An Autoencoding Method for Detecting Counterfeit Coins

Iman Bavandsavadkouhi

A Thesis
In the Department of
Computer Science and Software Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of Master of Computer Science at
Concordia University
Montreal, Quebec, Canada

September 2022

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: Iman Bavandsavadkouhi

Entitled: An Autoencoding Method for Detecting Counterfeit Coins

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

MASTER OF SCIENCE (Computer Science)

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. Yuhong Yan

_________________________________ Examiner
Dr. Yuhong Yan

_________________________________ Examiner
Dr. Sudhir Mudur

_________________________________ Thesis Supervisor
Dr. Ching Y. Suen

Approved by

____
Dr. Leila Kosseim, Graduate Program Director

____
Dr. Mourad Debbabi, Dean of Faculty

September 9, 2022

Abstract

An Autoencoding-Based Anomaly Method for Detecting Counterfeit Coins

Iman Bavandsavadkouhi

In our daily lives, we use coins to pay for goods and services. However, the market for antique and historical coins is another place where coin quality and genuinity are important. Since counterfeiting has become more common as a result of technological advancements, dealing with fake coins is unavoidable. As a result, researchers have considered various methods in coin detection studies. In recent years, image-based coin detection has made extensive use of 2-D and 3-D image processing approaches. We propose a method for detecting counterfeit coins based on image content in this paper. We used SIFT, SURF, and MSER to determine the degree of similarity between our datasets. Then, using statistical analysis, we determine which descriptor is the most effective criterion for counterfeit coin detection. SIFT was chosen as the most reliable algorithm for the Danish and Canadian coin image dataset according to the Experiments' results. The autoencoder is then trained to detect anomalies in the coin images. A coin image is fed to the trained autoencoder as input and outputs a new image. Using the chosen criterion, the output image is compared to a baseline image. If the similarity between these two images is greater than a certain threshold, the coin is genuine. For training, most counterfeit coin detection methods require fake data. Our autoencoding-based anomaly method can eliminate this. Our proposed method for distinguishing genuine coins from counterfeit coins yielded promising results.

In addition, we present a method for increasing the speed of counterfeit coin detection. We conducted our research on the Mint Mark of Canadian toonies coin images

and we were able to achieve acceptable results by combining the edge detection technique with GAN and autoencoder.

Acknowledgment

I would like to express my deepest appreciation to my supervisor Dr. Ching Yee Suen, who made this work possible. It has been an honor to work with him and I am grateful to him for believing in my abilities. His valuable guidance, helpful advice, and encouragement carried me through all the stages of writing my thesis. Throughout these years at university, I have always been proud to work with Dr. Ching Yee Suen as a superstar professor in image processing and pattern recognition. This endeavor would not have been possible without his support.

I would like to thank all my colleagues, friends, and staff at CENPARMI at Concordia University for their assistance and motivation. I would like to extend my sincere thanks to Dr. Saeed Khazaee and Dr. Maryam Sharifi Rad for their support and helpful insight into my work. Their kindness is unforgettable. Special thanks go to Mr. Nicola Nobile, CENPARMI's research manager for his excellent technical support. Also, thanks to Ms. Phoebe Chan for all her help and consideration.

I would also like to express my gratitude to my committee for reading and evaluating this thesis. Their comments and feedbacks are valuable and highly appreciated.

I would like to thank the Natural Sciences and Engineering Research Council of Canada for supporting this research project

I want to express my heartfelt appreciation and gratitude to my parents and sibling, who always inspired me to be positive, independent, and strong. Even though the distance between us caused them to suffer, they always encouraged me to achieve this goal.

Last but not the least, I would like to extend my sincere thanks to my brother Ehsan and his wife Mehrnoosh for their incredible support throughout my journey. Their kindness will always remain in my heart.

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

1.1 Motivation

Coins are an integral component of financial transactions and everyday social interactions. Although electronic payment mechanisms are widely available, coins are still extensively used in a variety of locations, especially retail markets, vending machines, parking meters and ticket sales centers. Additionally, a growing number of businesses, museums, and government agencies are requesting automatic coin classification systems to differentiate rare, historical, common coins and worn out coins.

There are numerous counterfeit coins available on the market today. Considering there is always the possibility that forgers will create a counterfeit coin for any common coin on the market, finding counterfeit coins in our pockets is not surprising. Furthermore, numerous coin collectors faced counterfeiting during their professional careers, as many illegal producers have counterfeited common and precious coins, wreaking havoc on the coin market and society [1]. Between 2013 and 2017, the Counterfeit Monitoring System (CMS) reported that 837,910 counterfeit Euro coins were confiscated in Europe, totaling 1,330,401 Euros. In terms of antique coins, the coin counterfeiting business generates billions of dollars annually throughout the world [2]. CBC News reported on the existence of a counterfeit coin in January 2022. According to this news, Ontario Provincial Police have warned residents to be cautious with their pocket change after discovering a fake two-dollar coin in a store. There are likely to be more fake toonies on the market [3].

The procedure for producing a coin is to use a die with a high-pressure stroke or a coining press machine pressing a piece of round-shaped metal. The term "blank" refers to this round-shaped metal. The counterfeit die is created by molding the genuine coin stamp. They cut a piece of metal roughly the same width as the genuine coin to create a blank and then create a counterfeit coin bearing an eye-catching fake die. Figure 1.1 demonstrates common parts used for coin forging. The coins produced using this method always have a slight shape difference from the original coins, making identification difficult at a first glance.

In previous methods for detecting counterfeit coins, detection was based on the physical characteristics of the coins. These techniques concentrated on dimensions, weight, diameter, thickness, electromagnetic, and frequency characteristics, among others. Thanks to rapid advancements in technology, counterfeit coins are remarkably similar to genuine ones. As a result, the traditional method could easily be replicated. For example, a counterfeiter can use a Computer Numerical Control (CNC) machine to create a counterfeit coin that resembles a genuine coin. As a result, computer vision and image processing techniques have been considered and recent studies have suggested several novel methods for detecting counterfeit coins. These methods are more systematic, more affordable, and more accurate. It is worth noting that coin recognition research is distinct from the field of counterfeit coin detection research. In coin recognition, the system detects only the type of coin. This method requires the fewest features to accurately identify the type of coin. In counterfeit coin detection, the system should recognize each feature of the coin's surface by investigating it more precisely, while comparing it to other coins of the same type to determine whether the coin is genuine or not. As a result, this method must be more exhaustive in terms of coin surface details. Figure 1.2 illustrates some samples of genuine and fake coins.

