.

The rest of the paper is organized as follows. To illustrate the current state of research on the field of image-based coin detection, we provide a literature review in Section 5.2. The preprocessing steps like resizing, rotation, and normalization are described in Section 5.3. Section 5.4 presents the concept and proposes a method for *precipice border* detection and analysis. Section 5.5

51

discusses to design an efficient ensemble classifier with a reject option for the classification process. In Section 5.6, the experimental results are given to show the performance of the proposed method regarding counterfeit coin detection. Finally, the paper concludes with a summary of the primary contributions of this research and suggests an outline for future work in Section 5.7.

5-2 LITERATURE REVIEW

In recent years, several studies have been made to recognize coins through different methods such as Gabor filter, Hough Transform, Heuristics, and Artificial Neural Networks. Recently, several image-based approaches to detect the fake coins have been proposed that extracted effective features from the texture of the coin image [8]. In particular, edge detection information has been widely used in the feature extraction process. In reference [9], an edge map was extracted and segmented into several parts. After that, the letters were extracted from the coin image, and the accuracy for fake coin detection was around 100%, while a small dataset tested the system. Although the results mentioned in the paper were significant, the method was not applicable for any other type of coin as it was limited to letters. In reference [10] the authors proposed a method to extract local texture features for coin recognition. They used Gabor wavelets and local binary pattern (LBP) to detect texture information. In the proposed method, a circular shift operator was used to provide the robustness of Gabor features against rotation variance. In reference [11], the authors proposed a region binary patterns method which was rotation-invariant. They applied gradient magnitudes instead of histogram analysis and extracted rotation-and-flipping-robust features using local difference magnitude transform. For matching between two-coin images, a distance measurement was used. Despite the acceptable feature extraction time, the experiments in this paper illustrate that the method was not tested for poor quality coins and might be sensitive during matching coins with high similarity. In reference [12], the authors proposed a multistage procedure that could recognize hundreds of coin types. In spite of the novelties and results, this approach was not able to distinguish fake from genuine as it did not concentrate on the similarity of counterfeit and genuine coins. Recently, the authors in reference [1] proposed a new image-based approach to detect counterfeit coins. There, the local descriptors were employed to generate image representations. Instead of resorting to the BOVW (Bag of Visual Words) model for a generation of vectorial representations, the authors represented a coin image in the dissimilarity space. They conducted one-class learning for fake coin detection. However, the method mentioned was sensitive to any variation and was not suitable for poor quality coins and degraded coin images.

Apart from image-based methods, there are some non-image-based methods for counterfeit coin detection. In [15], the authors proposed a method using the sensor of an optical mouse to detect counterfeit coin. They applied this method to the two-Euro coins and compared partial images of the coin under analysis with some partial reference coin images for matching in the detection process. The report in [15] also shows that the system cannot identify most of the well-forged coins as counterfeit. In [16] a method based on acoustic signal spectrum analysis for authentic and counterfeit coins has been proposed. In this paper, the authors optimized the parameters for time-spectrum co-analysis. They used a very small dataset of Euro coins to validate their methods. However, low costs and high accuracy classification were reported as properties of the method.

In recent years, studies on image processing techniques illustrate that since 3D approaches discover highly confident features, they can overcome some constraints introduced by other 2D image classification methods. As plane detection in height-map 3D images plays an important role in 3D object classification, much research is conducted to detect and analyze the planes for recognizing specific objects in a 3D scene. For example, the authors in [17] proposed a depth image-based plane detection algorithm. As stated in this paper, the algorithm could be used as preprocessing for planar object recognition that has a fast run-time compared with other methods. In [19], a method based on disparity transformation and 3D road surface modelling has been proposed for pothole detection. They used two stereo cameras for road data acquisition and finally extracted the point clouds of the detected potholes from the reconstructed 3D road surface and got remarkable results rather than 2D based images. In [20], a method of edge plane detection in three-dimensional images has been introduced by utilizing features of edge vector and edge reliability. This method can detect only reliable edge planes against noise influences. In [21], a method based on Hough transform for plane detection was proposed. The method could identify the clusters of coplanar points in the 2.5-D space using an implicit quadtree. They also applied a Hough-transform voting structure for the detection process to model "the uncertainty associated with the best-fitting plane" considering each cluster as a trivariate-Gaussian distribution. As concluded in this paper, the method was a real-time technique for plane detection.

Though at first glance, the *precipice border* detection is similar to the plane detection, there is an important distinction between them in 3D height-map image processing. In the edge plane detection, we find a set of planes that are very useful to recognize and distinguish different objects in a scene while in the *precipice border* detection, we find the curved surfaces and complicated details on an object. Therefore, in a special task like recognizing a coin that has a very complex

geometry, plane detection would not be very useful for feature extraction, while by analyzing the *precipice borders*, we extract features capable of discriminating the high-similarity objects.

5-3 PREPROCESSING

All the coins used in this study have circular shapes. However, there are rare cases that the output of scanning is an ellipse instead of a circle for a coin image. Therefore, the Hough transform is employed for ellipse detection [49] as the first preprocessing step for separating the coin from the background. The original resolution of the height-map images is *3550 × 3550,* and we resize them to *400 × 400* with the gray level of 0 to 255. Despite the advantages of 3D scanning [5], [6], there has been a serious challenge when shiny coins are processed [6]. As we encountered a lot of unexpected degradation and shadowing on shiny coin images, we faced wrong values of height or depth. Figure 5.1 (a) illustrates the degraded image after scanning a shiny coin image. The restored image by a proper filter is demonstrated in Figure 5.1 (b) while missing a lot of height information. To solve the above problems, we have two options:

a) Restoring the degradation problem without losing height information for which we provided two different proposed methods in our previous works [45], [48].

b) Ignoring the degradation problem for which in this paper, we propose a new method to analyze the *precipice borders* that are not affected by the degradation problem. Robustness against this problem is one of the advantages of our proposed methods.

(a)   (b)
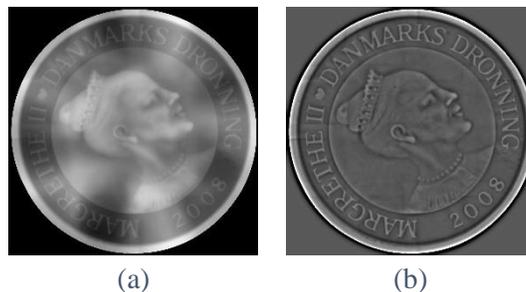
Figure 5.1. Degradation problem related to the 3D scan of the shiny coins: (a) an original image after 3D scanning, (b) restoration of the degraded image by high-pass filtering.

As the coins scanning process was performed manually, variations in the scale and rotation are inevitable. In this research, we consider the passive transformation in which the coordinate system is rotated counterclockwise by θ around the z axis, where θ is the angle between a fixed image and the current image on which the feature extraction process is being performed. Additionally, instead of using a manual approach to register the two images, we use feature-based techniques to

automate the registration process. Here, the MSER (Maximally Stable Extremal Regions) is used to extract the feature descriptors. As MSER performs well on images containing homogeneous regions with distinctive boundaries [50], in order to extract feature descriptors, we need to make an image with brighter regions. To address this issue, we calculate the new image by adding the original images to a coefficient of their gradient magnitude. Then we extract feature descriptors in both fixed and rotated images using MSER. After matching features by using their descriptors and retrieving locations of the corresponding points for the images, we find a transformation corresponding to the matching point pairs using the statistically robust M-estimator SAmple Consensus (MSAC) algorithm, which is a form of the RAndom SAmple Consensus (RANSAC) algorithm [51]. Finally, we use the geometric transform to recover the scale and find the angle between the fixed and rotated images. More information on how to calculate the normal scale and the angle of rotation is provided in [52].

5-4 PRECIPICE BORDER DETECTION ALGORITHM (PBDA)

In this part, we explain our proposed *precipice border* detection on the coin surface, which can be extended for any height-map images. Instead of the normal edge detection in 2D approaches, a new 3D bordering concept that we name it a *precipice border* is proposed for the height-map images. In two-dimensional edge detection, an edge is a line separating two segments or regions while in our proposed border detection, a *precipice border* separates two segments leading to a set of attributes. It helps us to extract effective features based on the normal vectors and the areas of the precipice around the letters, numbers, and shapes.

5-4-1 Proposed Method for Detecting *Precipice Borders*

In this subsection, we propose the *precipice borders* detection method for height-map images using a triangulating method and fuzzy clustering. To detect the *precipice borders* of the height-map image, several features are used for clustering the triangles, and one of these clusters will be the *precipice borders* cluster. In order to use triangles for clustering, we consider an array that contains the information about all triangles of an image and name it, triangle array. The triangle array contains rows, columns of triangle samples as well as their vertex coordinates. Figure 5.2 (a) shows the triangulation gridding on the image matrix applied to the coin image. As in the triangulation gridding, the gray triangles cover all pixels of the image, the white triangles are disregarded with the aim of reducing the processing time. Accordingly, in a height-map image with n rows and m columns, the triangle array includes $(n-1)$ rows and $(m-1)$ columns. In the clustering process,

we have a 5-dimension feature space consisting of three entries of the normal vector of the triangle, triangle area, and the angle between the triangle sample and its surrounding larger triangle. In many places on the coin triangulated image, it is difficult to define sharp boundaries where the features of the triangles change gradually from one to another. Therefore, we take advantage of fuzzy clustering to solve this problem. In this step, fuzzy c-means clustering [22] is applied to cluster the triangles with several features based on the normal vector and the area of the triangle. The normal vector of a triangle can be calculated by:

$$\vec{N} = (\vec{v_2} - \vec{v_1}) \times (\vec{v_3} - \vec{v_1}) \tag{5.1}$$

where v_1, v_2, and v_3 are vertices of the triangle. The area of a triangle is also calculated by Heron's formula:

$$A = \sqrt{s(s-a)(s-b)(s-c)} \; , \; s = \frac{(a+b+c)}{2} \tag{5.2}$$

where $a = |(\vec{v_2} - \vec{v_1})|$, $b = |(\vec{v_3} - \vec{v_1})|$, and $c = |(\vec{v_2} - \vec{v_3})|$. To improve the clustering process and ignore the very tiny scratches and noises on the coin surface, we also consider a larger triangle for each sample whose vertices are the neighbors of the triangle sample. Accordingly, there may be an angle between these two triangles. As mentioned earlier, this angle is another feature for each triangle sample through the clustering process. Therefore, each triangle sample has five features in the clustering process; N_x, N_y, N_z, A, and α, where $\vec{N} = (N_x, N_y, N_z)$ is the normal vector of the triangle sample, A is the area of the triangle sample, and α is the angle between the larger triangle and the triangle sample. Also, α can be computed by:

$$\alpha = \cos^{-1}\left(\frac{|\vec{N_s}.\vec{N_b}|}{\|\vec{N_s}\| \times \|\vec{N_b}\|}\right) \tag{5.3}$$

where N_s is the normal vector of the triangle sample, N_b is the normal vector of the larger triangle, and $\|\vec{N}\|$ is the magnitude of the normal vectors. The number of clusters is equal to 3. Since the clusters are very different in the mean value of their matrices, we can easily select the cluster containing *precipice borders*.

Figure 5.2 (b) illustrates the larger triangle, which covers a neighborhood of the triangle sample. A large α for any triangle sample demonstrates that this triangle is not following the pattern of its neighborhood and probably is a noise or a very tiny scratch. The result of PBDA is similar to a

Gabor Filter bank with different orientations. To see the outstanding result of the proposed method comparing with the Gabor filter, we provide an example. Figure 5.3 (a) shows a reconstruction of a Gabor filter bank of 8 different directions on a genuine twenty Kroner 1996 height-map image and Figure 5.3 (b) illustrates the result of PBDA for the image in a 2D view. There was no preprocessing on the coin images in these two methods for extracting the borders. As can be seen in these figures, the proposed border detection method is able to provide more precise information about characters and shapes than the Gabor filter.

(a) (b)
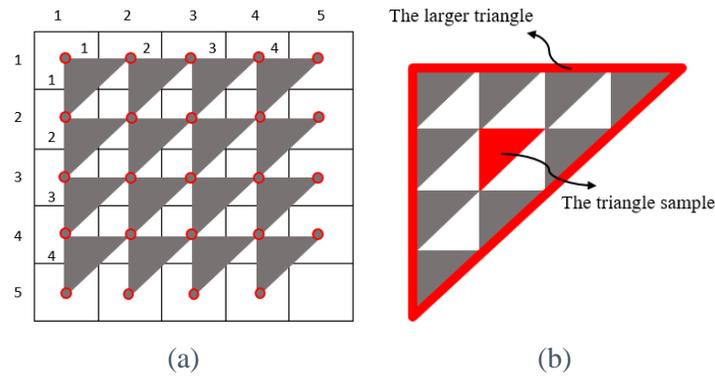
Figure 5.2. (a) Triangulation of the height-map image; the triangle array on the image matrix, and (b) the larger triangle which is considered for the fifth feature of the triangle sample.

(a) (b)

Figure 5.3. (a) Reconstruction of a Gabor filter bank on a genuine twenty Kroner 1996 image with wavelength=5 and 8 directions= {0,22.5,45,67.5,90,112.5,135,157.5} and (b) Precipice borders of the image resulted from PBDA.

In Figure 5.4 (a), which is a three-dimensional view of the coins after the *precipice border* detection process, the significant variations between the fake, in the right and genuine coin, in the left of the figure, it is clear, although they are degraded. Some of these differences were marked with arrows where the expected *precipice borders* at the same place on two coins are totally different and are worthwhile to be analyzed. Figure 5.4 (b) and Figure 5.4 (c) show the mesh grid of the *precipice borders* and low-slope surfaces for the genuine and fake coins, respectively.

As signified by the arrows in these figures, the immense scale of the *precipice borders* on the fake coin was more significant than the genuine counterpart. Also, several regions on the genuine coin with low slope surfaces have not been detected as the *precipice borders*, unlike the high-altitude borders detected in the same regions on the counterfeit one and vice versa. Since the *precipice borders* are separated from the texture, they can simply give us the correct information about the height of specific letters, numbers or shapes. Given that we want to analyze the *precipice borders* and PBDA results in precise values of heights, we no longer need to restore degraded images of shiny coins.

5-4-2 Binned Borders in Spherical Coordinates (BBSC)

In this subsection, we propose Binned Borders in Spherical Coordinates (BBSC) to analyze different parts of those *precipice borders* which are normally folded or curved. In spherical coordinates, any point on a surface has a normal vector equal to a counterpart normal vector on the sphere. BBSC is proposed to categorize the triangles which build surfaces on an object, using their normal vectors. In fact, a histogram of normal vectors on the sphere is built with several bins. Thus, each triangle on any object in a 3D triangulated image is assigned to a Bin. The mapping to the Bins can easily be computed by considering the normal vector of the triangle and the polar and azimuthal angles.

In this system, each BBSC has five parameters η, ρ, δ, θ, and φ illustrating the number of sub-divisions based on the polar angle, number of sub-divisions based on the azimuthal angle, $\delta \in \{1,2,3, \ldots, n_B\}$ polar angles in maximum range $(0, 2\pi)$, and azimuthal angles in maximum range $(0, 2\pi)$ respectively. From now on, a Bin in this concept is a set of triangles with similar normal vectors. The number of Bins, n_B, in $BBSC_{\eta,\rho}^{\delta,\theta,\varphi}$ system is equal to $\eta \times \rho$. Figure 5.5 (a) shows a $BBSC_{4,6}^{\delta,\theta,\varphi}$ with 24 Bins where $\theta = (0,2\pi)$ and a triangle which was assigned to Bin_2. Since the value of the objects like shapes, letters, and numbers on the coin image are positive in the *z* axis, we consider the positive part of the sphere in the *z* axis. As we apply the *precipice borders* of the coin surface for extracting features, we consider a histogram of normal vectors, in which the histogram bins cover only a small part of the hemisphere.
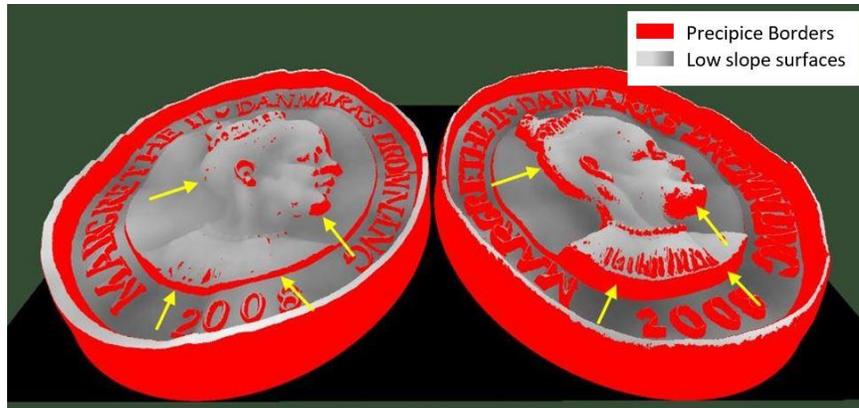
(a)
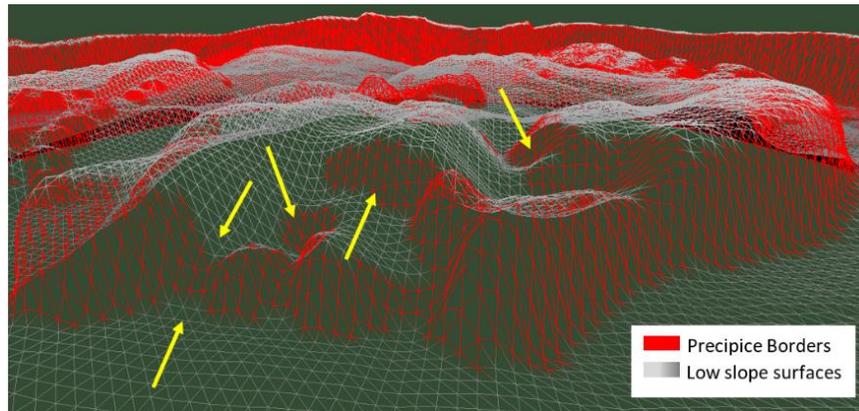
(b)
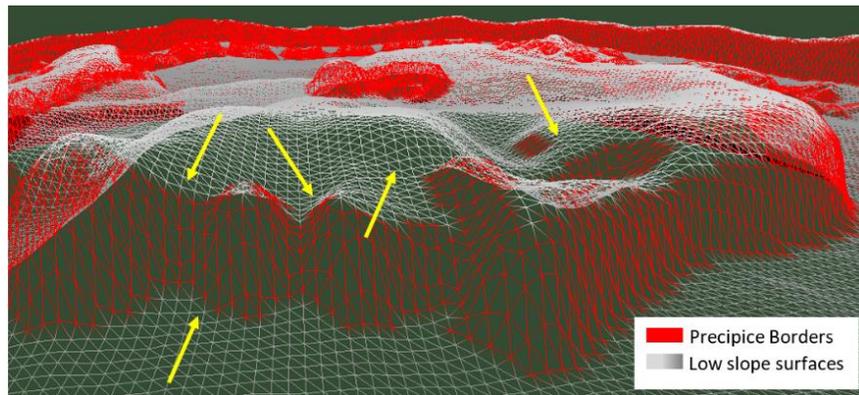
(c)

Figure 5.4. Significant differences between the *precipice borders*: (a) a genuine coin 20 kroner 2008 on the left side and the fake counterpart on the right side in a 3D view, (b) the *precipice borders* of a specific region on a genuine Danish 2008 coin, and (c) the *precipice borders* of the same region in (b) on a fake Danish 2008 coin.

The distribution of normal vectors for the *precipice borders* detected by PBDA is shown in Figure 5.5 (b). Therefore, we adjust $\theta = \left(\frac{12\pi}{26}, \frac{\pi}{2}\right)$ and $\varphi = (0, 2\pi)$ in polar and azimuthal angles to reach

the desired coin BBSC. Figure 5.5 (c) shows the Bins of the small part of the sphere which is used for the desired coin BBSC. Given that $BBSC_{2,8}^{\delta,\theta,\varphi}$ is the *precipice borders* set of the image which are classified as Bin_δ where $\delta \in \{1,2,3, ..., 16\}$, we have $BBSC_{2,8}^{\delta_1,\theta_1,\varphi_1}$ with $\theta_1 = \left(\frac{12\pi}{26},\frac{25\pi}{52}\right)$ and $\delta_1 = \{1,2, ...,8\}$, and $BBSC_{2,8}^{\delta_2,\theta_2,\varphi_2}$ with $\theta_2 = \left(\frac{25\pi}{52},\frac{\pi}{2}\right)$ and $\delta_2 = \{9,10, ..., 16\}$ emphasizing that $\varphi_1 = \varphi_2 = \varphi$.