(a) (b)

Figure 1.1. Two important parts of coin forging

(a) A fake die that is created by the forgers, and (b) A kind of coin press that is employed by the forgers to load fake dies and strike counterfeit coins.

(a) (b)

(c) (d)

Figure 1.2. Sample of genuine and fake coins

(a) Genuine Canadian two-dollar coin, (b) Fake Canadian two-dollar coin, (c) Genuine Danish 1996, and (d) Fake Danish 1996.

1.2 Objective

Due to the increasing demand for automated counterfeit coin detection techniques, image-based coin recognition has become critical in counterfeit coin detection [4]. Moreover, a sizable number of images have been produced in multimedia over the last few years. As a result, a growing number of experts are concentrating their efforts on developing a computer-assisted method for extracting estimable data from digital images [5]. Additionally, numerous research projects in the field of numismatics have been conducted to detect counterfeit coins, and several novel methods for counterfeit coin detection have been proposed in [4, 6]. Typically, many automatic fake machine detectors in coin detection are supplied by a preliminary mechanism for measuring a different aspect of the coin's characteristics [7]. However, when the physical characteristics of counterfeit and genuine coins are identical, these technologies are unable to distinguish them. In recent years, the literature has been flooded with studies on image-based methods for detecting counterfeit coins [3, 5, 8]. Besides, researchers are utilizing image processing techniques to extract features from coin images via texture analysis [9, 10]. As we all know, working with noisy and corrupted images, such as rust, dust, or sulfated coin images, is not a good idea. Members of the Centre for Pattern Recognition and Machine Intelligence (CENPARMI) lab at Concordia University in Montreal, Quebec have developed several novel methods for detecting counterfeit coins in recent coin detection studies. They recently advanced coin detection from two-dimensional to three-dimensional image processing. They used a 3D scanner to scan and model the number of coins in references [4, 11], capturing height and depth rather than color levels. Although 3D scanning is resistant to the low-quality coins mentioned previously, the lengthy processing time remains a significant disadvantage. The possibility of using an autoencoder to detect counterfeit anomaly coins has not been investigated, and this remains an open area of research. The authors of [12] used an autoencoder to detect coins and concentrated their research on coin weights.

We present a method for determining the degree of similarity between coin images in our datasets in order to detect counterfeit coins in this article. Furthermore, we are proposing a method for detecting counterfeit coins without the use of fictitious ones for training.

1.3 Challenges

During the study, we encountered several serious challenges. The following are a few of the most significant obstacles in conducting this research:

Size of coins: The small size of the coins is one of the primary reasons that distinguishing genuine from counterfeit coins is difficult since this differentiation occurs in centimeter-scale ranges. Also, distinguishing between small components of coins, such as the letter or symbol on the coin's surface, requires a higher level of sophistication. Thus, expert knowledge is required to distinguish genuine coins from counterfeit ones. The lack of access to appropriate tools for precisely measuring the size of coins to distinguish genuine from counterfeit coins is a problem that we face.

Well generated counterfeit coins: As previously stated, as technology advances, coin forgery becomes more sophisticated than ever. Not only has this made it more difficult for the public to detect fake coins, but it has also made it more difficult for skilled experts to distinguish genuine from fake coins in some cases. This evolutionary process in the production of counterfeit coins is accelerating. This issue has resulted in significant economic costs and has a negative impact on communities' collective feelings. As a result, research in the field of counterfeit coin detection has become significant.

Uncommon feature of fake coins: There are several methods for fabricating coins. As previously stated, forgers frequently create counterfeit coins by molding the original coin stamp into a fake die. Numerous counterfeit coins are manufactured in multiple locations at various factories or workshops. As a result, counterfeit coins and genuine coins are not always produced from the same location. Counterfeiters use a variety of different dies to strike their coins or employ a variety of different methods to fabricate a particular type of coin. As a result of these distinctions in the source of coin counterfeiting, we encounter distinct characteristics in imitation coins. These dissimilar styles present a significant obstacle to research in this field of study.

Inadequate data for counterfeit coins: Despite the widespread existence of counterfeit coins worldwide, obtaining comprehensive and labeled counterfeit coins is exceedingly difficult. It is against the law in many places to keep counterfeit coins or their images. For researchers, conducting research in the counterfeit coin detection field without using fictitious data presents a challenge. As a result, we are especially grateful to the Danish law enforcement agency for granting us access to their collection of counterfeit and genuine coins. Moreover, we used deep learning techniques in this research by employing the Generative Adversarial Neural Network (GAN) to generate counterfeit coin images to augment the size of our fake dataset.

1.4 Contribution

This thesis proposes a novel image-based method for autoencoder-based counterfeit coin detection. As for the method proposed in this study, we provide a detailed explanation of the major contributions made by our work in the following paragraph.

To begin with, we evaluate three well-known algorithms for feature extraction in counterfeit coin detection studies using our dataset: Speed Up Robust Features (SURF), Maximally Stable Extremal Regions (MSER), and Scale-Invariant Feature Transform (SIFT). At this point, we extract features from each coin image in our dataset and then perform pairwise comparisons of the images. We specified a range for each comparison based on the degree of similarity between images in each distinct dataset. By performing statistical analysis on this data, we can determine which algorithm has the best performance in our dataset and should be used in the subsequent level of study. At the next level of study, we employ the autoencoder to detect counterfeit coins. We develop a model for encoding and decoding the image of a coin. We propose a method for successfully distinguishing fake coins from genuine coins in this study by utilizing reconstruction in the autoencoder. Based on the approach proposed in this thesis, we suggest a method for training a model using an autoencoder without the use of fictitious data.

Furthermore, we utilize a Generative Adversarial Neural Network (GAN) to generate counterfeit coin images for Canadian toonies to compensate for the lack of fake coin images for testing the proposed system.

1.5 Thesis Outline

The remainder of this thesis is organized as follows:

In chapter 2, a thorough discussion of related research in the field of coin detection and the existing research studies on counterfeit coin detection are discussed. In Chapter 3, methods for preprocessing images are discussed. In chapter 4, we investigate three image comparison descriptors: MSER, SURF, and SIFT to choose one of them as a criterion by statistical analysis. In chapter 5, we design the autoencoder and propose our model for encoding and decoding the images. We talk about the procedure of training the model. Also, we show how the proposed model detects counterfeit coins. In section 6, we examine our proposed method in particular parts of Canadian toonies, and the counterfeit samples were successfully detected. Also, we take advantage of GAN in generating the fake sample in this chapter. Finally, the study is concluded with an overview of the main contributions of this work, and an outline of future work in chapter 7.

Chapter 2

Literature Review

2.1 Counterfeit Coin Detection

Coins have been used to pay for goods and services for centuries. Coins are still an integral part of our lives, even with the advent of credit cards. The coins are accepted in a variety of locations, including stores, gas stations, and ticket booths. Counterfeit coins can have a significant negative impact on the economy if there is a large number of them. Recognizing counterfeit coins has become significantly more difficult in the Eurozone in recent years. According to Royal Mint estimate in 2014, approximately 3.04 percent of all £1 coins distributed in the UK are counterfeit [2]. They chose to exchange these coins for new ones. Recently, CBC News reported the existence of counterfeit toonies in one of Canada's provinces [3]. Although the Royal Mint of Canada and the Government of Canada provided an authentication system for identifying genuine gold and silver collection coins, the identification of counterfeit coins as regular coins remains a significant concern. As a result of the critical nature of this issue, numerous governments have taken significant steps to prevent the production of additional counterfeit coins by using new technologies and to support researchers in this field of study. Given the high value of ancient coins, museum owners and collectors are urged to exercise greater caution when selecting coins for their collections and to employ new technologies to detect counterfeit coins.

Since ancient times, counterfeit coins have been produced. Today, counterfeit coins continue to be manufactured through the use of new technologies. What all counterfeit coins have in common is that their edges are shaped and appear irregularly. While the detection of counterfeit coins has traditionally been done manually by experts in this field. With the advancement of technology and the use of scientific methods, it has been demonstrated that this method has limitations in certain situations. Numerous studies have been conducted to determine the difference between counterfeit and genuine coins. Previously conducted research has focused on coin's electromagnetic, frequency, and physical characteristics. Typically, this specification is the most frequently used feature for detecting counterfeit coins by vending machines, game machines, and parking meters. These machines operate via electromagnetic systems and authenticate coins by using X-ray fluorescence. The patent authors [13] proposed a method for coin discrimination based on an electromagnetic method. The authors employ sensors to determine the characteristics of coins in such a way that their patent includes a mode for setting reference values and a mode for discrimination. In the first mode, data collected from sampling coins is statistically processed to determine the minimum and maximum reference values, which are then stored in memory. Besides, a patent [14] proposes an electromagnetic method for detecting counterfeit coins by passing a signal through a coin using an oscillation coil. When coins are recognized using the discrimination mode, their characteristics determine whether they fall within the range of minimum and maximum reference values. They automated the process of separating genuine coins from counterfeit coins using this method. Other methods for detecting counterfeit coins have been introduced recently as a result of advancements in technology and image processing methods. Numerous theories have been proposed for counterfeit coin detection, with some emphasizing image-based detection and others emphasizing non-image-based detection. Typically, studies are conducted in a variety of areas, including the shape of coins, the position of letters, and the position of numbers. This section discusses coin detection and counterfeit coin detection methods. Tresanchez et al. proposed a method for detecting counterfeit two-Euro coins using an optical mouse sensor in reference [7]. The authors captured an image of a small portion of the coins under analysis using an optical mouse sensor and comparing it to a small portion of coins used as a reference in this work. The primary advantage of this proposed method is its low cost, but the primary disadvantage is that they only use a small portion of the coin (fourteenth) to establish the method. Also, the proposed method's accuracy is compromised by worn and scratched coins.

The authors of [9] proposed a method for detecting counterfeit Danish coins based on the characteristics of the coin images. The feature extraction method was used to investigate the width, height, and stroke width of letters in this study. Additionally, the angle between the characters was considered in this study. While the outcome of this research was astonishingly accurate, the dataset used was quite small. On the other hand, the method does not work with any type of coin due to the type and shape of the letters. Also, damaged and noisy coin images were not suitable candidates for the proposed method. Maryam et al. [15] developed a blob detector image-based method for automatically detecting counterfeit coins using fuzzy association rules mining. Two steps were taken to implement the proposed method. They began by preprocessing the original coin images with a blob detector to ensure that all features were available for extraction in the subsequent stage. The following step involves the extraction of effective fuzzy rules via fuzzy association rules mining and the classification of coin image data automatically. Although this system is extremely sensitive, it is powerless to reject degraded coins, and thus classified them randomly in some cases.

CENPARMI members at Concordia University in Montreal have developed several novel methods for detecting counterfeit coins in recent years. They used a 3D scanner to scan and model the coins, capturing height and depth rather than color levels. In [4], the authors extract height and depth information from image coins using a 3D scanner in order to differentiate genuine and counterfeit coins. They reduce the complexity of their proposed method by converting a circular coin image to a linear rectangular image through the use of straightening algorithms. Also, they signal-processed the coin images to address the issue of shiny coin images. The authors of [11] proposed a three-dimensional image-based approach for examining the *precipice-borders* of coin surfaces and trained an ensemble classification system to extract critical features from coin images. Furthermore, the authors demonstrate the efficacy of 3D scanning in detecting counterfeit coins and create a height-map image dataset. Although 3D scanning is resistant to the low-quality coins mentioned previously, the lengthy processing time remains a significant disadvantage. Besides, access to 3D scanners is costly and requires the expertise of specialized professionals.