(a) (b) (c)
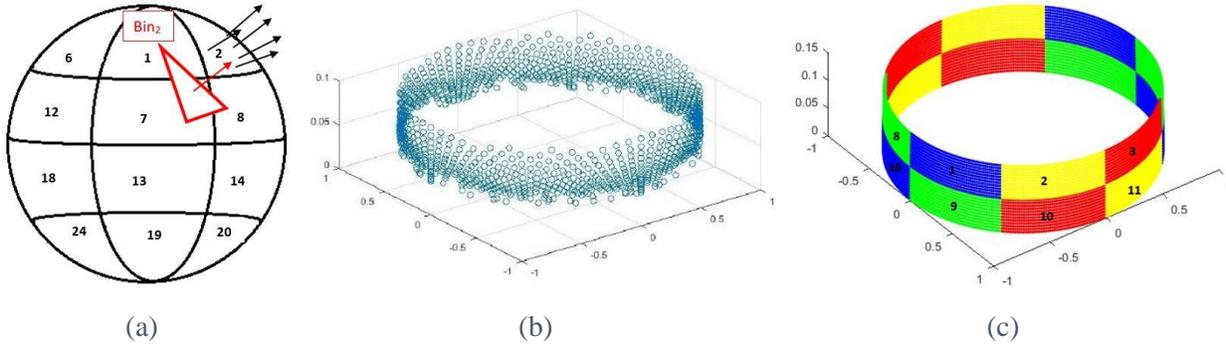
Figure 5.5. (a) Assigning a triangle to a specific Bin in $BBSC_{4,6}^{\delta,\theta,\varphi}$, (b) distribution of the normal vectors on the coin *precipice borders*, and (c) Desired binning for coin $BBSC_{2,8}^{\delta,\theta,\varphi}$.

Figure 5.6 shows the 2 Bin images of 16 Bin images of a Danish 20 kroner 1990 coin, for instance. Figure 5.6 (a) illustrates $BBSC_{2,8}^{\delta,\theta,\varphi}$, the *precipice borders* when $\delta = \delta_2$, $\theta = \theta_2$, and $\varphi = \varphi_2$ and Figure 5.6 (b) demonstrates the *precipice borders* when $\delta = \delta_1$, $\theta = \theta_1$, and $\varphi = \varphi_1$. In the next step, all the 16 sets of the $BBSC_{2,8}^{\delta,\theta,\varphi}$ will be used for feature extraction separately.

5-4-3 Feature Extraction and Eliminating Sparse Features

As mentioned earlier, the *precipice borders* in a 3D image consist of much useful information for feature extraction. In several methods for coin recognition, researchers used Local Binary Patterns (LBP) and Region Binary Patterns and divided a coin image into several rings and extracted LBPs. For example, in Region Binary Pattern, histograms or gradient magnitudes were calculated from each region and were used as a final descriptor. In this paper, however, we utilize the area of the triangles provided by the *precipice borders* of the coin surface instead of histograms or gradient magnitude of the color scale. In order to extract features from the *precipice borders* of the coin image, we combine the region pattern approach with the coin $BBSC_{2,8}^{\delta,\theta,\varphi}$ proposed in subsection 5.4.1. Using the coin BBSC allows us to include the normal vector of the triangles as well, which

is indispensable to extract the features from the *precipice borders* for detecting counterfeit coins. This is inspired by the act of coin experts who rotate and look at the coins with different azimuthal and polar angles to find dissimilarities.

<div align="center">(a)   (b)</div>
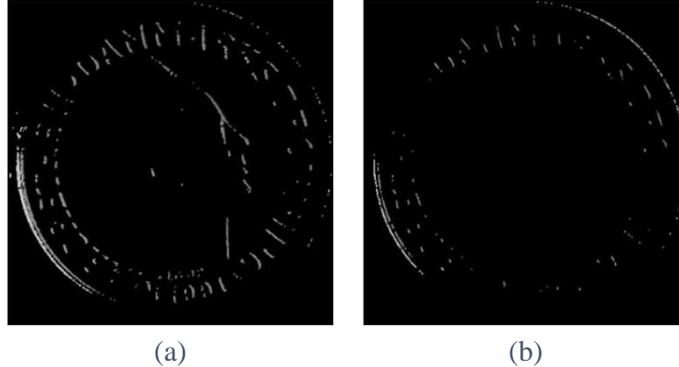
Figure 5.6. Examples of the *precipice borders* for a 20 Kroner 1991 coin: (a) $BBSC_{2,8}^{3,\theta,\varphi}$ where $\theta = \theta_2$ and (b) $BBSC_{2,8}^{11,\theta,\varphi}$ where $\theta = \theta_1$.

Using Information Gain for ranking the features extracted from BBSC illustrated that the discriminating capability of each feature was markedly different after concatenating the feature-sets of the Bins compared to when we consider each Bin feature-set separately. Hence, without concatenation, instead of a feature vector, we will have a set of vectors representing the coin considering all Bins of the coin BBSC. It means that each coin is represented by a feature matrix containing r rows and c columns where r is equal to the number of the Bins (n_B) and c is equal to the number of regions ($n \times m$) on the coin image. However, we will compare these two collecting feature approaches in Section 5.5. To extract features from each Bin, the Bin image is divided into n concentric rings, and each ring consists of m sub-regions. Then the sum of the area of triangles from each sub-region is added to the feature-set. Figure 5.7 (a) shows an example for 3 rings and 8 sub-regions then we have 24 small regions for any Bin image of the coin BBSC. Figure 5.7 (b) and (c) show the 24 small regions for Bin_3 and Bin_{11} of the coin BBSC respectively. After the feature extraction process, we have the feature matrices representing the coins, which are extracted from the related BBSC banks. As can be seen in Figure 5.7 (b) and Figure 5.7 (c), some of the features have information while some of them do not. Here, sparse features are those features which have frequently zero values. Therefore, the sparse features should be taken out of the feature-set. Eliminating the spare features is applied to each row of the matrix separately because each row comes from different Bin images of a coin BBCS bank and has a different meaning. It means that a specific row of the matrices has its own feature-set.
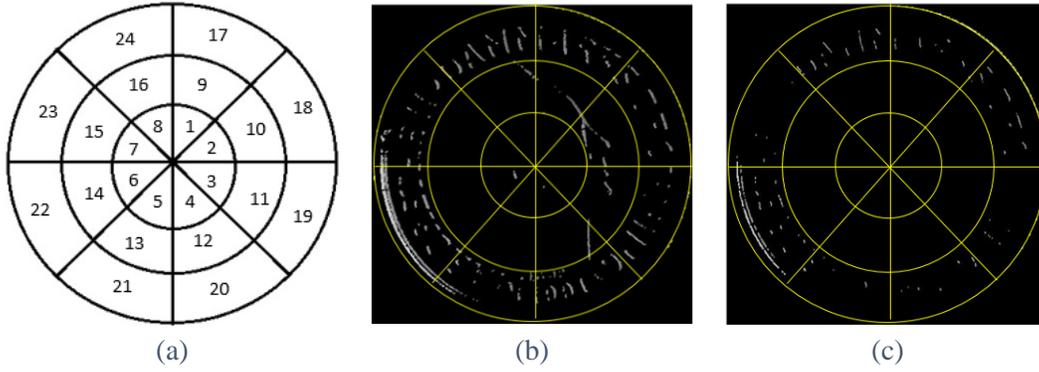
Figure 5.7. (a) Concentric rings and sub-regions structure, (b) the structure for $BBSC_{2,8}^{3,\theta,\varphi}$, and (c) the structure for $BBSC_{2,8}^{11,\theta,\varphi}$.

5-5 DETECTING COUNTERFEIT COINS USING AN ENSEMBLE CLASSIFIER

In this section, we explain how to apply an ensemble learning method in the proposed method. As mentioned in subsection 5.4.2, the proposed feature extraction method represents a coin by a matrix whose records are the set of features extracted from the various Bins of coin BBSC. It is clear that an individual classifier is not able to get a matrix as an input to build a model. To do this, we can easily concatenate these features to form a vector. However, we also take advantage of stacking classification to keep the nature of the matrix for the classification step. Regarding the characteristics of stacking classification, it is possible to train the classification module with the feature matrices if we change it slightly. In order to achieve an ensemble classifier, we consider each row of the coin matrix separately so that we will have $p = n_B$ datasets each of which contains all training samples with a different set of features and values. In the stacked generalization, a combination of several heterogeneous learning algorithms is applied to the same data set that should be divided into several sub-datasets [53]. However, this is not necessary for our ensemble classification algorithm thanks to having p different datasets for the same samples (not the same set of features).

Regarding reference [25], for improving stacking classification, the use of output class probabilities instead of class predictions is indispensable for the first-level classifiers. To this end, we may use several classifiers that are naturally probabilistic or can be turned into probabilistic. A probabilistic classifier generalizes the concept of conditional distributions instead of functions. For a given $x \in X$, $P(X| Y)$ signifies that probabilities are assigned to all $y \in Y$. The final class can then be predicted by this formula:

$$\hat{y} = \arg max_y \, P(Y = y| X) \qquad\qquad (5.4)$$

Generally speaking, the class with the highest probability is selected as the predicted class [54]. According to this, some classification models, such as Naive Bayes, Logistic Regression and Multi-Layer Perceptron with an appropriate loss function, are naturally probabilistic. Albeit some other models, such as support vector machines, are not naturally probabilistic, there exist some methods to convert them into probabilistic classifiers [26]. In reference [25], a formula for estimating output class probabilities has been provided. In this research, however, we apply to stack classification with a proposed reject option. A classification system with a reject option will ignore a case when making a decision is difficult. On account of the significant similarity between fake and genuine coins, in this research, the classifier with the proposed reject option is able to reject uncertain cases. Hence, in the first level classification, regarding reference [55] example $x \in D_k$ is classified as fake or genuine only if the probability that x belongs to $C_i \in \{fake, \, genuine\}$ is higher than or equal to a threshold t ($t = \,0.5$ is selected here):

$$h_k(x) = \begin{cases} argmax_{C_i}\big(p(C_i|x)\big) & if \; max_{C_i}(p(C_i|x)) \geq t \\ reject & if \; \big(p(C_i|x)\big) < t \; \forall i \end{cases} \qquad (5.5)$$

Regarding Figure 5.8, by p first-level classifiers, we construct a new dataset that contains $\{x_i', \, C_i\}$ where $x_i' = \{h_1(x_{i,1}), \, h_2(x_{i,2}), \; ..., \, h_p(x_{i,p})\}$. To provide the reject option in order to increase the reliability of the prediction, the new sample x_i' will remain in the train or test set if $max(F, \, G) > min(F, \, G) + R$ and the sample will be rejected if $max(F, \, G) \leq min(F, \, G) + R$ where F, G, and R are the number of predicting "fake", "genuine", and "rejected" of x_i' in the first-level classification, respectively. The performance of different strategies for selecting first-level and second-level classifiers will be discussed in Section 5.6.

Algorithm 5-1. Stacked generalization in the classification task algorithm with reject option used in the proposed counterfeit coin detection

Input: Training data $D_k = \left\{ \{x_{i,k}, c_i\}_{i=1}^{m} \right\}_{k=1}^{p}$ $(x_{i,k} \in \mathbb{R}^n, y_i \in C)$, where k is a Bin and p is the number of Bins in BBSC.

Output: An ensemble classifier H

1: Step 1: Learn first-level classifiers
2: **for** $k \leftarrow 1$ to p
3:     Learn a base classifier h_k using D_k
4: **end**
5: Step 2: Construct new datasets from D_k
6: **for** $i \leftarrow 1$ to m
7: Construct a new data set that contains $\{x_i', c_i\}$, where $x_i' = \{h_1(x_{i,1}), h_2(x_{i,2}), \dots, h_p(x_{i,p})\}$
8: **end**
9: Step 3: Learn a second-level classifier
10: Learn a new classifier h' based on the newly constructed data set
11: **return** $H(x) = h'(h_1(x), h_2(x), \dots, h_p(x))$

Figure 5.8. Stacked generalization in the Classification task algorithm with reject option used in the proposed counterfeit coin detection.

5-6 EXPERIMENTAL RESULTS

We conducted experiments to evaluate the performance of the proposed method for counterfeit coin detection. Several types of coins were used, and we applied a precise 3D scanner to scan a large number of Danish and Chinese coins. In addition, the effect of the parameters involved in the proposed method is rigorously examined. In order to demonstrate the impact of the proposed ensemble method, we compare it with the recently related published methods. Moreover, we compare the effectiveness of different classifiers on the stack classification task which is an important part of the proposed method. The hardware of the test environment consisted of an i7-4500U 4.2 GHz CPU (only one core was used), DDR3 6 GB RAM, the operating system used was Windows 8.1- 64 bit and the programming environment was MATLAB 2015 and Visual C++.Net 2015.

5-6-1 Datasets

In this research, we created six types of height-map image datasets by a 3D scanner to train and evaluate the performance of our proposed ensemble system. Four different years of 20 Kroner 1990, 1991, 1996, and 2008 and two different Chinese coins: half Yuan 1942 and one Yuan 1997 have been selected. The first reason for selecting these specific coins is to have enough fake samples of these types of coins provided by the Law Enforcement office as having fake coins is a crucial challenge for this research. The second reason for testing our method on the mentioned

datasets is that in these datasets, counterfeit samples are very well-forged as far as coin experts cannot detect them easily. The 3D scanner that was used for capturing the coins was a powerful scanning device named IBIS TRAX patented by Ultra Electronics Forensic Technology Ltd. Co in Montreal, Canada. The machine has a built-in microscope and a 5-group of adjustable LEDs are configured to view the object from different angles. The precision of the scanner is about 6 microns resulting in a *3550 × 3550* image resolution (with the gray level of 0 to 255 for JPG and 0 to 65535 for JPEG2000 images). As mentioned earlier in Section 3, in this research, we used the JPG images and resized them to *400 × 400*. Table 5.1 demonstrates the number of coin images for each dataset. In order to train and evaluate the system, each dataset is stochastically split into training (60%), validation (20%), and test (20%) sets. Apart from using the validation set to avoid over-fitting, the parameters used in the proposed method are also tuned by comparing the results on the validation set. The configuration with the lowest EER on the validation set is applied to the test set accordingly. Figure 5.9 shows the examples of the height-map images for the six different coin datasets resulted from 3D scanning.

Table 5.1. The number of coin images for Train, Validation, and Test sets in this research.

Dataset	Training set		Validation set		Test set	
	Genuine	Fake	Genuine	Fake	Genuine	Fake
20 Kroner 1990	1731	1480	577	493	577	493
20 Kroner 1991	1800	1740	600	580	600	580
20 Kroner 1996	1287	1356	430	452	430	452
20 Kroner 2008	2400	2142	800	714	800	714
Half Yuan Chinese 1942	261	360	87	120	87	120
One Yuan Chinese 1997	345	426	115	142	115	142
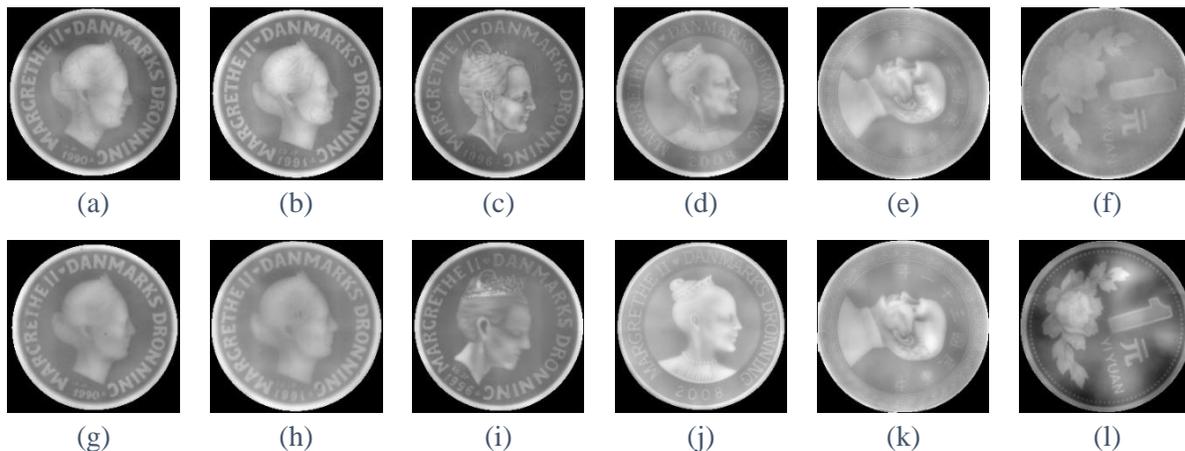
Figure 5.9. Examples of genuine and fake Danish and Chinese coin images: (a) Danish 1990, (b) Danish 1991, (c) Danish 1996, (d) Danish 2008, (e) Chinese 1942, (f) Chinese 1997; and (g), (h), (i), (j), (k), and (l) are the fake counterparts of the coins respectively.

5-6-2 Experimental Setup and Results

In this subsection, we describe the setup for tuning the parameters used in the explained algorithms. In the *precipice borders* detection algorithm using fuzzy clustering, the threshold τ on membership degrees is set equal to 0.78 manually. In more important setups, the proposed feature extraction and the coin BBSC are analyzed to determine the best parameters $\eta, \rho, m,$ and n for extracting features based on *precipice borders* detected beforehand. Firstly, we implemented the coin BBSC with a variable number of sub-regions η and ρ explained in subsection 5.4.2 which had 1 to 4 sub-divisions η with 8, 16, and 36 sub-divisions ρ. In this experiment, as Table 5.2 illustrates, we used two criteria (number of Zero-Matrix images in coin BBSC and the average of coin BBSC time) to find the best value for η and ρ. Regarding Table 5.2, by increasing the number of sub-divisions η, some of the coin BBSC images which are supposed to contain some information about *precipice borders* will be the Zero-Matrix images whose pixels are all black or zero. On the other hand, we want to take advantage of the coin BBSC which should have at least two sub-divisions η. In Figure 5.10 (a) to Figure 5.10 (d), the accuracy of the proposed method with respect to different values of η and ρ are shown for the four Danish coin datasets 20 Kroner 1990, 1991, 1996, and 2008. Therefore, considering $\rho \in \{8,16,32\}$ and the criteria, the best selection was $\eta=2$ and $\rho=8$. In addition, we implemented the feature extraction module with different Bins that had different values for n (the number of sub-regions) between 8 and 26 with 2 increments and 4 to 24 for m (number of rings) with 4 increments. Table 5.3 shows the accuracies according to these templates for the test data of Danish 1990 which is the most important case study in our research. Regarding Table 5.3, the accuracies are best when $m = 12$. The best result obtained in this experiment was 98.6% accuracy when $n = 20$ and $m = 12$. The total number of features extracted by the proposed method is 3840 which is calculated by $m \times n \times n_B$. The size of the selected feature-set after eliminating sparse features is 1311, which is about three times smaller than the whole feature-set after feature extraction. The number of features extracted from each Bin is illustrated in Figure 5.11. It is clear in this image that the size of the feature-set extracted from coin BBSC when $\delta = \delta_2$ and $\theta = \theta_2$ is almost two times larger compared to when $\delta = \delta_1$ and $\theta = \theta_1$. The accuracies mentioned in Table 5.3 and Figure 5.10 have resulted from the efficient ensemble classifier on the validation set, which is discussed in the next subsection.

Table 5.2. Comparison of coin BBSCs with different configurations for Danish dataset 1990.