2.2 Image Processing

Today, image processing is one of the most widely used technologies in the world, with applications in a variety of fields including industry, medicine, urban planning, security, science, and computer science. In image processing, an image is taken as input and transformed through a series of operations to produce a new output image. These procedures may alter the output in a variety of ways, including improving the image's quality, compressing the image, or extracting specific features from the image. The following are the steps involved in image processing to enhance image quality:

- **First:** The desired input, which is an image, is obtained by devices such as a camera, optical scanner, or digital sensor.
- **Second:** After receiving input an image is preprocessed if required in different ways such as noise reduction, filtering, or segmented to several parts.
- **Third:** The image is analyzed and manipulated.

- **Fourth:** In the final stage, using the result obtained from image analysis, the image is processed, which image can be altered or transferred to another form as a report.

In image-based coin detection studies, we are always confronted with rust, dust, sulfate, and a variety of other factors that affect the coin's quality. Thus, image processing has been used in a variety of ways in several coin recognition systems. Image filtering is a frequently used processing technique in coin detection for enhancing or altering the appearance of images. The authors of reference [15] state that they require brighter images from lower-quality coin images in order to extract feature descriptors for their study. As a result, they applied a median filter to the coin images contained in the dataset. Meanwhile [11], the authors encountered a significant loss of height information due to degradation and shadowing on shiny image coins following scanning. They overcame this obstacle by employing a high-pass filter in their study. Segmentation of images is also frequently used in counterfeit coin detection research. Image segmentation divides an image into several image segments in order to reduce complexity and increase calculation speed during the image processing and analysis process. The most frequently used image segmentation techniques in coin detection studies include the following: *Hough transformation,* which focuses on extracting a specific shape from an image [8, 17, 18]; *Edge-based segmentation*, which employs edge detectors to determine the boundaries of objects within images [19]; and *Ant Colony optimization*, which is a technique for selecting a portion of an image with a flexible shape [20]. Jain et al. [8] assessed the ability of edge-based segmentation and Hough transformation methods for ancient coins to differentiate. According to their research, the edge-based method performs better because the shapes of ancient coins are not circular. On the other hand, the *Hough transformation* technique works better because modern coins have a more regular shape.

2.3 Feature Extraction

Feature extraction is a critical process that is frequently used in coin recognition research. In images, features elements are parts or patterns of an object that aid in its identification. Following image processing, feature extraction methods are used to extract the relevant feature for classifying and recognizing images. SURF, MSER, and SIFT were used to extract features in this study. Additionally, we used these descriptors to compare images for our research in this thesis.

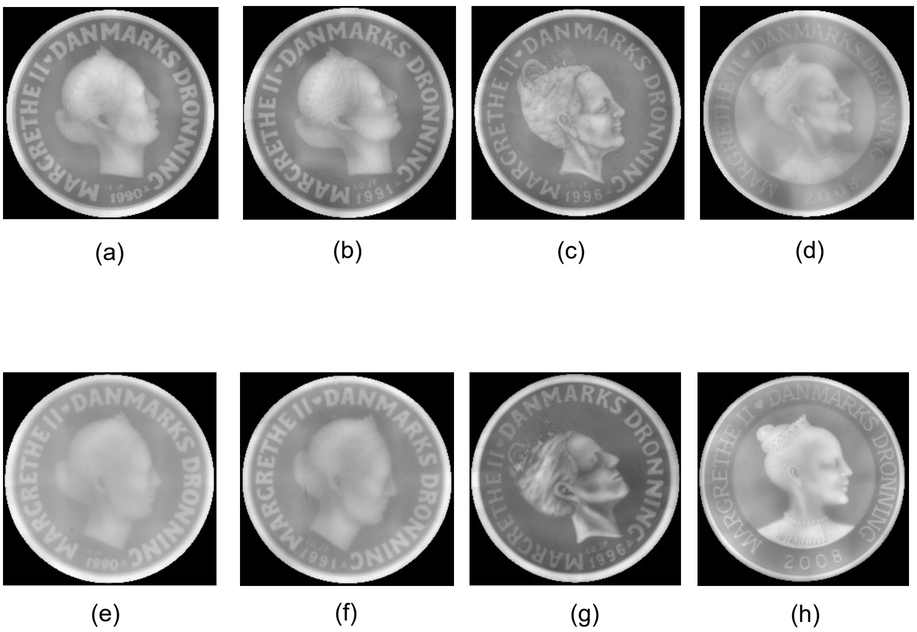
Speed Up Robust Features (SURF) is a feature extraction algorithm that Herbert Bay et al. proposed in 2006 [21]. In [22], the authors proposed an automatic coin recognition system based on machine vision by extracting the keypoint features from ancient Roman coins using the SURF algorithm. *Maximally Stable Extremal Regions* (MSER) is also a feature extraction algorithm proposed by Matas et al [23]. The authors of [9] used MSER to detect holes and indentations found on coin images. Moreover, they used this descriptor as an affine invariant feature when comparing test coin images to those in their dataset. In coin detection research, the *Scale-Invariant Feature Transform* (SIFT) algorithm is the best well-known feature extraction algorithm. David G. Lowe invented this algorithm in 2004 for the purpose of detecting, describing, and matching local features in images [24]. SIFT descriptor has been extensively used in recent years in several studies [18, 22, 25, 26, 27]. For instance, the authors of [25] employed SIFT to extract features from ancient coins. They reduced the complexity of their study by removing low-contrast keypoints or those located on the edge of coins using SIFT keypoints. Then they used the SIFT descriptor to compare keypoints between two images of the coin.

Chapter 3

Dataset Preparation and Image Preprocessing

3.1 Dataset Preparation

In this study, we used the CENPARMI Danish coin dataset [16], which includes Danish coins from 1990, 1991, 1996, and 2008, as well as Canadian toonies from 1996, 2009, 2011 and 2013. All the Danish coins are provided by the Danish police to a local company, with both genuine and fake coins to the CENPARMI lab. The images of these coins are captured by a high accuracy scanner with a robust infrastructure capable of scanning images with greater detail and quality. The machine includes a built-in microscope and five groups of adjustable LEDs that can provide users with remarkably high-resolution images from various angles. *Ultra Electronics Forensic Technology Ltd. Co* in Montreal owns the device patent. Figure 3.1 shows some coin images scanned by the IBIS TRAX.