η	ρ	# of Bins (= $\eta \times \rho$)	Avg. of Coin BBSC time (sec)	# of Zero-Matrix in coin BBSC
1	8	8	0.098	0
1	16	16	0.113	0
1	32	32	0.125	0
2	8	16	0.117	0
2	16	32	0.129	1
2	32	64	0.172	2
3	8	24	0.137	8
3	16	48	0.164	16
3	32	96	0.201	34
4	8	32	0.155	16
4	16	64	0.190	33
4	32	128	0.268	69

Table 5.3. Tuning the number of the sub-regions and the rings in the feature extraction process on dataset Danish 1990.

n (# of sub-regions)	m (# of rings)					
	4	8	**12**	16	20	24
8	72.4	85.6	88.8	84.8	79.2	72.8
10	74.9	90.3	94.3	94.1	82.4	78.5
12	78.5	90.8	94.7	95.4	90.2	80.3
14	82.2	91.2	94.9	93.2	92.0	82.5
16	84.6	90.4	95.5	90.8	88.4	84.2
18	80.1	88.9	96.2	90.6	88.7	78.8
20	86.3	88.2	**98.4**	87.4	86.3	78.1
22	88.2	86.3	94.6	86.4	86.7	76.1
24	82.3	80.9	92.6	87.8	82.1	78.6
26	78.3	80.9	88.2	84.1	80.6	73.7

(Accuracy (%) is the left-side vertical label spanning the data columns.)

(a)

(b)

(c)

(d)
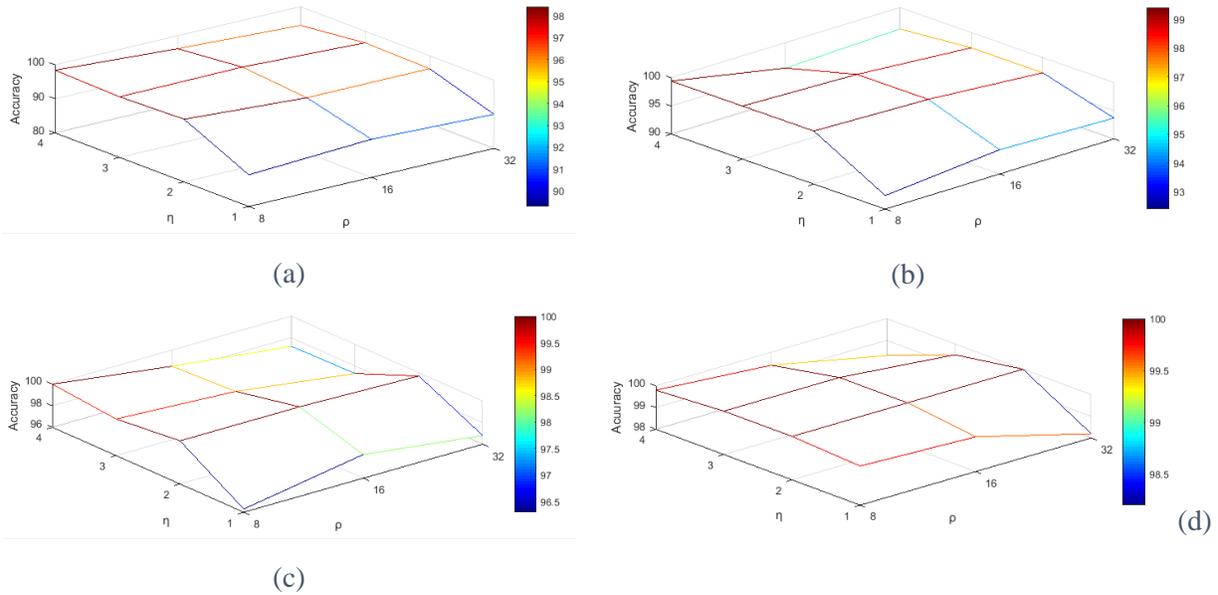
Figure 5.10. Accuracy of the proposed method with different values of η and ρ on datasets: (a) 20 Kroner 1990, (b) 20 Kroner 1991, (c) 20 Kroner 1996, and (d) 20 Kroner 2008.

5-6-3 Efficient Ensemble Classifier

Before applying ensemble classification, we used six state-of-the-art classifiers to train and test the method (Table 5.4). To do this, we concatenated each feature matrix of the coins to a single vector and used it as the input of the classifiers. To a fairer comparison of the impact of the concatenated feature vectors against feature matrices, we also applied an ensemble classification proposed in [25] and trained it by the concatenated features. Since the class fake and genuine in the datasets are not balanced well, comparing precision and recall as well as the rejection rate and the detection rate can be reasonable. Table 5.4 shows the result of the counterfeit coin detection by seven classifiers. The best detection rate 91.3% was about SGD (Stochastic Gradient Descent) while the lowest rejection rate 1.5% was about MLP (Multi-Layer Perceptron). Also, the accuracy reported for the ensemble classifier trained by this feature-set was by far lower than SGD.
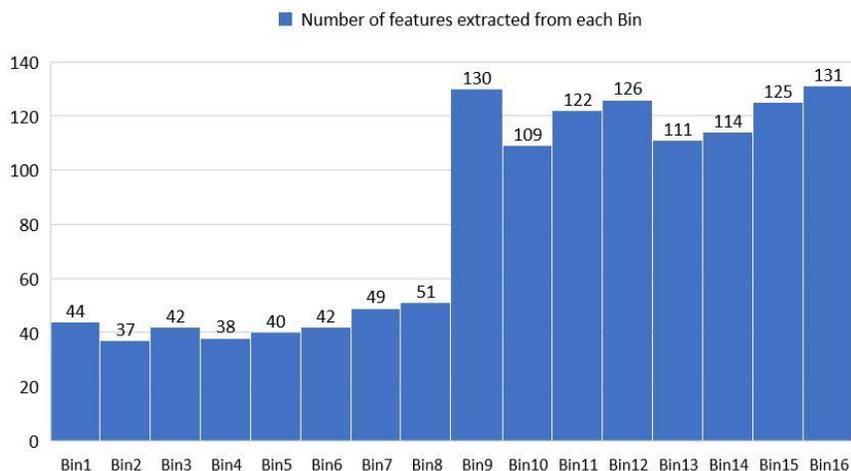
Figure 5.11. Number of features for each Bin for 20 Kroner 1990 coin.

Table 5.4. Result of classification by seven state-of-the-art classifiers for dataset 20 Kroner 1990.

Classifier	Precision	Recall	Rejection Rate (%)	Detection Rate (%)
MLP	0.902	0.888	**1.5**	87.5
KNN	0.864	0.833	3.0	80.8
SGD	**0.941**	**0.936**	2.5	**91.3**
Random Forest	0.824	0.840	3.2	81.3
SVM	0.908	0.896	1.8	88.0
MLR	0.842	0.848	2.0	83.1
Ensemble [25]	0.847	0.856	3.6	82.5

To demonstrate the discriminating capability of the feature matrices over the concatenated features, we apply the ensemble classifier designed in Section 5.5 and improve the performance of

the system. To find the efficient classifiers, first, we implemented the stack generalization with the same first-level classifiers and the same meta-classifier.

In this comparison, as it is clear in Table 5.5, SGD and MLR (Multiple Linear Regressions) performed better than other classifiers. Then we randomly made five groups of classifiers from the list available in Table 5.5 and used their predicted classes as inputs for SGD (with loss function: Hinge loss (SVM), learning rate: 0.01, epoch: 500, and lambda: 1.0E−4) and MLR classifiers as meta-classifiers in two separate examinations. Group 1 is the combination of eight MLR classifiers trained by the bank of $BBSC_{2,8}^{\delta_1,\theta_1,\varphi_1}$ and eight $C_{4.5}$ [56] classifiers trained by the bank of $BBSC_{2,8}^{\delta_2,\theta_2,\varphi_2}$. Group 2 is the combination of eight SVM classifiers trained by the bank of $BBSC_{2,8}^{\delta_1,\theta_1,\varphi_1}$ and eight IB1 [57] classifiers trained by the bank of $BBSC_{2,8}^{\delta_2,\theta_2,\varphi_2}$. Group 3 is a combination of eight KNN classifiers trained by the bank of $BBSC_{2,8}^{\delta_1,\theta_1,\varphi_1}$ and eight IB1 classifiers trained by the bank of $BBSC_{2,8}^{\delta_2,\theta_2,\varphi_2}$. Group 4 is a combination of eight KNN classifiers trained by the bank of $BBSC_{2,8}^{\delta_1,\theta_1,\varphi_1}$ and eight IB1 classifiers trained by the bank of $BBSC_{2,8}^{\delta_2,\theta_2,\varphi_2}$. Group 5 is a combination of eight KNN classifiers trained by the bank of $BBSC_{2,8}^{\delta_1,\theta_1,\varphi_1}$ and eight MLP classifiers trained by the bank of $BBSC_{2,8}^{\delta_2,\theta_2,\varphi_2}$. As can be seen in Table 5.6, the combination of Group 2 with both SGD and MLR produced remarkable results in terms of precision, recall, and RMSE while SGD outperformed MLR as a meta classifier. Therefore, we consider the combination of Group 2 and SGD as the efficient ensemble classifier.

Table 5.5. Performance of counterfeit detection by same classifiers for the first-level and the meta-classifier on dataset 20 Kroner 1990

16× First-level Classifiers	Meta-Classifier	Precision	Recall	Rejection Rate (%)	Accuracy (%)
MLP	MLP	0.941	0.936	0.8	92.8
KNN	KNN	0.946	0.933	1.0	92.4
SGD	**SGD**	**0.988**	**0.988**	1.1	**97.7**
Random Forest	Random Forest	0.968	0.961	1.0	95.1
Decision Tree $C_{4.5}$	Decision Tree $C_{4.5}$	0.930	0.923	1.4	91.0
Logistic Regression	Logistic Regression	0.966	0.965	1.4	95.1
Naïve Bayes	Naïve Bayes	0.899	0.896	1.0	88.7
KStar	KStar	0.940	0.941	1.6	98.4
SVM	SVM	0.944	0.944	0.8	93.6
IB1	IB1	0.946	0.933	0.9	92.5
MLR	MLR	0.982	0.980	**0.7**	97.3

Table 5.6. Performance of counterfeit detection by several groups of classifiers for the first-level and the meta-classifier on dataset 20 Kroner 1990

First-level Classifiers	Meta-Classifier	Precision	Recall	RMSE
Group 1	SGD	0.972	0.968	0.1782
Group 1	MLR	0.970	0.968	0.1810
Group 2	**SGD**	**0.993**	**0.992**	**0.0867**
Group 2	MLR	0.992	0.992	0.0891
Group 3	SGD	0.971	0.970	0.1658
Group 3	MLR	0.970	0.969	0.1810
Group 4	SGD	0.986	0.986	0.0935
Group 4	MLR	0.986	0.986	0.1003
Group 5	SGD	0.979	0.979	0.1448
Group 5	MLR	0.973	0.972	0.1454

In order to perceive the impact of *precipice border* analysis on extracting effective features, we compared the proposed BBSC with Gabor filter bank with different wavelength=$\{3, 5, 10, 15\}$. As in the proposed method, the number of Bins in azimuthal angle was 8, we consider 8 directions=$\{0, 22.5, 45, 67.5, 90, 112.5, 135, 157.5\}$ to build the Gabor filter bank. Then, instead of the area of triangles, we calculated the number of non-zero pixels in each sub-region for feature extraction (we display it here with the star sign). Another comparison for the feature extraction step is to use Circular Shift of Gabor coefficients (CSGabor) for feature representation which is presented in [10]. We also performed feature selection by eliminating sparse features on these three methods. As can be observed in the results of the accuracy listed in Table 5.7 for these three methods, the proposed BBSC method has a significant impact on the feature extraction process rather than using Gabor filter-based method. Although the feature dimension for CSGabor was 1177 and smaller than the proposed method, the feature extraction time was a bit slower. The feature extraction in method (*) was faster, although the feature dimension was 1706 and the highest.

Table 5.7. Comparison of two feature extraction methods based on Gabor filter with the proposed BBSC based feature extraction in terms of accuracy of the efficient ensemble classifier on the height-map datasets

Method	Danish 1990	Danish 1991	Danish 1996	Danish 2008	Chinese 1942	Chinese 1997
Proposed BBSC based	**98.6**	**98.0**	**99.8**	**99.9**	**95.5**	**92.2**
Gabor filter bank (*)	92.6	91.0	95.5	95.7	87.6	78.5
CSGabor [10]	90.2	76.8	92.2	88.1	64.9	70.7

In Table 5.8, we report the performance of the proposed method on the six datasets separately. As expected, the highest accuracies in counterfeit coin detection were about Danish 1996, and 2008

coins. While Precision, TP, and FP were 1.0, 1.0, and 0.0 respectively for both datasets, the accuracy for Danish 1996, and Danish 2008 were 99.8, 99.9 respectively because of a very low rejection rate in both tests. For the Chinese coins, we got acceptable results as we had a smaller dataset than Danish datasets. Also, the rejection rates for Chinese data sets, in this experiment, were by far higher than other data sets. The results for precision, recall, TP, and FP are weighted averages of the two classes. As mentioned previously, the accuracies for both 20 Kroner 1996 and 2008 were dramatically high while we got 98.6 and 98.0 accuracies for 20 kroner 1990 and 1991 respectively. The detection failures for Danish 1990 and 1991 are caused by the significant similarities between the genuine coins and their fake counterparts. The *precipice borders* detected in 2008 and 1996 for the fake coins are dramatically thicker than the genuine counterparts. Hence, there would be a big difference between the coin BBSCs of genuine and fake coins. In this case, the coin BBSC approach is more effective on the feature extraction and makes the features more separable insofar as the number of rings and sub-regions is no longer very important. By contrast, the number of rings and sub-regions should be optimized if there is no significant variation in the thickness of the details on the coins. In such circumstances, there should normally be significant differences in *precipice border* orientations which are considered by rings and sub-regions in feature extraction.

Table 5.8. Performance of the proposed method for all available datasets.

Datasets	Precision	TP Rate	FP Rate	Rejected-fake (%)	Rejected-genuine (%)	Accuracy (%)
20 Kroner 1990	0.993	0.992	0.007	0.4	0.2	98.6
20 Kroner 1991	0.984	0.984	0.034	0.1	0.3	98.0
20 Kroner 1996	1.000	1.000	0.000	0.15	0.0	99.8
20 Kroner 2008	1.000	1.000	0.000	0.1	0.0	99.9
Half Yuan Chinese 1942	0.973	0.972	0.025	1.7	0.0	95.5
One Yuan Chinese 1997	0.935	0.929	0.262	0.7	0.0	92.2

5-6-4 Comparison with Previous Methods

We compared the proposed method with five recent works published in the field of counterfeit coin detection and coin recognition, noting that the data used for this comparison was exactly the same as we used to train and evaluate our proposed method. First, we calculated the accuracy of the models trained by the six datasets and compared them.

Table 5.9. Comparison of previous research in terms of accuracy.

Datasets	[11]	[9]	[48]	[10]	[1]	Proposed
20 Kroner 1990	87.2	NA	94.2	94.1	93.2	**98.6**
20 Kroner 1991	86.4	92.1	95.6	92.2	96.6	**98.0**
20 Kroner 1996	98.0	96.8	99.5	97.7	**100**	99.8
20 Kroner 2008	92.8	95.5	**100**	95.2	99.6	99.9
Half Yuan Chinese 1942	70.5	NA	99.7	76.3	87.0	**95.5**
One Yuan Chinese 1997	64.6	NA	78.2	75.5	82.5	**92.2**

(a)

(b)

(c)

(d)
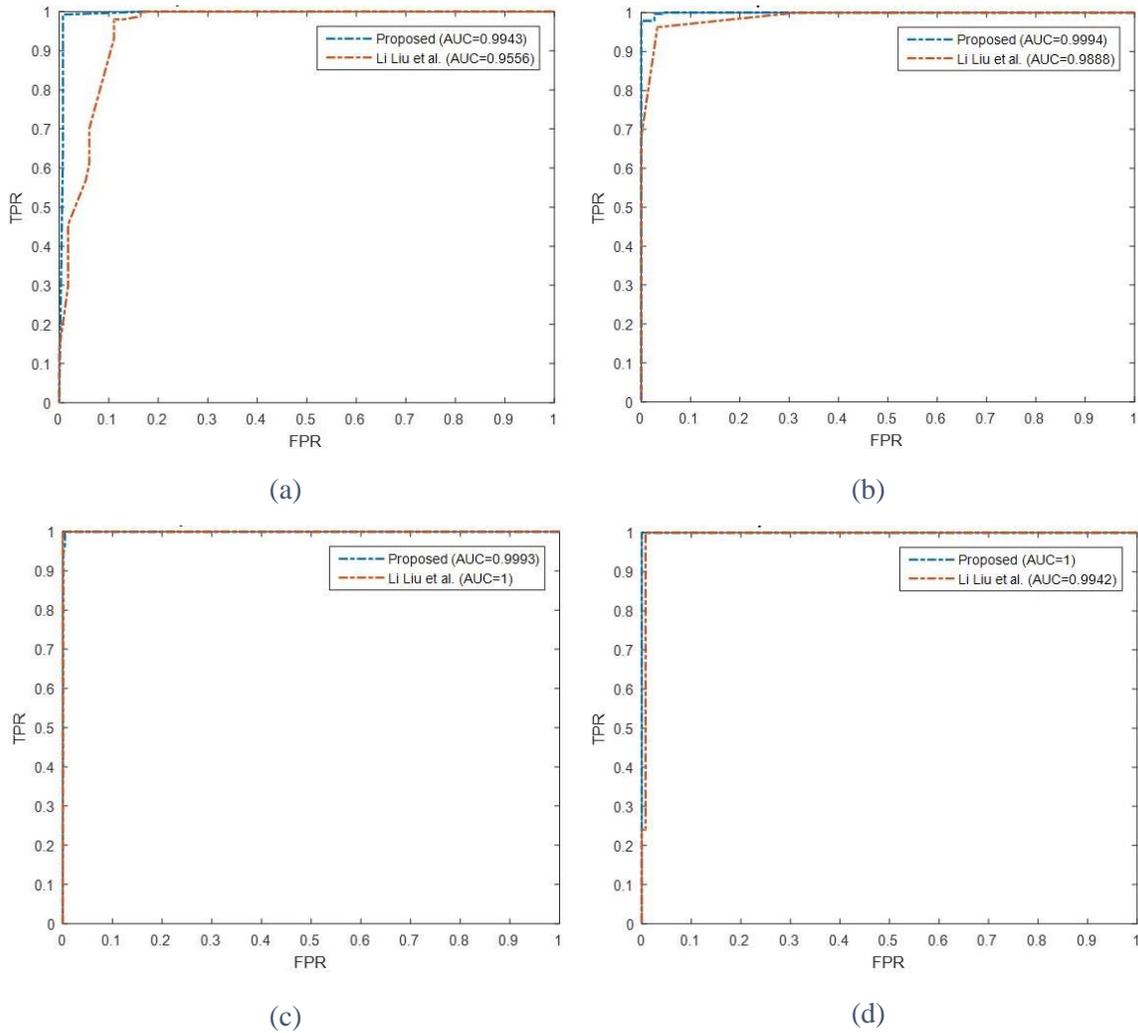
Figure 5.12. ROC comparison on four Danish coin datasets: (a) 20 Kroner 1990 dataset, (b) 20 Kroner 1991 dataset, (c) 20 Kroner 1996 dataset, and (d) 20 Kroner 2008 dataset.

As we can see clearly in Table 5.9, the proposed method with the efficient ensemble classifier discussed in subsection 5.6.3, outperformed the other previous methods in most cases. The results demonstrate that the proposed method has obtained a significant improvement in the classification

of the coins, especially for Danish 1990, 1991 dataset, and Chinese coin datasets. As the image features employed in Sun et al.'s method are specified for the 20 Kroner 1991, 1996, and 2008 coins, they cannot be applied to the 20 Kroner 1990 and Chinese coin datasets. Since the performance of the system, in this research, is more important when the system classifies the fake coins, we evaluated the proposed method by comparing it with Li Liu et al.'s method [1], in terms of Receiver Operating Characteristic curve (ROC curve) and Area Under the Curve (AUC) for the fake coins. The ROC comparison on 20 Kroner 1990, 1991, 1996, and 2008 datasets are illustrated in Figure 5.12. The ROC curves in Figure 5.12 (a) to Figure 5.12 (d) simplify the trade-off between the True Positive Rate (TPR) and False Positive Rate (FPR) for the models using various probability thresholds. In these figures also, the AUCs demonstrate a substantial performance over Li Liu et al.'s method, especially for 20 Kroner 1990, and 1991.