(i) (j) (k) (l)

Figure 3.1. Danish and Canadian coin images scanned by IBIS TRAX.

(a) Genuine Danish Kroner year 1990, (b) Genuine Danish Kroner year 1991, (c) Genuine Danish Kroner year 1996, (d) Genuine Danish Kroner year 2008, (e) Fake Danish Kroner year 1990, (f) Fake Danish Kroner year 1991, (g) Fake Danish Kroner year 1996, (h) Fake Danish Kroner year 2008, (i) Genuine Canadian year 1996, (j) Genuine Canadian year 2009, (k) Genuine Canadian year 2011, and (l) Genuine Canadian year 2013.

3.2 Image Preprocessing

Preprocessing is one of the most important requirements during project implementation. Since most data are not pure, it should be preprocessed to make it suitable for specific future processing. We can improve the quality of coin images and increase the feature extraction reliability using image preprocessing. Preprocessing also helps to reduce noise and distortions in coin images, allowing for more accurate analysis. Furthermore, preprocessing can be applied to an image in a variety of ways, such as rotation, resizing, flipping, noise reduction, and changing the brightness. We encountered many noisy and degraded coins in our dataset; therefore, to improve image quality and manageability, we preprocessed coin images to better fit each step of our project. For coin image preparation, we employ the following preprocessing technique.

3.2.1 Data Cleaning and Resizing

When working with data, there is always the possibility of encountering noise, inaccuracies, defects, and errors that cause the dataset to be inconsistent. A dataset is never completely ready to be processed. Therefore, to have a perfect dataset, some action must be taken. Every scanned coin in this study was not in perfect condition. Some coins were completely damaged and degraded to the point. For example, the edge of some coins was completely worn out, and the image was not in perfect condition for the processing and feature extraction in the following steps. We remove some contaminated coin images from the dataset because we want to take the best result from our proposed method and have an ideal training set. In contrast, all coin images in the provided dataset were captured at high resolution. The original size of the Danish coin was 3550*3550, while the resolution of the Canadian coin was 2976*2976. Working with these dimensions is time-consuming and requires a lot of memory. As a result, to improve processing time and lack of memory, we reduced the size of image coins to 600*600 for Canadian coins. Considering the coin images were captured manually, they were not all of the same size. By reducing the dimension to a manageable size, we could solve these issues.

3.2.2 Transformation of the RGB Images To Grayscale

Reducing the complexity of a processing step is an important consideration. Color images contain more information than grayscale images, which complicates the proposed method. RGB images, on the other hand, are not required for some processing steps in this study, for example, MSER feature extraction descriptor, the input image should be stated in grayscale. As a result, in some steps of this study, we converted all RGB coin images to grayscale to simplify the algorithms and reduce computational requirements.

Figure 3.2. Colored image and grayscale image.

3.2.3 Image Restoration

As previously stated, we are dealing with some degraded coin images in our Danish dataset. These coin images have high brightness, which causes them to shine, and they are not suitable to be fed to the system. This issue makes it difficult to extract features from coin images in the next step of the research. To address this issue, we preprocess the coin image to improve its quality. Figure 3.3 shows a sample of a degraded coin on the left and a coin image after preprocessing on the right. First, we use a nonlinear operation to reduce the luminance of the image using the gamma correction method. Then, we use a sharpening filter to enhance the edges of objects on the coin images, making it easier to find the key points. The output image from the Gaussian and Bilateral filters is then passed. Using this method, we improve image quality to have better functionality for the feature extraction process in the next stage of the study.

(a) (b)

Figure 3.3. (a) Degraded Danish coin image, and (b) The same image after restoration

3.2.4 Generative Adversarial Network (GAN)

A generative adversarial network (GAN) is a machine learning model with two neural networks competing to improve their prediction accuracy. The GAN is one of the most well-known deep neural networks for creating counterfeit coins from original image datasets [26]. Figure 3.4 depicts the overall structure of the proposed GAN for generating fake samples. Based on the training set, this approach learns to generate new data of the same statistics as the training set. For example, a GAN trained on images can generate new images that appear to be at least partially true with many realistic characteristics. To establish GAN, we must provide the generator with a dataset of the desired image. Two models are trained concurrently during this process, which is accomplished through adversarial learning. At this point, the generator model is learning to produce images that are the most similar to those of its peers. On the other hand, the discriminator learns to look for differences between these two images in order to determine which is fake and which is real. As a result of compete between, the generator network and the discriminator network new images are generated by the machine. These generated images have a striking resemblance to real-world images [11]. At this point, we have found some images that are similar to the main images in our datasets.

As previously stated, one of our major challenges in this research is the scarcity of counterfeit coins. One of our primary research interests in this study is Canadian toonies. As a result, we need fake toonies coin images for our dataset. Even if the police department reported the presence of counterfeit Canadian toonies in the market [3], but we do not have access to these coins for our research. To address this issue, we used GAN to generate fake coin image samples for testing our proposed method. We fed the system several Canadian toonies coin images that are of the same shape, year, and size to train the model. Since we do not have counterfeit samples for Canadian toonies, we use data augmentation such as Flipping and Rotating to use them as fake input images to the system. Otherwise, we will not be able to obtain the most accurate result. Figure 3.5 shows a selection of input toonies images. In order to achieve the best results, we create the dataset with the highest quality coin photos. Figure 3.6 shows an example of a GAN-generated fake image of a coin.

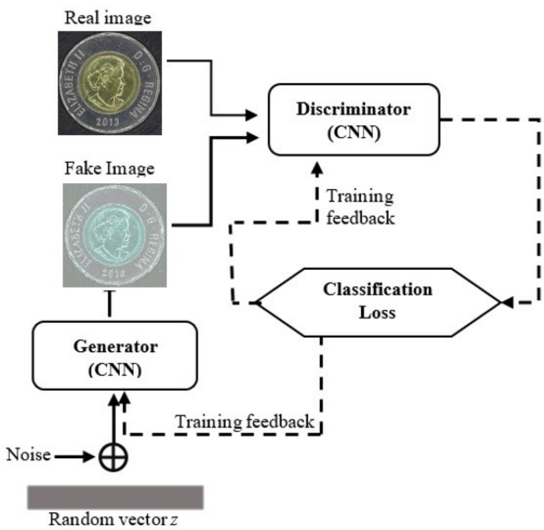


Figure 3.4. Fake coin generator.

Figure 3.5. A selection of Canadian toonies images which are provided as input to the GAN.

Figure 3.6. Fake coin images generated by GAN.

Chapter 4

Feature Extraction and Selecting Similarity Criterion

In this section, we investigate image similarity using certain well-known feature extraction descriptors such as, MSER, SURF and SIFT. We determine the degree of similarity by these three descriptors between the images of our dataset coin images. We build a module to rank each comparison to find similarity criteria in the Danish and Canadian coin image datasets. We placed them in separate tables to have a numerical dataset for Statistical analysis to select the most fit descriptors as criterion for our dataset to use it in next level of study. It is worth to mention that we are not using the mentioned descriptors for recognizing the counterfeit coin in this study.

4.1 Feature Extraction Descriptors and Comparison of Images

Feature extraction is a critical process that is frequently used in coin recognition research. In images, feature elements are parts or patterns of an object that aid in its identification. Following image processing, feature extraction methods are used to extract the relevant feature for classifying and recognizing images. SURF, MSER and SIFT were used to extract features in this study. Additionally, we used these descriptors to compare images for our research in this thesis.

4.1.1 Image Comparison with MSER

Maximally Stable Extremal Regions (MSER) is one of the most common methods of feature detection in image processing. This method extracts a large number of corresponding image elements and contribute to the wide-baseline matching, and it has led to better stereo matching and object recognition algorithms [27]. The MSER detector incrementally steps through the intensity range of the input image to detect stable regions in such a way that, the ThresholdDelta parameter determines the number of increments that the detector tests for stability. MSER works based on the concept of watershed, binarizes the image. Therefore, the result of the comparisons is grayscale. Figure 4.1 demonstrates an example of image comparison by MSER on Canadian toonies.

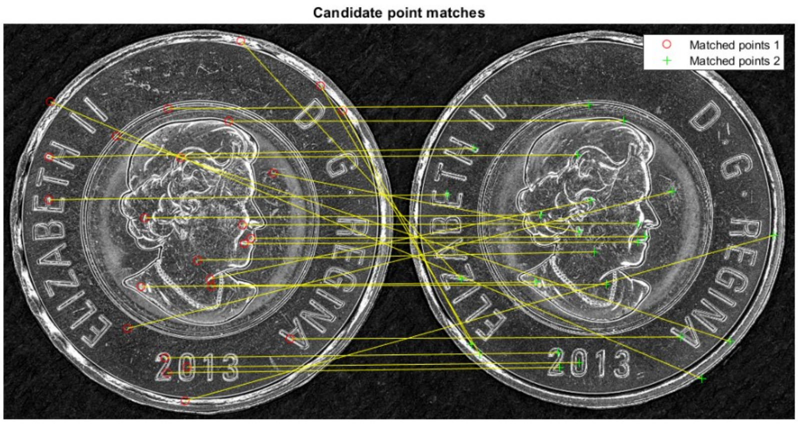


Figure 4.1. Applying MSER for comparison of a pair of Canadian toonie images.

4.1.2 Image Comparison with SURF

SURF is a patented local feature detector and descriptor. According to the SURF algorithm, the goal is developing both descriptor and detector which are faster to compute during the process. To succeed in their purpose, they must strike a balance between the above requirements, like reducing the descriptors dimension and complexity, while keeping it sufficiently distinctive. Figure 4.2 demonstrates the result of the comparison of a pair of images by SURF in the Canadian dataset.

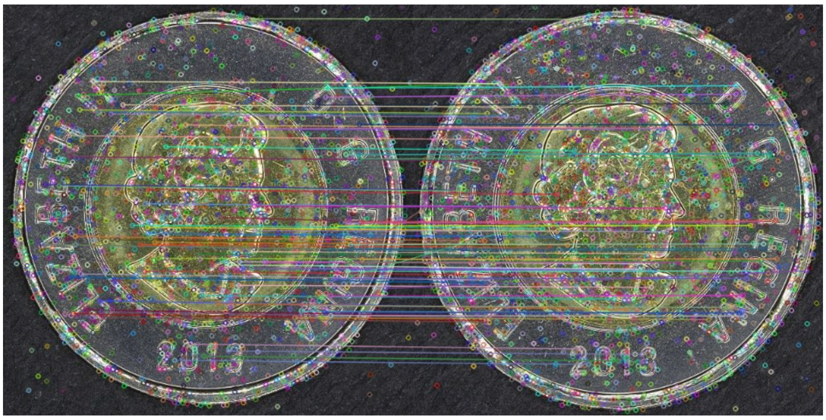


Figure 4.2. Applying SURF for comparison of two Canadian toonie images.

4.1.3 Image Comparison with SIFT

SIFT algorithm was proposed by D. Lowe in [24]. It has been widely used for the extraction of image local features owing to its distinctiveness and robustness. SIFT algorithms follow four stages to generate a set of image features. These stages are respectively called: 1. Scale-space extrema detection, 2. Keypoint localization, 3. Orientation assignment and Keypoint descriptor. Then, SIFT algorithms transform Image data into scale-invariant coordinates relative to local features. It is worth mentioning that SIFT descriptor is the most common algorithm employed for feature extraction in coin detection. A pair of images from the Canadian dataset was compared using SIFT in Figure 4.3.