5-6-5 Example of a Misrecognized Coin

In this subsection, we provide an example of a coin misrecognized in our system. Figure 5.13 illustrates the image of *precipice borders* $\left(\sum_{\delta=1}^{16} BBSC_{2,8}^{\delta,\theta,\varphi}\right)$ of a 20 Kroner 1996 coin on which the regions of features are specified. Albeit the coin is a genuine one and is not a very damaged or difficult case for recognition, our system is not able to distinguish it from the fake pattern. As it can be seen clearly from the example, there are a lot of regions with deep scratches.

To make the issue clearer, we should compare the coin with another genuine coin which has also deep scratches but the system was able to recognize them. As mentioned before in subsection 5.4.3, when features have frequently zero values entire the dataset, they are removed from the feature-set and will have no effect on the classification process. Normally, the scratches have values and are on those regions of coins, which have frequently zero values. These scratches will be omitted because of the removal of sparse features. The problem occurs when the scratches are placed on those regions of coins that remain in the features set. Therefore, the values related to the scratches will be added to the correct values and mislead the classifier. In Figure 5.13, some scratches with this condition have been marked by yellow arrows. To solve this problem, we could shrink the regions by increasing the number of rings and sub-regions. As discussed before in subsection 5.6.2, increasing the number of rings or sub-regions will not necessarily lead to a better classification, although it could solve the problem for our rejected example.

Figure 5.13. An example of 20 Kroner 1996 with deep scratches rejected by the system.

5-7 CONCLUSION

In this study, we found an efficient method to detect different types of counterfeit coins. In this research, four major contributions have been made. First, we created six height-map image datasets of coins and proved that 3D approaches are remarkable in counterfeit coin detection. Secondly, in our proposed method, there was no need for restoring the degraded images. Thirdly, we extracted high discriminating features based-on *precipice border* analysis. Finally, an ensemble classifier was designed and trained by the extracted feature matrices. The experimental results for Danish and Chinese coin datasets were remarkable. It is also worth noting that the method can be simply applied to various 3D applications like face recognition and medical imaging with configuring BBCS parameters.

For future research, considering the similarity between fake and genuine coins and the lack of fake samples, implementing a system that can classify the coins and determine how much a coin belongs to a specific class, could be interesting. Because in many cases, we need to detect the coins which are not fake but are of very poor quality or worn out so that governments can take them out of circulation and replace them with new ones. Another future work for this research is to build a model that can be trained with all kinds of datasets without considering the type of coins due to the informative and discriminating features extracted by our proposed methodology.

Article 4

DECOMPOSING RELIEF MAPS TO DETECT COUNTERFEIT COINS USING A HYBRID DEEP LEARNING APPROACH

Saeed Khazaee, Maryam Sharifi Rad, Ching Y.Suen
CENPARMI, Concordia University, Montreal, Canada
{s_khaza, sey_shar, suen}@encs.concordia.ca
Under review in:
Image and Vision Computing, Elsevier

Abstract. In this research, we take advantage of deep learning approaches to improve the performance of counterfeit coin detection. We compensate for the lack of fake coins by using Generative Adversarial Network. As most of pre-trained networks accept three channels for their input, we propose a new method to represent the relief map (height-map image) in three geometrical forms with Steep, Moderate, and Gentle slope. Therefore, we propose a new channel for the height-map images for generating fake coins and training a pre-trained network. Then, we proposed a hybrid method using fine-tuning pre-trained deep neural networks to detect fake coins and provide a rejection option to increase the reliability of the system. While a small number of fake coins available in this research are used in the training process, the model is mostly trained by the images that are generated as fake coins from the genuine ones. However, the system produces remarkable results to classify the coins. It is also worthwhile to note that the method can even be applied to those types of coins that their genuine or fake counterparts have never previously been seen by the model.

6-1 INTRODUCTION

Counterfeit coin detection has been enticing researchers in the field of forensic science for decades. In non-circulating coin markets, ancient and valuable coins, counterfeiting is the most serious issue facing collectors today that could cost billions of dollars every year. In the past, rare coins were forged to deceive tourists, inexperienced people, or novices. These days, the technology of counterfeiting has grown exponentially that can even fool coin experts in some cases [3].

6-1-1 Problem Statement

Thanks to the increased necessity on the intelligent approaches to recognize counterfeit coins, image-based coin detection has continuously been evolving in recent years. According to previous works, coin detection is mostly based on 2-Dimensional image processing techniques. Many researchers have extracted features from the surface of the coin images and analyzed them. The difficulty level of coin detection vitally depends on the quality of the coin datasets used in coin recognition studies. Fortunately, all the standard datasets used for coin detection are captured with plain backgrounds, such as pure white or conveyor belt with gray levels [1]. However, fake coin detection is more challenging than coin recognition because of the very high similarity between counterfeit and genuine coins, needing more sophisticated features to distinguish between classes. Some of the standard features in coin recognition like diameter, thickness, weight, or shape may be effective for coin recognition or detecting some counterfeit coins. But we should note that these preliminary attributes are easy for counterfeiting, and the fake coins made nowadays are of high quality. Therefore, these systems cannot distinguish counterfeit coins from genuine ones when their primary physical properties are precisely the same. Recently, several methods based on image processing techniques and classification algorithms have been proposed to improve the performance of counterfeit detection systems, and many lectures and tutorials have been devoted to them [45]. Some of these methods are not very complicated and use coin colors and radius-based features to detect counterfeit coins. Unfortunately, these approaches are incapable of detecting well-forged counterfeit coins and are also not able to handle the problems related to poor-quality or worn-out coins.

In this research, we applied 3-D scanning to create six height-map image-datasets. Using such relief map images can compensate for the quality problems of the coins in many cases.

In this paper, we apply deep learning algorithms in different steps to design our proposed method. As Generative Adversarial Network (GAN) is being used for generative fake images in image processing applications, we propose a method to augment our fake coin class and compensate the lack of counterfeit coins for training the classifier based on GAN. We also decomposed the coin height-map image into three proposed channel Steep, Moderate, and Gentle slopes. Therefore, the grayscale height-map image is turned into the proposed SMG height-map channels. Then, we used transfer learning to retrain a pre-trained neural network and classify these new SMG images.

6-1-2 Related Works

Recently, a lot of articles were published for presenting image-based and non-image-based methods to distinguish fake from genuine coins. For example, in non-image-based methods, the authors in [15] proposed a method using an optical mouse to detect fake two-Euro coins. Although the method was useful for coin authentication, the report in this paper indicates that the system cannot identify most of the well-forged coins as counterfeit. In [16] a method based on acoustic signal spectrum analysis for authentic and fake coins, has been proposed. While low costs and high accuracy were reported as the strength of the method, this method was evaluated by a very small dataset only.

However, exploiting images to detect counterfeit coins can benefit from low cost and high efficiency. Feature extraction is the most critical part of image-based counterfeit coin detectors. Therefore, coin images are processed by using different techniques of feature extraction such as Fast Fourier Transformation, Gabor Wavelets, image subtraction, edge detection, and segmentation, blob detection, and other image processing algorithms. Accordingly, after extracting effective features from the coin images, other image processing techniques and machine learning are applied to classify the set of features and detect the counterfeit coins. For example, in [58], a system for Indian coin recognition by Heuristic approach and Hough Transform (HT) has been proposed. The proposed method has been limited to recognizing only the Indian Coins. Furthermore, there are some methods for coin classification in which edge information has been used as features of images with Fourier transform [12], [38]. In [4], a feature extraction method was applied to the letters and characters that were previously segmented. Albeit the results mentioned in the paper were significant, the method was not applicable for any type of coin as it was limited to characters of specific coins. However, edge features are not strong enough because they are undeniably distorted by noises such as rust, dust, and abrasion.

In [8], the structure of the coin was exploited by using spatially local coding to extract useful features from the surface of the coin image. In reference [11], the authors proposed a region binary patterns method which was rotation-invariant. They applied gradient magnitudes instead of histogram analysis and extracted rotation-and-flipping-robust features using local difference magnitude transform. Although the result is acceptable to coin classification, the method is not able to distinguish fake from genuine as it does not concentrate on the similarity of counterfeit and genuine coins. The authors in reference [1] proposed to employ local descriptors to generate image representations. However, instead of resorting to the BOVW (Bag of Visual Words) model for a

generation of vectorial representations, the authors represented a coin image in the dissimilarity space. They conducted single-class learning for counterfeit coin detection. However, the method was susceptible to any variation and not suitable for poor quality coins.

Besides the classic image processing techniques, in [13], a transfer learning approach by fine-tuning a pre-trained network to analyze the features of characters on the coin surface has been used. The authors used an ensemble method to combine results of two classifiers trained by deep features from convolutional layers as well as a third classifier that was trained on a distance-based feature of characters. The method, to the best of our knowledge, was the first use of deep learning to counterfeit coin detection. Yet, the results illustrated that the method did not have superiority over recent methods.

Since the use of 3-D height-map images discover highly confident features, they can overcome some constraints introduced by other 2-D image classification methods. In our recent works, we took advantage of height-map image analysis to detect counterfeit coins while other researchers in the literature have not applied it. In references [45], [48], the characters of the coin images were straightened by a proposed straightening algorithm. The height-map image, then, was decomposed row-wise to a set of 1-D signals, which were analyzed separately and restored by two different proposed methods. Although these approaches produced remarkable results in accuracy rather than related works, they were not efficient in detection time.

6-1-3 Proposed Framework Overview and Contributions

In this paper, we propose a new method based on deep learning algorithms in two steps to detect counterfeit coins. Here, we use six height-map image datasets to take advantage of 3-D image processing. As Generative Adversarial Network (GAN) is being used for generating fake images in image processing applications, we propose a novel method using GAN to produce new samples to our fake coin class and compensate for the lack of counterfeit coins for training the classifier. In this method, we decompose the coin height-map image into three types of Steep, Moderate, and Gentle slopes. Therefore, the grayscale height-map image is turned to the proposed SMG height-map channel. Then, we propose a hybrid deep neural network to train and classify these new SMG images. Figure 6.1 illustrates a general schema for the proposed method. In this research, the proposed methods are trained and tested with four types of Danish and two types of Chinese coins with encouraging results.

This paper concentrates on proposing a deep learning-based method to detect counterfeit coins. However, the proposed approach has much broader applications in the context of height-map image classification. The major contributions of the proposed framework are summarized as the following:

a) Creating six SMG image datasets that are proposed for counterfeit coin detection: We successfully decompose a relief map to three separate images with no overlap in the value of pixels and convert them to SMG channel. This conversion causes CNNs to pay more attention to the worthwhile relief map information like slopes of different regions.

b) Generating fake coin images for training the models by Generative Adversarial Networks to compensate for the lack of counterfeit coins in the research.

c) Proposing a hybrid CNN method based on fine-tuning by SMG relief map images instead of grayscale images.

d) No need for some essential preprocessing methods to normalize, rotate and scale the images that are essential in most image-based coin recognition systems.

e) Building a model that uses only one type of coin and detects counterfeit coins of other types. This can be very useful in case that we do not have any fake samples of specific coin to train a model and distinguish genuine from fake ones.

The rest of the paper is organized as follows. Background of research and preliminary concepts related to the proposed method will be described in Section 6.2. In Section 6.3, the proposed method will be presented. Section 6.3.1 presents the concept and proposes a method to decompose height-map images to Steep-Moderate-Gentle (SMG) slope channels. Section 6.3.2 describes the proposed GAN for generating fake samples. Section 6.3.3 designs a new hybrid CNN classifier with a rejection option for counterfeit coin detection.

In Section 6.4, the experimental setup and results are given to show the performance of the proposed method regarding counterfeit coin detection. Finally, the paper concludes with a summary of the primary contributions of this research and suggests an outline for future work in Section 6.5.
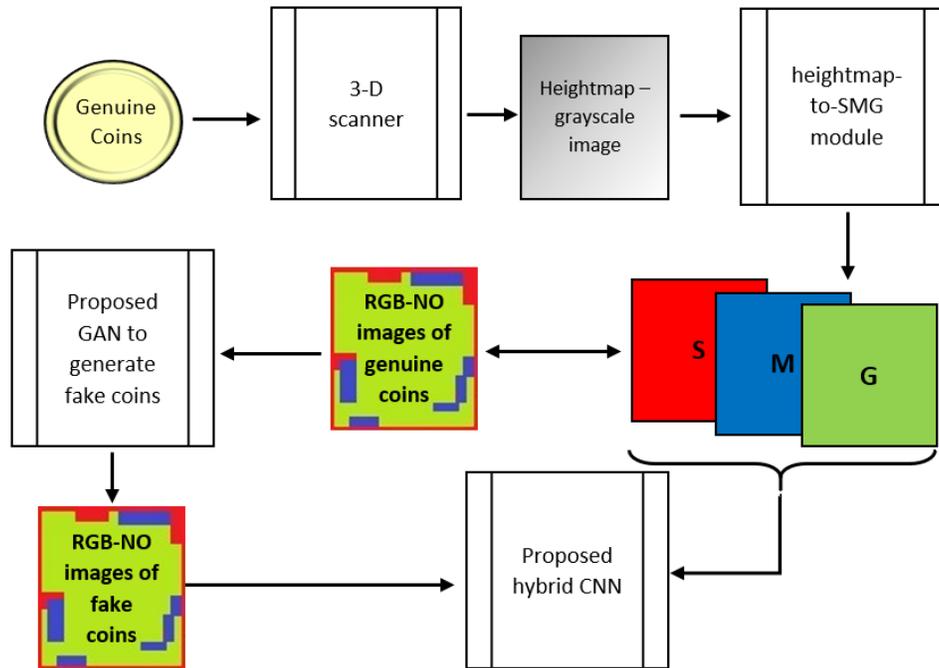
Figure 6.1. Structure of the proposed framework.

6-2 Background and preliminary concepts

6-2-1 Height-Map Image Benefits

A height-map contains one channel interpreting a distance or height from the background of a surface and is stored as a grayscale image. Darker colors represent the shorter height, and brighter ones represent longer where black is the minimum and white is the maximum height. Pixels are effectively valued by height or depth, instead of colors related to coin luminance. Therefore, the most important benefit of 3-D scanning is robustness against the quality of the coin. This capability contributed to purifying images captured from sulfated, rusted, or colored coins [45].

Representing a depth/height-map can also be found by a colormap. It is shown in Figure 6.2 (b) a color depth-map for Figure 6.2 (a). In Figure 6.2 (b), the depth image is represented as a jet colormap image, where blue and red are meaning closer and farther objects, respectively. It can be visualized that the depth change is gradual in this figure [59].

<div align="center">(a)   (b)</div>
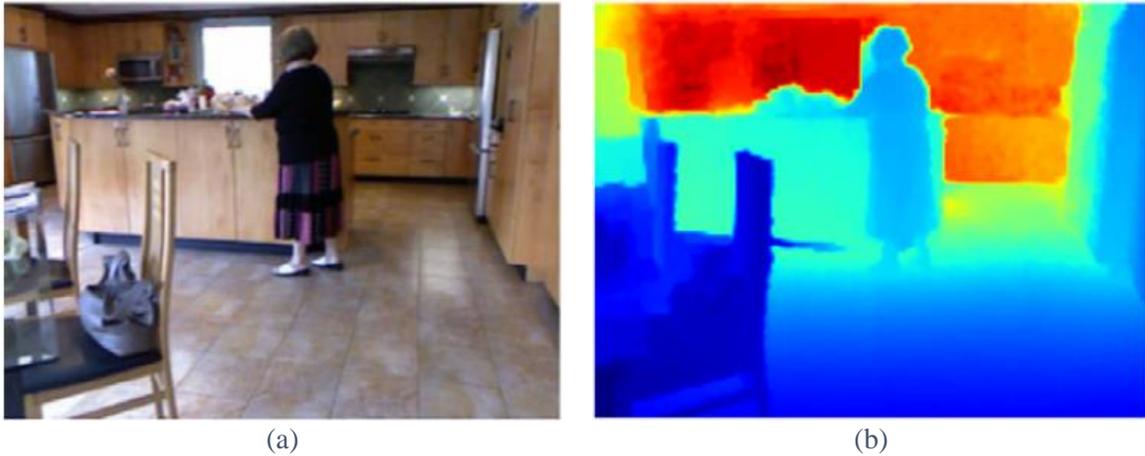
Figure 6.2. (a) Color map, and (b) normal map for height/depth image [59].

However, a colormap of depth or normal is only used for better visualization in many applications. They could also be considered in the feature extraction process. In this paper, instead of focusing on depth information, we decompose objects and important parts of objects on a coin surface based on three slopes (Steep, Moderate, and Gentle) that are very useful for feature extraction or training a deep neural network.

6-2-2 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN), also known as a ConvNet is one of the most commonly used deep neural networks in the field of image processing. A Multilayer perceptron usually has fully connected networks in all layers, also called dense layers, where each neuron of a layer is connected to all other neurons in the next layer. This characteristic of the dense networks leads them to overfit in many cases. Without concentrating on any specific segments of images, dense layers learn global patterns in input images. In CNN, the network is not fully connected in all layers. With *convolution* operation, convolution layers learn the local pattern. As can be seen in Figure 6.3, convolution operations are used to find patterns in small 2D windows of the input image. Another essential action in CNNs is *pooling,* also known as *subsampling* or *down-sampling*. The main reason to use pooling is to reduce the dimensionality and complexity. Pooling layers are usually placed after convolution layers. By reducing the dimensionality as well as the complexity of the input and parameters, the overfitting phenomenon would be suppressed.

It should be noted that *max-pooling* outperforms alternative solutions like using strides in the prior convolution layer or average pooling. As is clear in the Figure 6.3, the fully connected layer is typically put after the convolutional and pooling layers meaning that the output of the last layers

is the input to the fully connected layers. To assign each node to a specific class (a person nationality in this example), an activation function naming *softmax* is used to produce the membership degree of the input to each class, where the sum of all probabilities is 1.
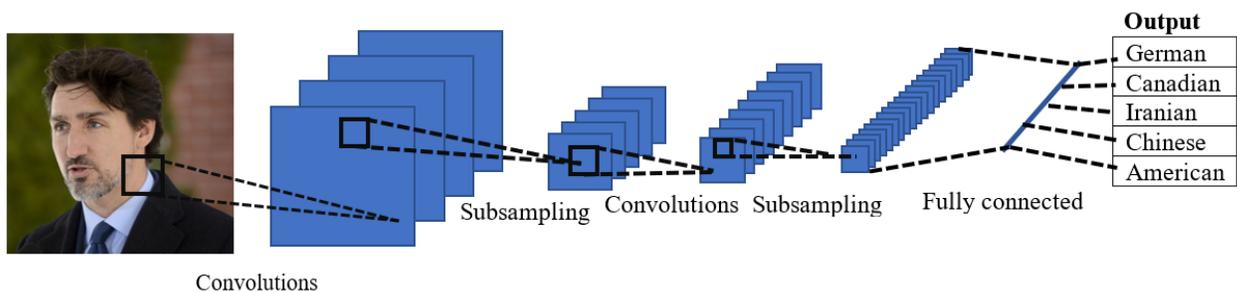
Figure 6.3. Structure of Convolutional Neural Networks.

A common and highly effective approach to deep learning when image datasets are very small is to use a *pre-trained* network. A pre-trained CNN is a saved network that was trained previously by a large dataset [27]. CNN models are getting deeper and improving by new structures that are being proposed frequently. Several structures of state-of-the-art pre-trained CNNs like LetNet-50 [28], AlexNet [29], VGGNet [30], ResNet [31], MobileNet [32], DenseNet[60], and Google's Inception v3 [33] were proposed in the literature.

A pre-trained network can be used in feature extraction and fine-tuning in two different approaches. Hence, feature extraction is to take the convolutional base layers of a previously trained network, and run the new data through it, and train a new classifier on top of the output layer. Fine-tuning changes and optimizes the filter weights of newly added blocks of the network iteratively. It freezes the first two convolution base layers. While those the frozen layers already extracted generic types of features like edges, fine-tuning will update all intermediate layers with new data [27].

6-2-3 Deep Convolutional Generative Adversarial Networks (DCGAN)

Generative Adversarial Networks (GANs) are one of the most popular deep neural networks which are used to generate fake images from a set of original images [34]. Here, two models are trained at the same time by the adversarial learning process. A generator model learns to create fake images with high similarity to its counterpart, while a discriminator model learns to distinguish real images apart from fakes. Here, both generator and discriminator models are types of CNNs and fine-tuned. The generator tries to fool the discriminator network during the training process.