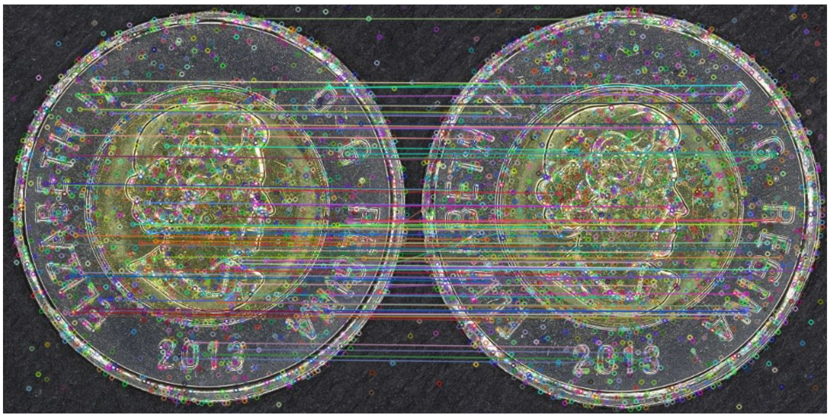


Figure 4.3. Applying SIFT for comparison of two Canadian toonie images.

4.2 Procedure of Image Comparison in Dataset

In this section, we produce data from MSER, SIFT, and SURF descriptors for coin images in our datasets through pairwise comparison to achieve a criterion for coin similarity. In this study, Danish 20 Kroner coins dataset that includes four types made in 1990, 1991, 1996, and 2008 is used and Canadian toonies coins dataset including those from 1996, 2009, 2011 and 2013. As a first step to select similarity criterion, a pairwise comparison is made among all images in our dataset. The first image of our dataset is compared with all other images in the dataset and the second image is compared with all other images. Similarly, we continue this process for all the remaining images. For a letter understanding of this step, we show the process in Figure 4.4 that represents how a pair of images is made for comparison in descriptors.

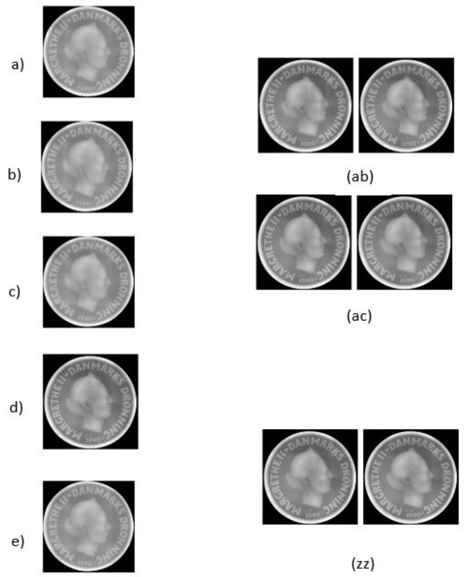


Figure 4.4. The process of choosing pairs of images for comparison. (ab) Shows the comparison of the first image of the dataset with the second image of dataset. (zz) shows the last comparison between two images in a dataset.

According to our preliminary studies, the value produced from each descriptor is different in scales and it is better to be normalized. The values of all three comparisons from the descriptors are normalized to the range (0, 1) using the following equation, where α is the similarity between two images and α′ is the normalized value.

$$\alpha' = \frac{\alpha - \min(\alpha)}{\max(\alpha) - \min(\alpha)}$$

Tables 4.1 to Table 4.11 demonstrate the values for comparing the similarity of one pair of images in terms of MSER, SIFT, and SURF descriptors for genuine and counterfeit coins, in different datasets respectively. We prepare more information regarding these values in the following step to observe how these values will lead us to choose the most appropriate descriptor for the next level of study. The number of pairs of images depends on the number of datasets in each year. Table 4.13 shows the number of images and pairs of images in each separate year for both genuine and counterfeit coins. For example, with 22 images in the 2008 genuine Danish coins dataset, we will have 231 pairs of images. As we mentioned earlier, based on how the pairs of images are selected, the number of pair of images, n', is obtained using the following equation, where n is the number of images in the dataset.

$$n' = \frac{n(n-1)}{2}$$

Table 4.1. The values for comparing the similarity of pair of images for 1990 Danish genuine coins.

Number of pair of images	MSER	SURF	SIFT
1	0.232758	0.253901	0.319148
2	0.369601	0.566275	0.468085
3	0.473576	0.630941	0.702127
4	0.646153	1	0.851063
…	…	…	…
1274	0.526289	0.678879	0.702127
1275	0.269407	0.293585	0.276595

Table 4.2. The values for comparing the similarity of pair of images for 1991 Danish genuine coins.

Number of pair of images	MSER	SURF	SIFT
1	0.152321	0.347998	0.180327
2	0.524666	1	0.508196
3	0.403871	0.478260	0.573771
4	0.422523	0.691461	0.721311
…	…	…	…
1274	0.262463	0.399150	0.377049
1275	0.200564	0.229508	0.456187

Table 4.3. The values for comparing the similarity of pair of images for 1996 Danish genuine coins.

Number of pair of images	MSER	SURF	SIFT
1	0.131688	0.293478	0.208433
2	0.263582	0.377622	0.337349
3	0.060051	0.137918	0.172289
4	0.155112	0.271604	0.168674
…	…	…	…
4004	0.189599	0.461971	0.301204
4005	0.241887	0.164062	0.277108

Table 4.4. The values for comparing the similarity of pair of images for 2008 Danish genuine coins.

Number of pair of images	MSER	SURF	SIFT
1	0.190476	0.095238	0.380952
2	0.222222	0.259259	0.333333
3	0.125435	0.341463	0.142857
4	0.142857	0.2	0.238095
…	…	…	…
230	0.190476	0.311111	0.238095
231	0.144578	0.168674	0.3

Table 4.5. The values for comparing the similarity of pair of images for 1990 Danish fake coins.

Number of pair of images	MSER	SURF	SIFT
1	0.065028	0.479243	0.461538
2	0.043478	0.495652	0.205128
3	0.133647	0.391080	0.435897
4	0.083612	0.530473	0.358974
…	…	…	…
299	0.103448	0.381191	0.615384
300	0.080645	0.702909	0.666666

Table 4.6. The values for comparing the similarity of pair of images for 1991 Danish fake coins.