By generating realistic images, discriminator learns to find dissimilarity. The result of this competition is to generate artificial images that are not easily distinguishable from genuine ones. One of the most critical challenges in this study is the lack of fake coins. Since counterfeiting a coin for the research is impossible and the fake samples provided by the Law Enforcement Office are not enough for both training and test processes, we consider GANs to produce fake samples for training our counterfeit detector.

6-3 PROPOSED FRAMEWORK

6-3-1 Steep-Moderate-Gentle (SMG) Slope Channels for Height-Map Images

Most of the pre-trained networks for image processing applications get three channels as input. To employ the pre-trained models and use grayscale images for fine-tuning, we can also repeat the grayscale images over three channels, while it would be an overload without benefiting all power of the pre-trained network. Therefore, having three channels carrying meaningful information can strengthen the networks for classification. In addition, augmenting height-map grayscale images should be more than shifting, rotating, adding noise, or other commonly used methods. Using GANs for augmenting grayscale images is remarkable. While the augmented images by GANs are useful for 2-D image processing, they are not always suitable for height-map image analysis, since the height information is not protected after augmentation, and object luminance in the image is concentrated for generating processes instead.

In this research, we propose a new method to decompose a height-map image into three different slopes. In a coin height-map image, considering spherical coordinates, we define three slopes. We first, triangulate the height-map image and find normal vectors for tringles on the coin surface by proposed gridding. Figure 6.4. (a) shows the triangulation gridding on the image matrix applied to the coin image. As in the triangulation gridding, the gray triangles cover all pixels of the image, the white triangles are disregarded with the aim of reducing the processing time.

Since the resolution of the height-map images affects the normal vectors of triangles on a specific region, we normalize the height-value by the coin diameter and thickness for normal vector computations. Then, we normalize all values by min-max normalization by:

$$Max_z = \frac{r \times t}{d} \qquad (6.1)$$

$$z_2 = \frac{z_1}{Max_z} \qquad\qquad (6.2)$$

where t and d are the coin thickness and diameter respectively, and r, z_1 and z_2 are number of rows of image matrix, the pixel value of the point, and the normalized value for z_1, respectively. To improve the decomposition process, we also consider a larger triangle for each sample whose vertices are the neighbors of each triangle. Figure 6.4. (b) illustrates the larger triangle which covers a neighborhood of the triangle sample. Accordingly, there may be an angle between these two triangles. α is the angle between the larger triangle and the triangle sample. Also, α can be computed by:

$$\alpha = \cos^{-1}\left(\frac{|\vec{N_s}.\vec{N_b}|}{\|\vec{N_s}\| \times \|\vec{N_b}\|} \right) \qquad\qquad (6.3)$$

where N_s is the normal vector of the triangle sample, N_b is the larger triangle, and $\|\vec{N}\|$ is the magnitude of the normal vectors.

A large α computed for any triangle demonstrates that this triangle is not following the pattern of its neighborhood and probably is a noise or a very tiny scratch. Therefore, we count these tiny scratches as Gentle slope, which are less important to represent the characteristics of the coin. Assuming *level* with the normal vector of $N_l = (0, 0, 1)$, Gentle slope covers all parts on the coin surface where the angle between their normal vector and N_l, is smaller than a threshold τ_1. Accordingly, Moderate and Steep slopes must meet the conditions with thresholds τ_1 and τ_2, as defined in Equation 6.4 where function C gets a normal vector of each triangle and returns its class. Figure 6.5 illustrates the normal vector distribution on coin surfaces considering Steep, Moderate, and Gentle slopes. Therefore, after decomposing a height-map image, we have three matrices representing the coin image, and we name it SMG channel.

We should note that S, M, and G matrices have no overlap in their elements. It means that if an element in matrix S has a non-zero value, the other matrices have zero value in the same elements. Hence, we can easily store them in JPEG files and visualize them in a single image with a specific type of RGB color model, which we name it SMG or RGB-NO (RGB with No Overlap). Turning the SMG image into S, M, and G matrices is also very simple. Figures 6.6 (a), (b), (c), (d), and (e) demonstrate a grayscale height-map image of a fake Chinese 1942-coin, Gentle slope, Moderate slope, Steep slope, and SMG image of the coin, respectively. Figures 6.6 (f) through (j) are the same images for a genuine Chinese coin. In this difficult case, distinguishing the genuine and fake coins is very challenging by the grayscale image. In contrast, SMG images of the coins provide

discriminative information to predict the class of the coins. Focusing on the color of the slopes of the same regions on the genuine and fake coins clearly illustrates a significant difference between them. This is inspired by the coin experts that rotate and look carefully at the coins with different azimuthal and polar angles to find dissimilarities in the edges and the slopes. However, they can also be deceived by a very-well forged coin like the Chinese coin example.

$$C(n) = \begin{cases} Gentle & if & \theta < \tau_1 \\ Moderate & if & \tau_1 \le \theta < \tau_2 \\ Steep & if & \theta \ge \tau_2 \end{cases} \tag{6.4}$$

Figure 6.7 shows a 3D visualization of an SMG height-map image in RGB-NO color map model. As can be seen in this figure, the objects are perfectly decomposed to SMG channel leading to significant details.

(a)   (b)
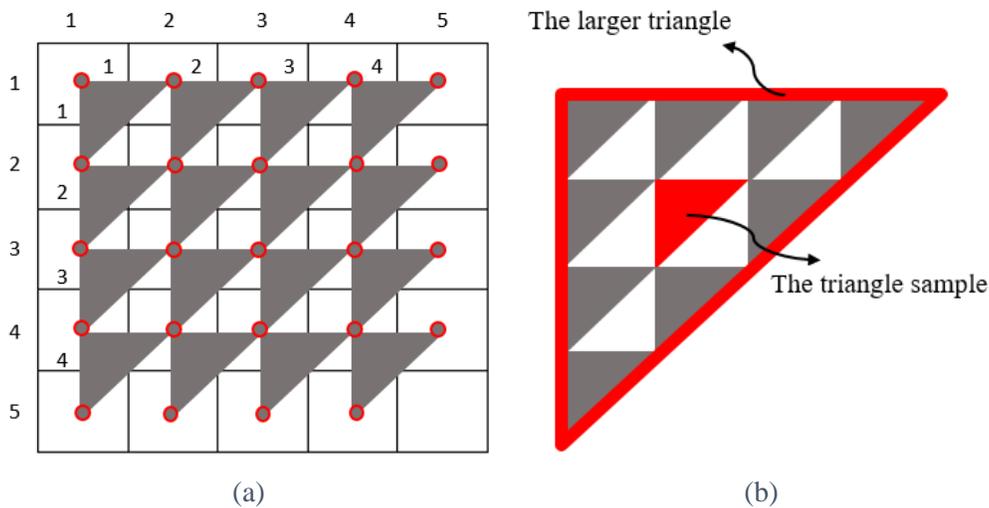
Figure 6.4. (a) Triangulation of the height-map image; the triangle array, and (b) the larger triangle.
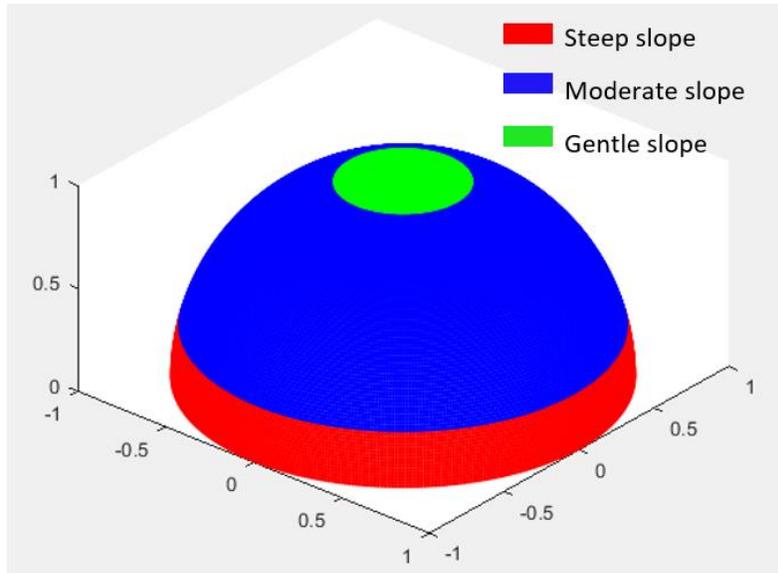
Figure 6.5. Normal vector distribution for the SMG channel.

6-3-2 Proposed Generative Adversarial Network for Generating Fake Coin

In this subsection, we design our proposed DCGAN to generate fake coin images. This network gets several SMG images of genuine and a very small number of fake coins as input and produces fake coin SMG images. As shown in Figure 6.8, the general structure of the proposed GAN is rather standard with a minor change. Since the network is fed by RGB-NO images, the output must also be the same. Therefore, we need a post-processing module to purify S, M, and G into the non-overlapping condition. Inside the network, we build a sequential model for both generator and discriminator. We initialize the network input by random noise from the latent space using a normal distribution like Gaussian distribution.

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

(f)   (g)   (h)   (i)   (j)
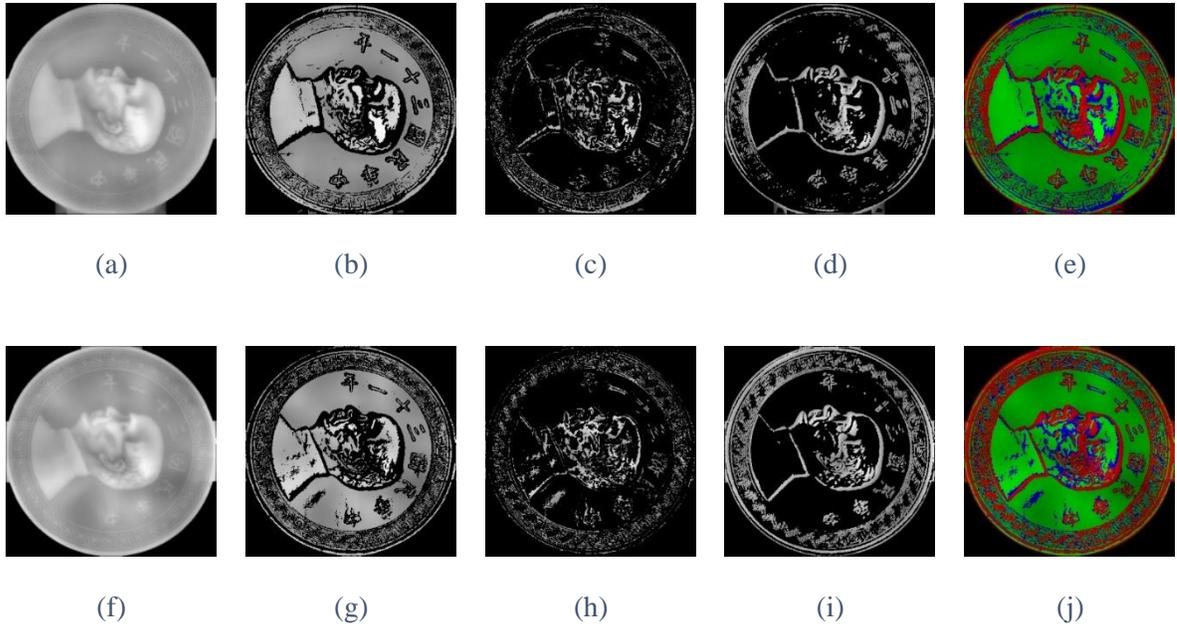
Figure 6.6. fake and genuine Chinese 1942 coin: (a) grayscale height-map image of a fake Chinese half Yuan coin, (b) Gentle, (c) Moderate, and (d) Steep slopes of the fake coin and (e) an SMG image for the fake coin. (f), (g), (h), (i), and (j) are the same image as above for the genuine coin.
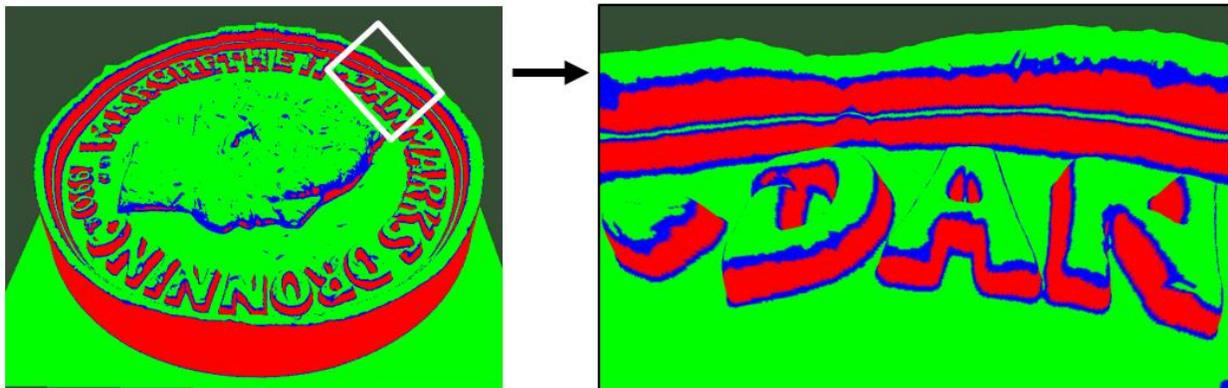
Figure 6.7. Steep, Moderate, and Gentle slopes on a 1990 Danish coin.

Figure 6.9 provides details about the layers added to generator and discriminator models. The last activation function in the generator is set to *tanh* instead of *sigmoid* or other commonly used activations. Although in most cases of deep neural networks, sparsity can play a positive role, in GAN, sparse gradients can adversely affect the training process. To tackle this issue, as suggested by [27], we use *strided* convolutions for down-sampling and LeakyReLU layer as an activation that can handle the sparsity problem by small negative activation values. In this research, the first

block of the generator transforms the input into a 50×50 512-channel feature map where the size of the training images is selected be 200×200 in this study.

Since the real images of the network inputs are SMG images, unequal coverage of the pixel space in the generator is lesser compared to when the images are the standard RGB. To stabilize the learning processes, as can be seen in Figure 6.9, we employ batch normalization after using the Conv2DTranpose or Conv2D in both the generator and the discriminator.
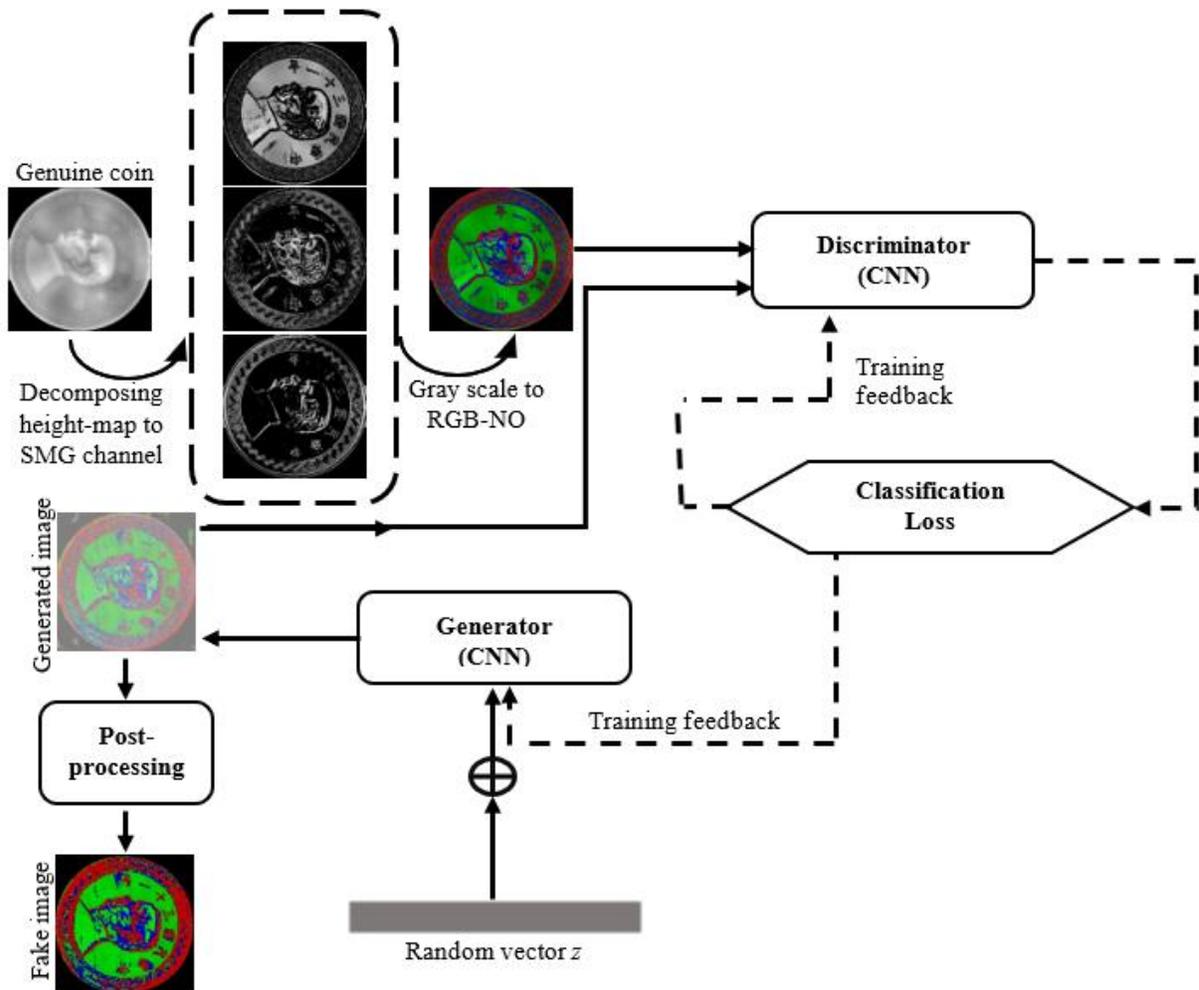
Figure 6.8. Proposed fake coin generator.

Figure 6.9. The structure of the GAN proposed for generating fake coins.

Generator

Discriminator

6-3-3 Proposed Hybrid Convolutional Neural Networks

In this subsection, we propose a hybrid CNN based on fine-tuning to detect counterfeit coin and provide a rejection option. Fine-tuning is a widely used technique for reusing a model that was pre-trained by a massive benchmark dataset. In this study, before achieving a new idea, we fine-tuned several state-of-the-art CNNs like VGG16, VGG19, and Resnet50 networks and found the best pre-trained model for our research. As VGG16 outperformed other networks, we propose a new structure based on a combination of two VGG16 pre-trained networks and three custom densely connected classifiers. These networks are pre-trained on ImageNet dataset [61], which contains millions of images with multiple dogs and cats classes. For counterfeit coin detection, we have two classes plus a rejection option. Hence, the top fully connected layers of these networks are truncated and must be replaced by the custom fully connected networks.

Figure 6.10 illustrates the general workflow of our hybrid method. In this method, we build a black box, including Model 1 and Model 2. The black box is trained in the first stage of training. In this stage, Model 1 and Model 2 are trained separately by the same training dataset. When the black box is trained, all layers in both models are frozen for the second stage of training. In the second stage of training, we train the last fully connected classifier by tuning the weights. The input of the last classifier is the output of the black box, which is a tensor of two real values in the range (0,1), which are produced by using all training samples for the second-round training. Normally these values are used for classification, meaning that a value greater than 0.5 is assigned to class 0; otherwise, it is classified as class 1. As figure 10 illustrates, the outputs go indirectly to neurons A and B of the densely connected classifier. Still, only one of the neurons A and B are turned on as the input of the classifier. Before one of the outputs O_1 and O_2 participates in the training of the last classifier, $S(O_1)$ that is a selective function turns on one of the neurons. The input, models, blocks, and the function employed in the proposed framework are defined below.

Input: of the hybrid method is an SMG height-map image that facilitates feature extraction for the models due to clarification of the important regions in the process of decomposition discussed earlier in subsection 6-3-1.

Model 1: is a completely frozen VGG16 with a custom densely classifier on top. The custom classifier has a flatten, a fully connected layer with 256 neurons and activation *relu*, and a dense output node with one neuron and activation *sigmoid*. It would merely be used for extracting features to avoid misleading the classifier by poor quality or uncertain cases and enabling a rejection option. Model 1 is trained in only one epoch. Therefore, we employ Model 1 to handle

the misleading issue related to damaged coins, which may happen in Model 2. The logic behind this decision will be discussed in section 4, comprehensively.

Model 2: is a VGG16 pre-trained network that we unfreeze a few frozen layers at the top to fine-tune the network. We fine-tune the last three convolutional layers and all other layers are frozen. We also add the custom model the same as Model 1 to define our densely connected classifier for Model 2. Fine-tuning the network is started by RMSProp optimizer with a very low learning rate. The initialized weights of unfrozen layers are continuously updated over the training process leading the network to learn features specific to counterfeit coin detection. Because this model tries to update the weights to assign the samples to one of *fake* or *genuine* classes, it is not able to keep interclass distance related to damaged coins or uncertain cases for rejecting. Therefore, the model can perfectly classify the coins when their quality is not very poor. As mentioned earlier, we let Model 2 to train enough; at the same time, we apply Model 1 with the aim of balancing the problem of misleading the last densely connected classifier.

Selective: is a function that turns on one of the neurons A or B as the input of the next fully connected network. This function gets O_1 to make the decision. $S(O_1)$ turns on neuron A and turns off neuron B if $t_1 < O_1 < t_2$; otherwise, neuron B would be turned on and vice versa. The thresholds t_1 and t_2 are selected experimentally to 0.3 and 0.6.

Since we have different quality of coins in this study, the very damaged and worn-out samples are difficult to be classified. In this case, a better decision could be to reject the coin to increase the reliability of the method. Apart from reliability, most of the very damaged coins do not have much value in the market or circulation that can be rejected even if they are genuine.

Last Densely Connected classifier (LDC): is a very simple fully connected network that has only one hidden layer that experimentally has four neurons. As mentioned earlier, the output O_2 is very close to the classification results except when the input sample is a very damaged, worn-out, technically speaking, misleading by outliers. Therefore, this simple network does not need many hidden layers, since it only has the mission for keeping a distance between certain and uncertain predictions. The last densely connected classifier is trained by one input neuron and four neurons in the hidden layer. We select the activations *relu* and *sigmoid* for hidden and the output layer with *Adam* optimizer. Therefore, in the test step only, the output of LDC (O_{LDC}) is assigned to one of the three classes, fake, genuine, or rejected. It should be noted that we train the model by only fake and genuine samples, and there were no rejected samples in our preliminary hypothesis. Here also,

if $O_{LDC} < 0.3$ then the coin is assigned to class fake. If $O_{LDC} > 0.6$, the sample is supposed to be a genuine coin, and if $0.3 \leq O_{LDC} \leq 0.6$ the sample is rejected.
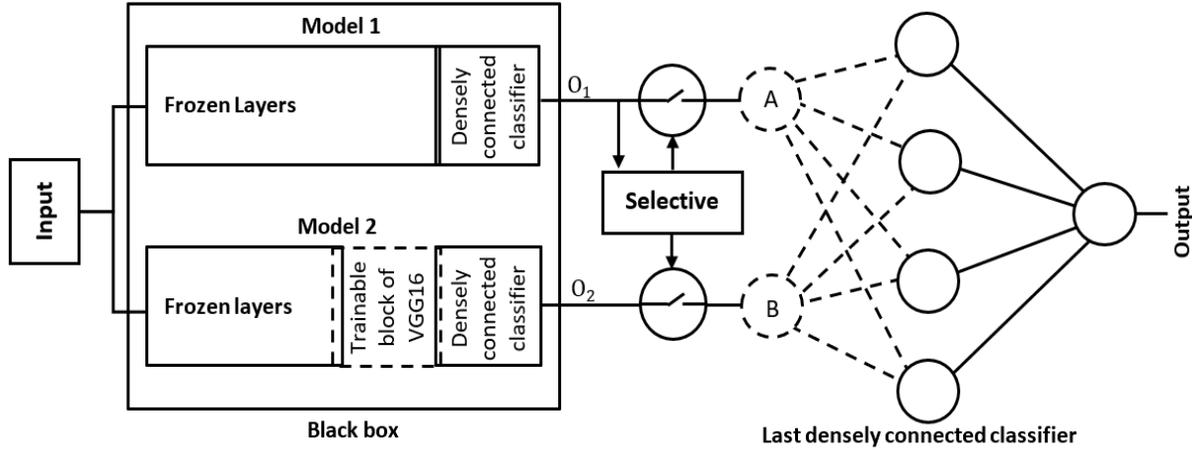
Figure 6.10. Proposed hybrid Convolutional Neural Network structure.

6-4 EXPERIMENTAL RESULTS

6-4-1 Datasets

The coins in this research were provided by the Law enforcement office of Denmark, including fake and genuine coins. In collaboration with *Ultra Electronics Forensic Technology Ltd. Co* and coin experts in Montreal, we labelled Danish coins available for this research. We also applied the majority votes for labelling the Chinese coins by at least five coin-experts.

To image acquisition, we employed a very precise 3-D scanner in the name of *IBIS TRAX*. The scanner was patented by *Ultra Electronics Forensic Technology* company.

Inside the machine, they installed a built-in microscope. With a very high resolution of the height-map images in the order of 6 microns and lateral resolution in sub-micro, users can extract useful information from tiny topographical features. The machine uses a five-group of adjustable LEDs which allow users to acquire the coin image from different angles. The machine can produce both 2-D and height-map images as a result of scanning [4].

In this study, we created six types of height-map image datasets by the powerful 3D scanner. Four different years of 20 Kroner 1990, 1991, 1996, and 2008 and two different Chinese coins: half Yuan 1942 and one Yuan 1997 have been selected. We also generate SMG images of these datasets to apply them for training and testing the proposed hybrid classifier. The coins provided by the

Law Enforcement office as having fake coins is a crucial challenge for this research, and the counterfeit samples of the selected coins are very well-forged, and even coin experts can be deceived in many cases. These datasets are a mix of good and poor-quality coins or degraded images resulted from poor scanning.

The scanner that we used in this research results in a 3550×3550 image resolution (with the gray level of 0 to 255 for JPG and 0 to 65535 for JPEG2000 images). In this study, we used the JPG images and resized them to 224×224.

Table 6.1 illustrates the statistics of coin images for each dataset. In order to train and evaluate the system, coin samples are stochastically split into training (20%), validation (40%), and test (40%) sets before any augmentation. The validation set is used to avoid over-fitting and tuning the parameters of the neural networks. The proposed augmentation approach allows us to keep most of the samples for the test and validation processes. Table 6.1 shows the statistics about the datasets and some information on the physical and chemical characteristics of the coins. Figure 6.11 provides examples of the height-map images of the fake and genuine coins from the six datasets.

Table 6.1. statistics of the six coin datasets.

Type of coin	Diameter	Thickness	Weight	Composition	Genuine	Fake
20 Kroner 1990	27 mm	2.35 mm	9.3 g	Cu, Al, Ni	300	100
20 Kroner 1991	27 mm	2.35 mm	9.3 g	Cu, Al, Ni	300	128
20 Kroner 1996	27 mm	2.35 mm	9.3 g	Cu, Al, Ni	300	140
20 Kroner 2008	27 mm	2.35 mm	9.3 g	Cu, Al, Ni	300	234
Half Yuan 1942	28 mm	2.30 mm	9.06 g	Cu, Ni	30	32
One Yuan 1997	25 mm	2 mm	6.1 g	Ni, plated steel	28	32

Figure 6.11. Examples of genuine and fake Danish and Chinese coin images: (a) Danish 1990, (b) Danish 1991, (c) Danish 1996, (d) Danish 2008, (e) Chinese 1942, (f) Chinese 1997; and (g), (h), (i), (j), (k), and (l) are the fake counterparts of the coins respectively.

6-4-2 Augmentation

To provide enough training data and prevent overfitting problems, we apply two types of augmentation techniques. We employ one of them to generate genuine coins, and another to produce fake coins in SMG channel since the input of the proposed method is an SMG image.

6-4-2-1 Classic Augmentation of the SMG Images for Genuine Class

A specific type of a genuine coin in the market or circulation is produced from a single source. Therefore, the genuine coins are physically the same except their quality and capturing rotation. Hence, augmenting train data for genuine class can be done by rotation and adding some noises. We add some Gaussian, salt and Pepper, and speckle noises to produce poor quality coins, and in some cases, we only rotate the images to produce good quality coins.

6-4-2-2 Augmenting SMG Images by GAN for Fake Class

Unlike genuine coin images that follow the same patterns, fake ones are mostly from different sources. Therefore, classic augmentation may not be enough to generate fake samples. As mentioned earlier, we proposed a GAN based method to produce fake samples. Figure 6.12 in rows (a), (b), and (c) illustrate examples of SMG images for genuine, fake, and fake generated by GAN of the dataset used in this research, respectively. The quantity for enough augmented data for training set is selected by a logic reported in reference [1]. In the proposed method, a significant decrease in EER can be seen when the size of the training set increases to the size mentioned in

94

Table 6.2, after which increasing the size of the training set only contributes to asymptotic performance. This phenomenon may be due to the fact that some images of the original coin in the data set are too corrupt. As the number of instances in the training set increases, it becomes more likely that these broken images will be involved. Table 6.2 provides information about the number of samples used in train, validation, and test data. We applied the entire process above for the grayscale images for other experiments.

Table 6.2. Number of image samples for train, validation, and test data.

| Datasets	Train		Augmented Train		Validation		Test	
	Genuine	Fake	Genuine	Fake	Genuine	Fake	Genuine	Fake
20 Kroner 1990	60	20	500	500	120	40	120	40
20 Kroner 1991	60	26	552	552	120	51	120	51
20 Kroner 1996	60	28	455	455	120	56	120	56
20 Kroner 2008	60	48	520	520	120	93	120	93
Half Yuan 1942	6	8	250	250	12	12	12	12
One Yuan 1997	6	6	228	228	10	12	12	12
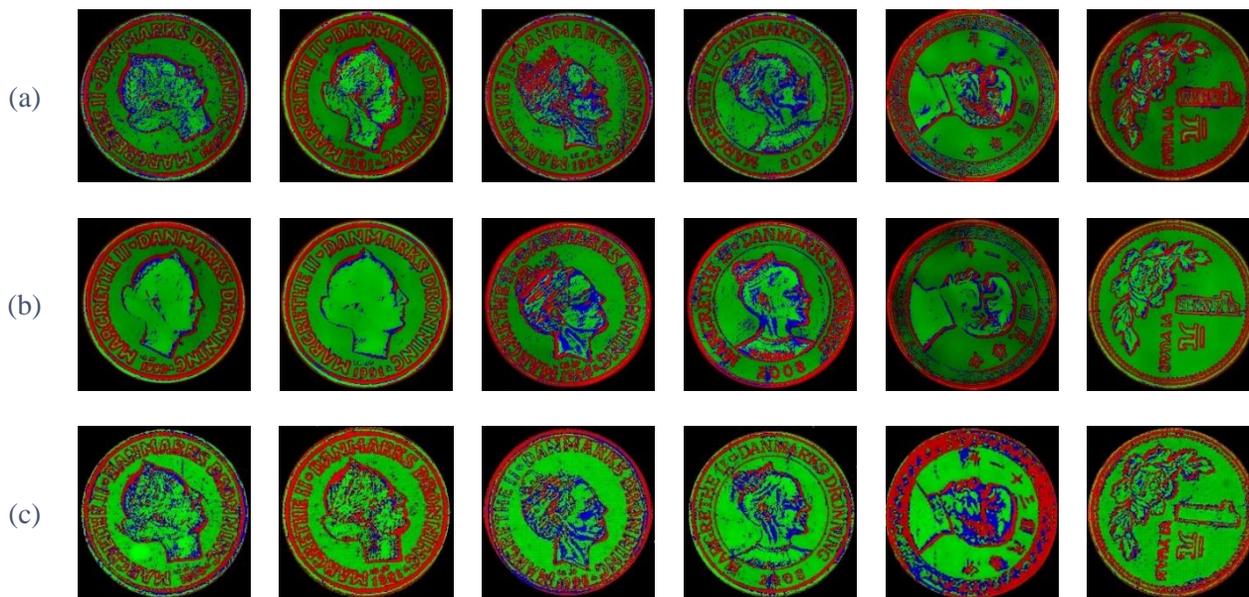
Figure 6.12. SMG images for the six datasets: (a) SMG of original genuine images, (b) SMG of the original fake images, and (c) SMG of the generated fake images.

6-4-3 Experimental Setup and Results

In this subsection, we set up some parameters of the three main proposed methods. In decomposition of the grayscale images, we set $\tau_1 = 30$ and $\tau_2 = 82$ degrees. The setting for the

proposed GAN is provided in Figure 6.9. We also provide the results of experiments to setup our proposed hybrid neural network for counterfeit coin detection. To select pre-trained CNNs for Model 1 and Model 2 of the *black box*, mentioned in subsection 6-3-3, we compared some state-of-the-art pre-trained CNNs in terms of accuracy. In Table 6.3, we demonstrate the remarkable effect of using SMG channels in the training process. As Table 6.3 illustrates, all the models fine-tuned by SMG Danish 1990 images outperformed the models trained by grayscale images. By this experiment, we found that VGG16 pre-trained by ImageNet dataset performed better than others by far. Therefore, we selected VGG16 for the base of Model 1 and Model 2. To emphasize the significant role of SMG channel for feature extraction, in Figures 6.13 and 6.14, we visualize some activations of block1_conv1 and block1_conv2 of the fine-tuned VGG16 that was trained separately by datasets SMG Danish 2008 and grayscale Danish 2008. In these figures, we can compare a fake coin and a genuine one and can see the effective features extracted by the SMG-based model rather than the grayscale-based one.

Table 6.3. Comparison of some fine-tuned state-of-the-art pre-trained models in terms of accuracy for grayscale and SMG images on the validation set.

Fine-tuned Models	Grayscale images			SMG images		
	Epoch=10	Epoch=20	Epoch=50	Epoch=10	Epoch=20	Epoch=50
VGG16	70.00	72.50	73.12	94.37	93.75	95.62
VGG19	65.62	75.00	70.62	82.50	84.37	82.50
ResNet50	60.00	65.62	67.50	74.37	73.75	74.37
Google's Inception-v3	73.75	76.25	79.37	85.00	82.50	86.87
DenseNet	70.00	68.75	72.50	81.25	85.62	84.37

The important configuration about Model 1 and Model 2 that we left in the previous section was the reason for setting the number of epochs equal to 1 for Model 1 and 50 for Model 2 in the first stage of training. To clarify this issue, we fine-tuned three VGG16 networks, experimented and visualized their output values (predictions) by three different training epochs equal to 1, 5, and 20. As can be seen clearly in Figure 6.15, with increasing the number of epochs in all experiments, the models learned to predict a value close to 1 or 0. While the predictions of the good quality coins had a fast upward trend to approach 0 or 1 depends on their classes, the model which is trained by samples with uncertain classes needed more training epochs for reaching this goal while most of these coins misled the classifier. Hence, less training epochs can keep the uncertainty of the sample. Since in the training step, we have a mix of all difficult and easy cases together and have

no prior knowledge of them, we employ Model 1 in the black box of the hybrid model to keep uncertainty of the outliers to avoid misleading the classifier and enabling the rejection option.

As the proposed hybrid method benefits from Model 1 and function *Selective*, it can stop the adverse effect of most of the outliers on the training process and prevent misleading the classifier. Figure 6.16 shows training and validation accuracy for a single fine-tuned VGG16 and the proposed method. The high accuracy of 83.1% and 91.8% for both training and validation in the first epoch visualizing in Figure 6.16 (a) illustrates that the VGG16 network can produce remarkable results even before fine-tuning. A comparison of our proposed classifier and the single fine-tuned VGG16 in Figures 6.16 (a) and (b), demonstrates the hybrid method is successful in both training and validation accuracy rather than the single VGG16.

In Table 6.2, the results on the test set for the single VGG16 and the proposed classifier for six coin-datasets have been provided. To comprehend the effect of the rejection option on the performance of the models, we define True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN), Rejected Positive (RP), and Rejected Negative (RN) for counterfeit coin detection. For a counterfeit detector, TP, which is equal to *Detection Rate* indicates the percentage of detecting counterfeit coins. TN illustrates the percentage of recognizing genuine coins. FP is the percentage of the system failure when the genuine coins are falsely classified as the fake class. False Negative shows the rate of the system failure when the counterfeit coin detector wrongly assigns the fake coins to genuine class. Here, we define RP and RN, meaning the percentage of counterfeit and genuine coins that are rejected, respectively. In the single fine-tuned VGG16, the samples with the prediction value between 0.3 and 0.7 were rejected. In this experiment, we also trained the models by only Danish 2008 coins and tested them with all types of coins together and obtained encouraging results.

As can be seen in Table 6.2, in the proposed hybrid classifier, the errors FP and FN are less than the fine-tuned single VGG16, while rejection rate RN and RP in most datasets are greater in the proposed method, as we expected. This portion demonstrates that by increasing the reject rate, the model predicts better since uncertain samples (that probably would be rejected in the test process), participate differently in the training step. In all datasets, the hybrid method outperformed in the mentioned criteria. TPs in all datasets are less than TNs, meaning that detecting fake coins is more challenging than detecting genuine ones. It can be resulted by the imbalance nature of data (before augmentation) for fake and genuine coins in the training set. For example, the system errors about classifying fake coins as genuine (FNs) in Danish 1990, 1991, 1996 and 2008 have reached zero, and the system failure when the fake coins are falsely classified as genuine (FPs) in all types of

97

coin datasets have been dramatically reduced. This means that there are several samples with uncertain classes that must be ignored in the training process. Rejection of the samples with reducing the errors at the same time illustrates that similar samples have not been select by the *selective* function in the training process.
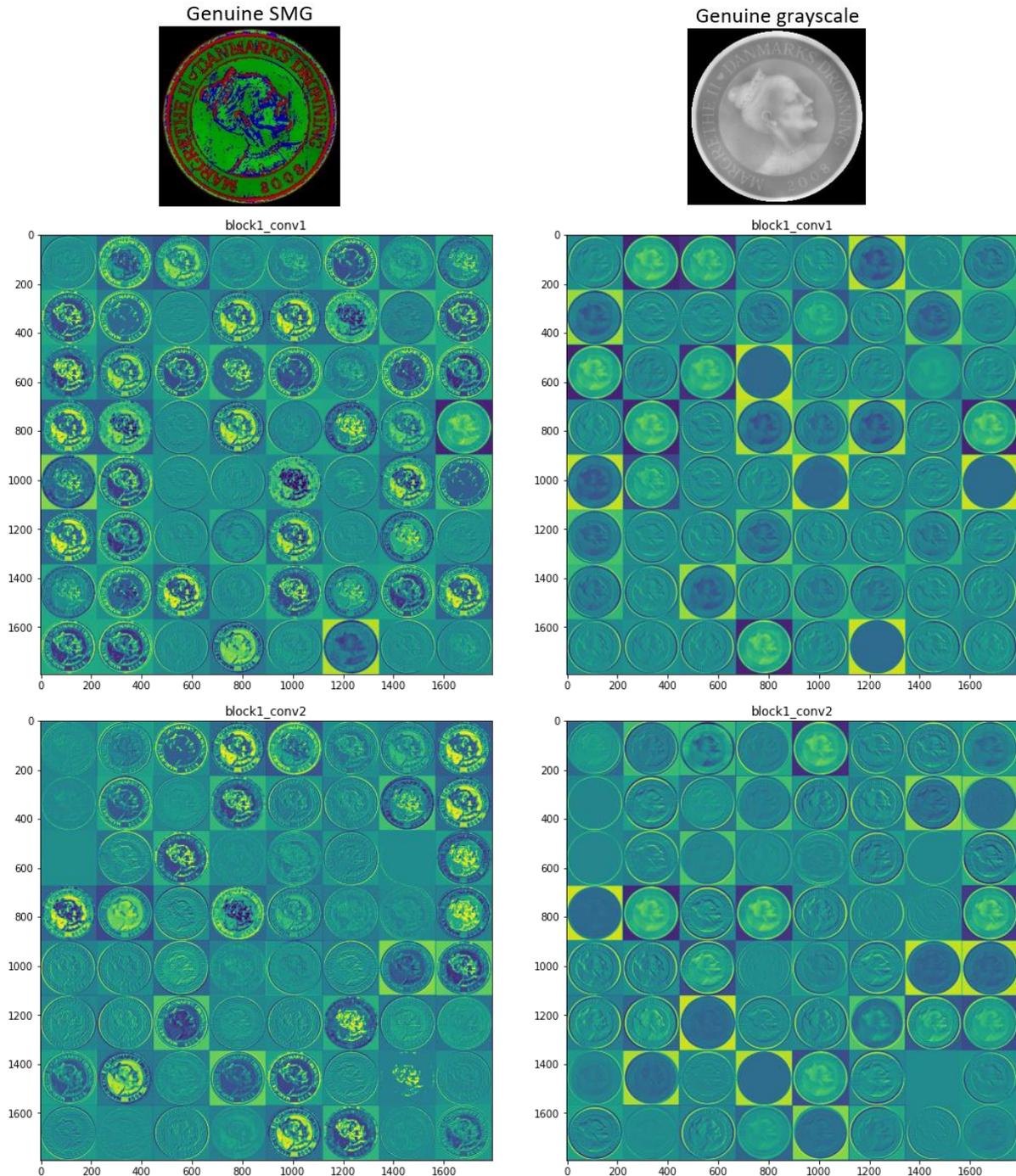
Figure 6.13. Channels of layer activation on a genuine 20 Kroner 2008 SMG and grayscale images.

Fake SMG
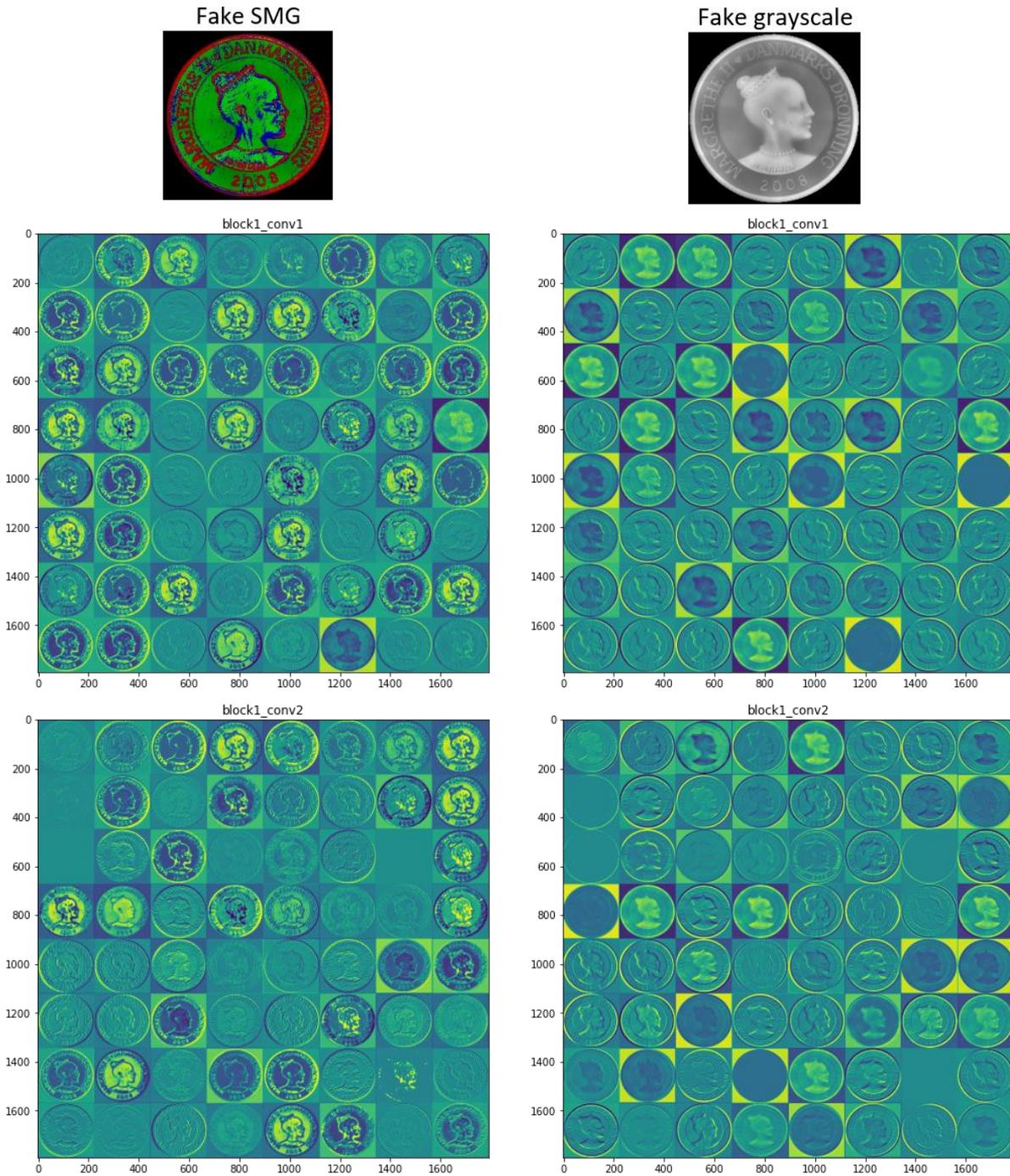
Fake grayscale

Figure 6.14. channels of layer activation on a fake 20 Kroner 2008 SMG and grayscale images.

In Figure 6.17, we provide the results of accuracy, Rejection Rate (RR), FPR, and FNR on all-types-together dataset by selecting different thresholds of *selective* function. As can be seen in this figure, in terms of accuracy, FPR, and FNR the best selection for t_2 is 0.6. Although by selecting $t_1 = 0.1$ we obtained an accuracy a bit higher than when we selected $t_1 = 0.3$, in terms of FPR and FNR errors, the best selection was $t_1 = 0.3$.
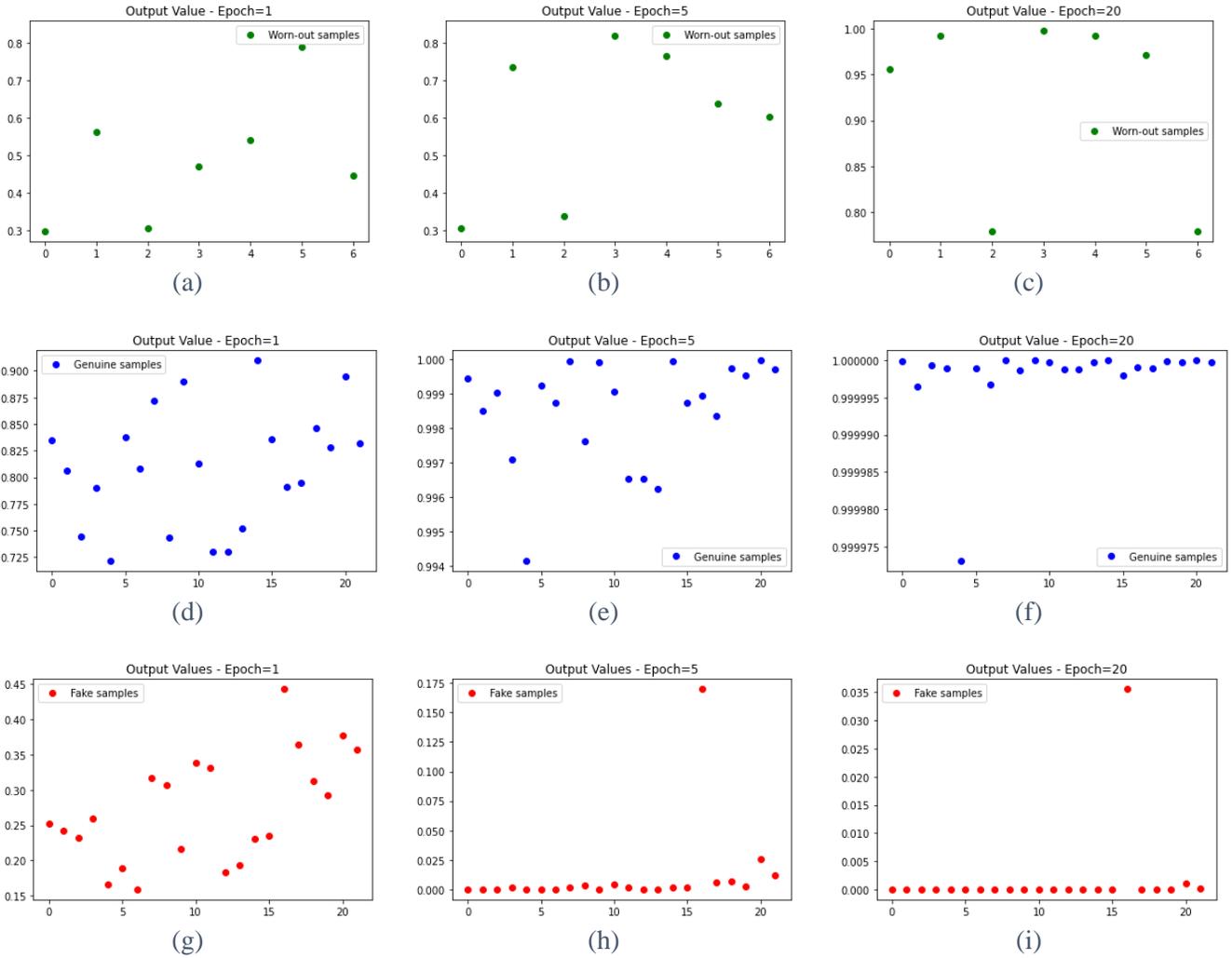
Figure 6.15. Effect of the number of epochs in the training fake, genuine, and uncertain cases. (a) through (c), (d) through (f), and (g) through (i) are results of a fine-tuned VGG16 for some worn-out, genuine, fake coins, respectively, when epochs numbers are equal to 1, 5, or 20.
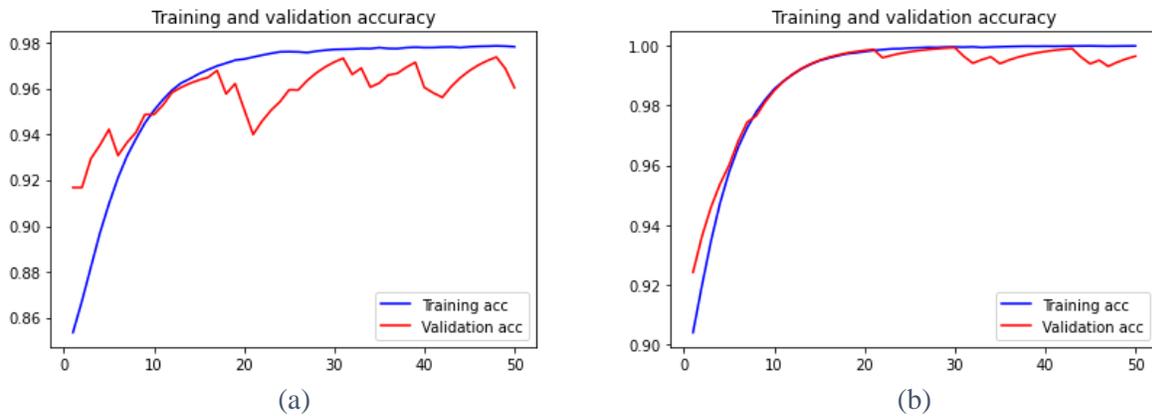
Figure 6.16. Comparison of a single fine-tuned VGG16 and the proposed classifier in the training process.

Table 6.4. Comparison of the fine-tuned VGG16 and the proposed hybrid classifier in terms of TP, FN, RP, FP, TN, and RN.

| Datasets	Fine-tuned VGG16 with rejection option						Proposed hybrid classifier					
	TP	FN	RP	FP	TN	RN	TP	FN	RP	FP	TN	RN
20 Kroner 1990	0.75	0.225	0.025	0.1	0.892	0.008	0.921	0	0.079	0.017	0.942	0.042
20 Kroner 1991	0.765	0.235	0	0.1	0.883	0.017	0.922	0	0.078	0.024	0.952	0.024
20 Kroner 1996	0.946	0.054	0	0.017	0.983	0	1	0	0	0	1	0
20 Kroner 2008	0.806	0.183	0.011	0.058	0.942	0	0.989	0	0.011	0	1	0
Half Yuan 1942	0.75	0.25	0	0.25	0.75	0	0.833	0	0.167	0	0.833	0.167
One Yuan 1997	0.667	0.25	0.083	0.2	0.8	0	0.833	0.083	0.083	0	0.917	0.083
All types together	0.742	0.174	0.083	0.105	0.873	0.022	0.835	0.075	0.09	0.026	0.917	0.058

Accuracy is another popular metrics to compare the performance of methods. As the method has a rejection option, the accuracy of the models can be calculated by:

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN + RP + RN)} \tag{6.5}$$

Figure 6.18 illustrates the proposed hybrid classifier performs better than a fine-tuned single VGG16 in terms of accuracy. The proposed method has an accuracy of 87.56% when we trained the model by only 20 Kroner 2008 and applied to all types of coins together in the testing process. In this case, we keep only the fake and genuine classes and ignore the type of coins and mix all data. As in both approaches in train and test the system, the results are remarkable; it could be the significant effect of SMG channel in recognizing fake coin patterns. This means that the Steep, Moderate, and Gentle slopes on the relief map of coins follow a certain pattern while counterfeit coins do not. Without decomposing the relief maps and using grayscale images, the method did not perform well with an accuracy of 61.24% when we trained the model by only 20 Kroner 2008 and test it by all types of coins.
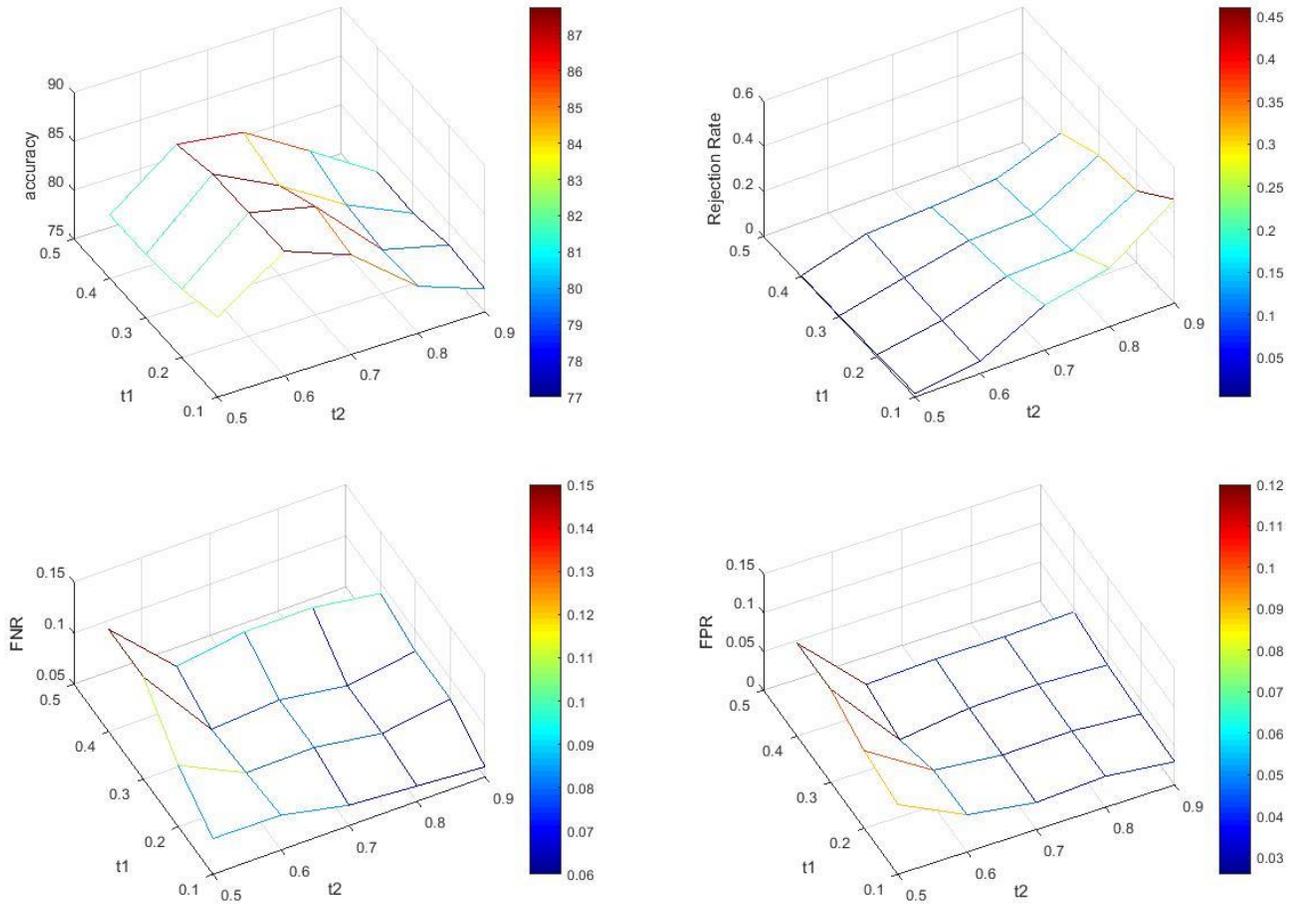
Figure 6.17. Results of Accuracy, Rejection Rate, False Positive Rate, and False Negative Rate by selecting different thresholds of $t_1\ and\ t_2$ in the *selective* function on all-types-together dataset.
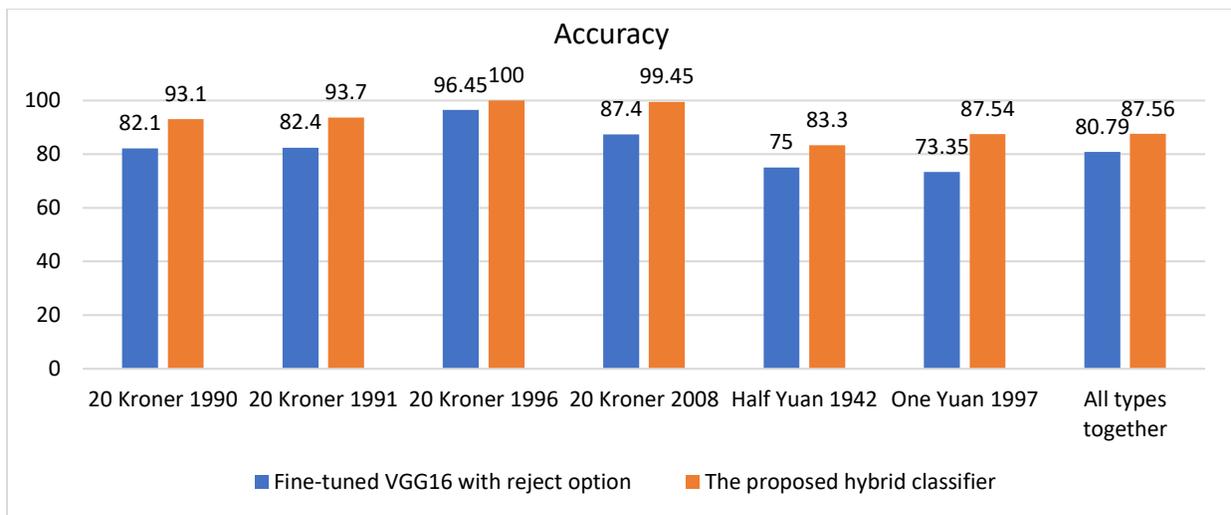

Figure 6.18. Comparison of the fine-tuned VGG16 and the proposed hybrid classifier in terms of accuracy.

6-4-4 Comparison with Other Methods

We compared the proposed method with five recent works published in the field of counterfeit coin detection and coin recognition, noting that the data used for this comparison was the same as we used to train and evaluate our proposed method. First, we calculated the accuracy of the models trained by the six datasets and the case of all coins together and compared them. Table 6.5 shows that the proposed method has a higher accuracy than other methods, except for a slightly lower accuracy in dataset Danish 1990 than the method in [1].

Table 6.5. Comparison of the proposed method and the previous methods in terms of accuracy.

Datasets	[10]	[11]	[1]	[48]	[13]	Proposed
20 Kroner 1990	90.62	85.62	**93.12**	91.25	76.25	93.1
20 Kroner 1991	89.47	88.30	92.98	90.05	77.78	**93.7**
20 Kroner 1996	93.90	95.77	96.71	94.37	91.48	**100**
20 Kroner 2008	95.2	92.8	98.12	97.65	93.90	**99.45**
Half Yuan Chinese 1942	70.83	66.67	70.83	79.17	62.5	**83.3**
One Yuan Chinese 1997	66.67	62.5	79.17	70.83	70.83	**87.54**
All types together	73.18	71.74	75.78	79.04	70.44	**87.56**

6-5 CONCLUSION AND FUTURE WORK

We scanned the coins by a powerful 3-D scanner that could produce both 3-D and 2-D images.
In this article, we proposed a new hybrid method based-on fine-tuning CNNs for counterfeit coin detection. In this paper, we proposed new methods to decompose the grayscale images of coins to the SMG channel and produced six SMG image datasets. The SMG images have been fed by a proposed GAN to generate fake samples since the real fakes were so limited in our research. The augmented data were used to train the hybrid CNN method, which could provide the rejection option for the samples with uncertain class. We demonstrated by several experiments using the proposed SMG channel has a significant effect on counterfeit coin detection than using grayscale of the height-map images. The method performed better than previous ones related to fake coin detection. The method could even be applied to those types of coins that their genuine or fake counterparts have never previously been seen by the model.

For future work, it could be interesting to include the grade of the coins in the details of the datasets to enrich the rejection option of the classifier. Then the system could be trained to keep coins with

specific grades and reject the coins with lower ones. In addition, using the proposed SMG decomposition can be extended to more channels. Then it can be useful for other applications like face recognition and medical imaging.

Conclusion

The Danish police department started the investigations collaborating with a forensic technology company in Montreal, Canada. The Danish police and law enforcement department have provided the coins, especially the counterfeit ones. We scanned the coins by a powerful 3-D scanner that could produce both 3-D and 2-D images.

In this thesis, we analyzed 3-D height-map images by four proposed methods to extract new sets of effective features that were useful for counterfeit coin detection. In Chapters 3 and 4, we solved the degradation problem of shiny coin images due to the scanning process and extracted features from row-wise signals of the coin images. In Chapter 5, a 3-D approach to detect and analyze the precipice borders from the coin surface were proposed. We also introduced the Binned Borders in Spherical Coordinates (BBSC) to extract significant features while ignoring the degradation problem. We also took advantage of stack generalization to classify the coins and add a reject option to it. In Chapter 6, we proposed a new hybrid method that used a box of fine-tuning CNNs. A new approach to decompose the grayscale images of coins to the SMG channel and produced six SMG image datasets was proposed to feed a proposed GAN for generating fake samples. Then, we trained the hybrid deep learning method by the augmented data and equipped the classifier with a rejection option for classifying the samples with uncertain classes.

7-1 ADVANTAGES AND DISADVANTAGES OF THE PROPOSED METHODS

All the proposed methods introduced in this thesis have advantages and disadvantages that we describe briefly in Table 7.1.

Table 7.1. Advantages and disadvantages of the proposed methods in Articles 1, 2, 3, and 4.

	Advantages	**Disadvantages**
Articles 1&2	1) These methods were simple to understand and easy to implement. 2) They did not need much RAM or a powerful CPU or GPU for the training process comparing with other proposed methods.	1) Since the proposed methods focused only on the characters on the ring part of the coins, they are not applicable to those kinds of coins that do not have characters on their ring part. 2) These methods concentrate on restoring the degradation problem due to poor scanning of shiny coins to precise the height information while this was an overload to the system and was not successful for all cases. Therefore, the straightening algorithm and restoring process that was used in both methods prolong the detection time. 3) Apart from the detection time, the methods are not competitive with the two latest proposed methods in terms of the detection rate, accuracy, and other metrics.
Article 3	1) In this proposed method, the degraded problem of shiny coins was ignored, and there was no need for restoring the degraded images. 2) A high discriminating feature set based-on *precipice border* analysis was extracted that was inspired by coin experts' acts. The experimental results illustrated much better performance than the previous methods. 3) In terms of RAM, processor, and other hardware required for the research, this method is comparable with the previously proposed methods.	1) Albeit the method ignored the degradation problem and faster than the previous method, it could not be used for real-time purposes and was suitable for forensic intentions of counterfeiting ancient and precious coins. 2) The method cannot handle some of the specific problems related to scratched coins explained completely in subsection 5.6.5.

	4) The methods supplied by a rejection option handling the samples with uncertain classes.	
Article 4	1) The attitude towards features in the proposed method resembles the proposed method in Section 5 by different extraction strategies. However, by using deep learning, the feature extraction from SMG images was performed automatically and much faster in the detection process comparing the other proposed methods. 2) The method could handle most of the scratched, damaged, and worn-out coin samples and raised above the other proposed methods in this matter. 3) The rejection option proposed in this method prevented the confusion of the classifier by outliers and poor-quality coins, while the samples participated in the training process. 4) The proposed method has an encouraging accuracy of 87.56 when we trained the model by only 20 Kroner 2008 and tested it by all types of coins together.	1) Unlike the methods proposed in Sections 3, 4, and 5, this method required a powerful system with GPU. Although we received the advantage of transfer learning for the proposed classifier, the process of generating fake coins by SMG images can take several weeks without a powerful GPU.

7-2 WHICH OF THE PROPOSED METHODS SHOULD BE APPLIED?

Since the proposed methods in this study have their advantages and disadvantages (Table 7.1), selecting one of the proposed methods as the best approach is an important question. Therefore, providing some tips will be useful for researchers who want to continue the research and industry to implement one of the methods to widen its usage.

Regarding the advantages and disadvantages of the proposed method stated above, the methods related to Chapters 3 and 4 are useful for Danish coins and no other types of coins. We preferred to use these methods for only Danish coin images if we do not have access to GPU and a powerful machine or scanner.

The method proposed in Chapter 5 is capable of detecting counterfeit coins of any type of coins and has remarkable accuracy. However, in case that we do not have fake samples for each type, the method cannot work properly.

The method proposed in Chapter 6 is capable of detecting counterfeit of any type of coins, surprisingly, any type even if we do not have any fake sample of it. Although the accuracy is not as high as the previous method in chapter 6, it can handle worn out and damaged coins much better. In conclusion, the best method for the future work can be the method proposed in Chapter 6. Since the deep learning methods are growing so fast, they can be very effective on the proposed method.

References

[1] L. Liu, Y. Lu, and C. Y. Suen, "An Image-Based Approach to Detection of Fake Coins," *IEEE Trans. Inf. Forensics Secur.*, vol. 12, no. 5, pp. 1227–1239, 2017.

[2] "The protection of Euro coins in 2017," 2017. [Online]. Available: https://ec.europa.eu/info/sites/info/files/economy-finance/2017_annual_report_euro_coins_public_final.doc.pdf.

[3] "Counterfeit coins cause for concern among Canadian collectors," 2017. [Online]. Available: https://canadiancoinnews.com/counterfeit-coins-cause-concern-among-canadian-collectors/.

[4] K. Sun, "Detection of Counterfeit Coins and Assessment of Coin Qualities.," Master thesis, Concordia University, Montreal, Canada, 2015.

[5] J. Toriwaki and H. Yoshida, *Fundamentals of three-dimensional digital image processing*. Springer Science & Business Media, 2009.

[6] C. Zhang, J. Xu, N. Xi, J. Zhao, and Q. Shi, "A robust surface coding method for optically challenging objects using structured light," *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 3, pp. 775–788, 2014.

[7] A. K. K. Al-Frajat, "Selection of Robust Features for Coin Recognition and Counterfeit Coin Detection," Ph.D. thesis, Concordia University, Montreal, Canada, 2018.

[8] J. Kim and V. Pavlovic, "Ancient coin recognition based on spatial coding," in *Pattern Recognition (ICPR), 2014 22nd International Conference on*, 2014, pp. 321–326.

[9] K. Sun, B. Y. Feng, P. Atighechian, S. Levesque, B. Sinnott, and C. Y. Suen, "Detection of counterfeit coins based on shape and letterings features," in *Proceedings of 28th ISCA International Conference on Computer Applications in Industry and Engineering, San Diego, USA*, 2015, pp. 165–170.

[10] L. Shen, S. Jia, Z. Ji, and W.-S. Chen, "Extracting local texture features for image-based coin recognition," *IET Image Process.*, vol. 5, no. 5, pp. 394–401, 2011.

[11] S. Kim, S. H. Lee, and Y. M. Ro, "Image-based coin recognition using rotation-invariant region binary patterns based on gradient magnitudes," *J. Vis. Commun. Image Represent.*, vol. 32, pp. 217–223, 2015.

[12] R. Huber, H. Ramoser, K. Mayer, H. Penz, and M. Rubik, "Classification of coins using an eigenspace approach," *Pattern Recognit. Lett.*, vol. 26, no. 1, pp. 61–75, 2005.

[13] A. K. Hmood and C. Y. Suen, "An ensemble of character features and fine-tuned convolutional neural network for spurious coin detection," *a chapter Lang. Process. Pattern Recognition, Intell.*

Syst., 2018.

[14]   M. S. Rad, S. Khazaee, and C. Y. Suen, "Counterfeit Coin Detection Based on Image Content By Fuzzy Association Rules Mining," in *ICPRAI 2018*, 2018, pp. 280–286.

[15]   M. Tresanchez, T. Pallejà, M. Teixidó, and J. Palacín, "Using the optical mouse sensor as a two-euro counterfeit coin detector," *Sensors (Basel).*, vol. 9, no. 9, pp. 7083–7096, 2009.

[16]   A. Gavrijaseva, O. Märtens, and R. Land, "Acoustic Spectrum Analysis of Genuine and Counterfeit Euro Coins," *Elektron. IR ELEKTROTECHNIKA*, vol. 21, no. 3, pp. 54–57, 2015.

[17]   S. Dalila and C. Mario, "3D face shape prediction from a frontal image using cylindrical coordinates and partial least squares," *Pattern Recognit. Lett.*, vol. 34, no. 4, pp. 389–399, 2013.

[18]   R. Hulik, V. Beran, M. Spanel, P. Krsek, and P. Smrz, "Fast and accurate plane segmentation in depth maps for indoor scenes," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 1665–1670.

[19]   R. Fan, U. Ozgunalp, B. Hosking, M. Liu, and I. Pitas, "Pothole Detection Based on Disparity Transformation and Road Surface Modeling," *IEEE Trans. Image Process.*, vol. 29, pp. 897–908, 2020.

[20]   Z. Jin, T. Tillo, w Zou, X. Li, and E. Lim, "Depth image-based plane detection," *Big Data Anal.*, vol. 3, no. 1, pp. 1–10, 2018.

[21]   E. Vera, D. Lucio, L. Fernandes, and L. Velho, "Hough Transform for real–time plane detection in depth images," *Pattern Recognit. Lett.*, vol. 103, pp. 8–15, 2018.

[22]   J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: The fuzzy c–means clustering algorithm," *Comput. Geosci.*, vol. 10, no. 2–3, pp. 191–203, 1984.

[23]   J. C. Bezdek, "Objective function clustering," in *Pattern recognition with fuzzy objective function algorithms*, Springer, 1981, pp. 43–93.

[24]   C. C. Aggarwal, *Data classification: algorithms and applications*. CRC press, 2014.

[25]   K. M. Ting and I. H. Witten, "Issues in stacked generalization," *J. Artif. Intell. Res.*, vol. 10, pp. 271–289, 1999.

[26]   Q. Tao, G.-W. Wu, F.-Y. Wang, and J. Wang, "Posterior probability support vector machines for unbalanced data," *IEEE Trans. Neural Networks*, vol. 16, no. 6, pp. 1561–1573, 2005.

[27]   F. Chollet, *Deep Learning with Python*, First. Shelter Island, NY, US: Manning, 2018.

[28]   Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient–based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[29]   A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C.

Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105.

[30]   K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in *International Conference on Learning Representations*, 2014.

[31]   K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[32]   A. G. Howard *et al.*, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," *ArXiv*, vol. abs/1704.0, 2017.

[33]   J. Tang, *Intelligent Mobile Projects with TensorFlow: Build 10+ Artificial Intelligence Apps Using TensorFlow Mobile and Lite for IOS, Android, and Raspberry Pi*. Packt Publishing Ltd, 2018.

[34]   I. Goodfellow *et al.*, "Generative Adversarial Nets," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 2672–2680.

[35]   R. Pugazhenthi, "Harris-Hessian Algorithm for Coin Apprehension," *Int. J. Adv. Res. Comput. Eng. Technol.*, vol. 2, no. 5, pp. 1689–1693, 2013.

[36]   R. Goldsborough, "Counterfeit Coin Detection," 2013. [Online]. Available: http://rg.ancients.info/guide/counterfeits.html.

[37]   Inksure Technologies, "Coin anti-counterfeiting," 2014. [Online]. Available: http://www.inksure.com/banknotesecurity/254-coin-anti-counterfeiting.

[38]   L. Van Der Maaten and E. Postma, "Towards automatic coin classification," in *In Proceedings of the EVA-Vienna*, 2006, pp. 19–26.

[39]   R. . Gonzales and R. . Woods, *Digital Image Processing*, Third Edit. Prentice Hall, 2008.

[40]   J. F. Kaiser, "On a simple algorithm to calculate the'energy'of a signal," in *International conference on acoustics, speech, and signal processing*, 1990, pp. 381–384.

[41]   S. Khazaee and K. Faez, "A novel classification method using hybridization of fuzzy clustering and neural networks for intrusion detection," *Int. J. Mod. Educ. Comput. Sci.*, vol. 6, no. 11, p. 11, 2014.

[42]   G. Holmes, A. Donkin, and I. H. Witten, "Weka: A machine learning workbench," in *Proceedings of ANZIIS'94-Australian New Zealnd Intelligent Information Systems Conference*, 1994, pp. 357–361.

[43]   D. Mehta and A. Sagar, "An efficient way to detect and recognize the overlapped coins using Otsu's algorithm based on Hough transform technique," *Int. J. Comput. Appl.*, vol. 73, no. 9, pp. 18–21, 2013.

[44]   R. S. Hassoubah, A. F. Aljebry, and L. A. Elrefaei, "Saudi riyal coin detection and recognition," in

2013 IEEE Second International Conference on Image Information Processing (ICIIP-2013), 2013, pp. 62–66.

[45]   S. Khazaee, M. Sharifi Rad, and C. Suen, "Detection of counterfeit coins based on modeling and restoration of 3D images," in *Lecture Notes in Computer Sceince, International Symposium Computational Modeling of Objects Represented in Images*, Niagara falls, USA: Springer, 2017, pp. 178–193.

[46]   E. Catmull and R. Rom, "A class of local interpolating splines," in *Computer aided geometric design*, Elsevier, 1974, pp. 317–326.

[47]   S. A. Travers, *The coin collector's survival manual*. Random House Digital, Inc., 2008.

[48]   S. Khazaee, M. Sharifi, and C. Suen, "Restoring height-map images of shiny coins using spline approximation to detect counterfeit coins," in *ICPRAI 2018*, 2018, pp. 383–387.

[49]   Z. Yao and W. Yi, "Curvature aided Hough transform for circle detection," *Expert Syst. Appl.*, vol. 51, pp. 26–33, 2016.

[50]   Z. Liu, H. Ukida, P. Ramuhalli, and K. Niel, *Integrated Imaging and Vision Techniques for Industrial Inspection*. Springer, 2015.

[51]   M. A. Fischler and R. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, 1981.

[52]   MathWorks, "Find Image Rotation and Scale Using Automated Feature Matching." [Online]. Available: https://www.mathworks.com/help/images/find-image-rotation-and-scale-using-automated-feature-matching.html.

[53]   M. Ahmed, A. G. Rasool, H. Afzal, and I. Siddiqi, "Improving handwriting based gender classification using ensemble classifiers," *Expert Syst. Appl.*, vol. 85, pp. 158–168, 2017.

[54]   N. Bishop and M. Christopher, "Pattern recognition and machine learning (information science and statistics) springer-verlag new york," *Inc. Secaucus, NJ, USA*, 2006.

[55]   B. Hanczar and E. R. Dougherty, "Classification with reject option in gene expression data," *Bioinformatics*, vol. 24, no. 17, pp. 1889–1895, 2008.

[56]   R. Quinlan, *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, 1993.

[57]   D. Aha and D. Kibler, "Instance-based learning algorithms," *Mach. Learn.*, vol. 6, no. 1, pp. 37–66, 1991.

[58]   C. M. Velu, P. Vivekanandan, and others, "Indian coin recognition system of image segmentation by heuristic approach and houch transform (HT)," *Int. J. Open Probl. Compt. Math*, vol. 2, no. 2, pp. 224–271, 2009.

[59]   R. P. Padhy, X. Chang, S. K. Choudhury, P. K. Sa, and S. Bakshi, "Multi-stage cascaded

deconvolution for depth map and surface normal prediction from single image," *Pattern Recognit. Lett.*, vol. 127, pp. 165–173, 2019.

[60] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[61] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei–Fei, "ImageNet: A Large–Scale Hierarchical Image Database," in *CVPR09*, 2009.

APPENDIX:

OTHER COLLABORATIONS

Apart from the proposed methods for counterfeit coin detection in this thesis analyzing 3-D height-map images, I cooperated in several other research solving the problem by 2-D image processing. I have also been studying and collaborating in some other deep learning-based projects that led me to propose a hybrid CNN based method described in Section 6. The related papers and their abstracts are listed below.

Appendix A: Other Articles Related to Counterfeit Coin Detection as the Second Author

Counterfeit Coin Detection Based on Image Content by Fuzzy Association Rules Mining

Sharifi M., Khazaee, S., Suen, C.

Abstract: In this paper, a new framework is proposed for counterfeit coin detection that shows the effectiveness of image mining techniques in this field. The proposed system is developed in two modules. In the first module, the segmentation of digital coin images is applied to find the region of interest (ROI). These ROIs serve as inputs to the next stage. In the next module, image mining is applied to find frequent image patterns present in coin images using fuzzy association rules mining. The experimental results show that fuzzy association rules mining based on image content is feasible and gives strong rules that can be further used for the effective classification of coin images.

A Blob Detector Images-Based Method for Counterfeit Coin Detection by Fuzzy Association Rules Mining

M. Sharifi Rad, S. Khazaee, L. Liu, and C.Y. Suen

Second International Conference on Pattern Recognition and Artificial Intelligence, ICPRAI 2020, Zhongshan City, China, In press.

Abstract: Image processing techniques using the knowledge obtained from known historical data has become recently one of the most intensively studied topics in decision science and computer science. This paper presents an automatic system for fake coins detection based on image content. In this study, a blob detector image-based method by fuzzy association rules mining is proposed to detect counterfeit coins. This method consists of two-stages. In the first stage, the original image dataset is preprocessed by a blob detector. This provides all frequent features that must be mined in the next stage. In the second stage, fuzzy association rules mining extracts the effective fuzzy rules and classifies the coin image data automatically. The performance of the proposed method has been compared with some other methods, and we demonstrate that our framework surpasses in terms of classification accuracy, which is a desirable level when compared with recent studies in this field. This research demonstrates the proposed framework is a reliable, intelligent detection system and can be utilized for other applications based on image content.

A Pruned Fuzzy Associative Classifier to Detect Image-Based Counterfeit Coins

M. Sharifi Rad, S. Khazaee, L. Liu, and C.Y. Suen

Pattern Analysis and Applications, (Under review)

Abstract: In this research, a novel framework is proposed for counterfeit coin detection that shows the effectiveness of image mining techniques. We develop an image miner system on top of the fuzzy concept that helps us to discover the implicit information from the images in the way closer to the human's viewpoint. Our proposed framework is developed in two modules, and the principal privilege of it is a compressed and white-box system that can be considered as a knowledge attainment tool. In the first module, a method to detect the region of interests (ROIs) is applied that focuses on blob detection. In the second module, image mining is applied to find image patterns present in coin images using fuzzy association rules mining. In this paper, we preserve the full power of fuzzy association rule mining to reduce the amount of redundant and insignificant rules by focusing on pruning methods. Furthermore, a new algorithm named Engine_*FAFS* for

feature selection and also a pruned based fuzzy associative classifier named Classifier $_{PFA}$, have been proposed to create a robust counterfeit coin detector system. By comparing the achieved results with some other methods obtained from the same dataset, we demonstrate that our framework surpasses in terms of lower feature dimensions, and smoother boundaries while maintaining satisfactory accuracy. In this study, the problem with a general form will be described to provide a common framework for appearing issues in other domains.