Number of pair of images	MSER	SURF	SIFT
1	0.227642	0.322493	0.823529
2	0.171717	0.141414	0.5
3	0.158333	0.291666	0.558823
4	0.075409	0.229508	0.676471
…	…	…	…
90	0.098361	0.267759	0.176471
91	0.269841	0.407407	0.5

Table 4.7. The values for comparing the similarity of pair of images for 1996 Danish fake coins.

Number of pair of images	MSER	SURF	SIFT
1	0	0.075520	0.057142
2	0.310160	0.344212	0.457141
3	0.019489	0.054072	0.028571
4	0.171452	0.163094	0.2
…	…	…	…
44	0.130630	0.084567	0.171428
45	0.446682	0.228795	0.371428

Table 4.8. The values for comparing the similarity of pair of images for 2008 Danish fake coins.

Number of pair of images	MSER	SURF	SIFT
1	0.458598	0.634667	0.782608
2	0.187050	0.533658	0.565217
3	0.240343	0.332618	0.608695
4	0.253961	0.333901	0.347826
…	…	…	…
6440	0.092165	0.204081	0.217391
6441	0.191387	0.201298	0.434782

Table 4.9. The values for comparing the similarity of pair of images for 1996 Canadian toonies.

Number of pair of images	MSER	SURF	SIFT
1	0.618737	1	0.684615
2	0.618737	1	0.684615
3	0.593007	1	0.669231
4	0.546001	1	0.519231
…	…	…	…
2849	0.676902	1	0.726923
2850	0.546009	1	0.519231

Table 4.10. The values for comparing the similarity of pair of images for 2009 Canadian toonies.

Number of pair of images	MSER	SURF	SIFT
1	0.804638	1	0.951612
2	0.704999	1	0.782258
3	0.868281	1	0.975806
4	0.681810	1	0.850806
…	…	…	…
1080	0.72531	1	0.955645
1081	0.61209	1	0.862903

Table 4.11. The values for comparing the similarity of pair of images for 2011 Canadian toonies.

Number of pair of images	MSER	SURF	SIFT
1	0.797161	1	0.908045
2	0.745455	1	0.919541
3	0.761572	1	0.873563
4	0.832517	1	0.942528
…	…	…	…
495	0.804886	1	0.908045
496	0.901917	1	0.965517

Table 4.12. The values for comparing the similarity of pair of images for 2013 Canadian toonies.

Number of pair of images	MSER	SURF	SIFT
1	0.813835	1	0.941747
2	0.887213	1	0.954692
3	0.701563	1	0.847896
4	0.852025	1	0.915857
…	…	…	…
779	0.843856	1	0.880258
780	0.858301	1	0.954692

Table 4.13. The properties of coins and pair of images used in this research.

Dataset	Number of images		Number of pair of images	
	Genuine	**Fake**	**Genuine**	**Fake**
Danish 1990	51	25	1275	300
Danish 1991	51	14	1275	91
Danish 1996	90	10	4005	45
Danish 2008	22	114	231	6441
Canadian 1996	76		2850	
Canadian 2009	47		1081	
Canadian 2011	32		496	
Canadian 2013	40		780	

4.3 Selecting a Similarity Function

As mentioned previously, the purpose of this study is to select an algorithm among the mentioned descriptors as the best similarity function. This approach is based on our dataset and will help us in the subsequent steps. The number of comparisons depends on the number of images in our dataset. For this purpose, the result of descriptors in the histogram are compared. Since histograms are used to get the density of the underlying data distribution [13], it is leveraged to find out what values of the similarities occur most often. Figure 4.5 (a) to (c) represents histograms of the results of each descriptor in 1990s Danish coins. For instance, (a) illustrates the similarity by using the SIFT for 140 pairs of images in the 1990s Danish genuine coins, which is equal to 0.4.

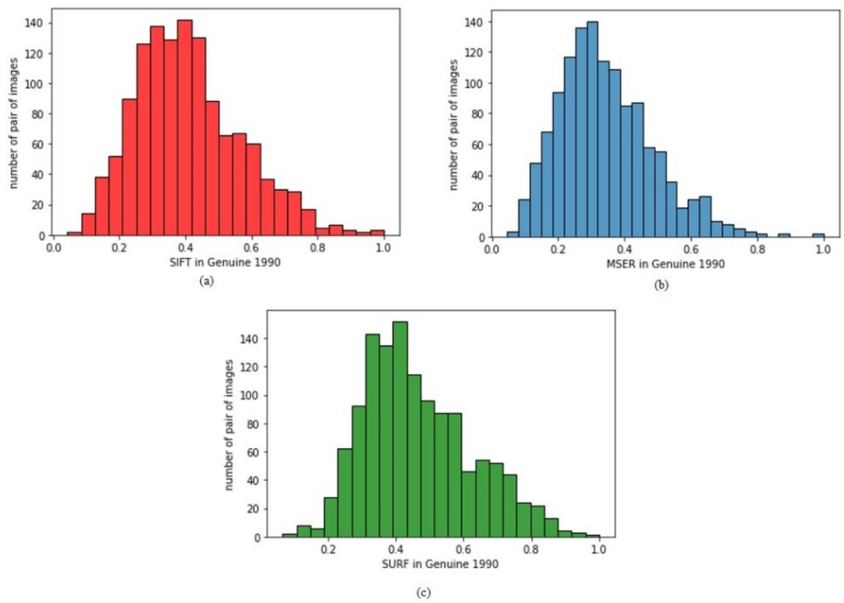


Figure 4.5. (a), (b), and (c) histograms of the result of the similarity between pair of images by using SIFT, SURF, and MSER in Danish 1990s genuine coins respectively.

In the next level, Median, Mean, Variance and Standard Deviation are used for the comparisons to find the effectiveness of values. Median is used to get a good idea of where a dataset's center is located. In addition, Mean is used to determine the center of our numerical datasets. Variance is used to measure the average degree to which point it differs from the mean. Finally, Standard Deviation is used to determine how measurements for a group are spread out from the average (mean or expected value). Figure 4.6 illustrates the bar chart of Mean, Median, Std, and Variance for SIFT, SURF, and MSER resulting from the Danish genuine coin dataset in the different years.

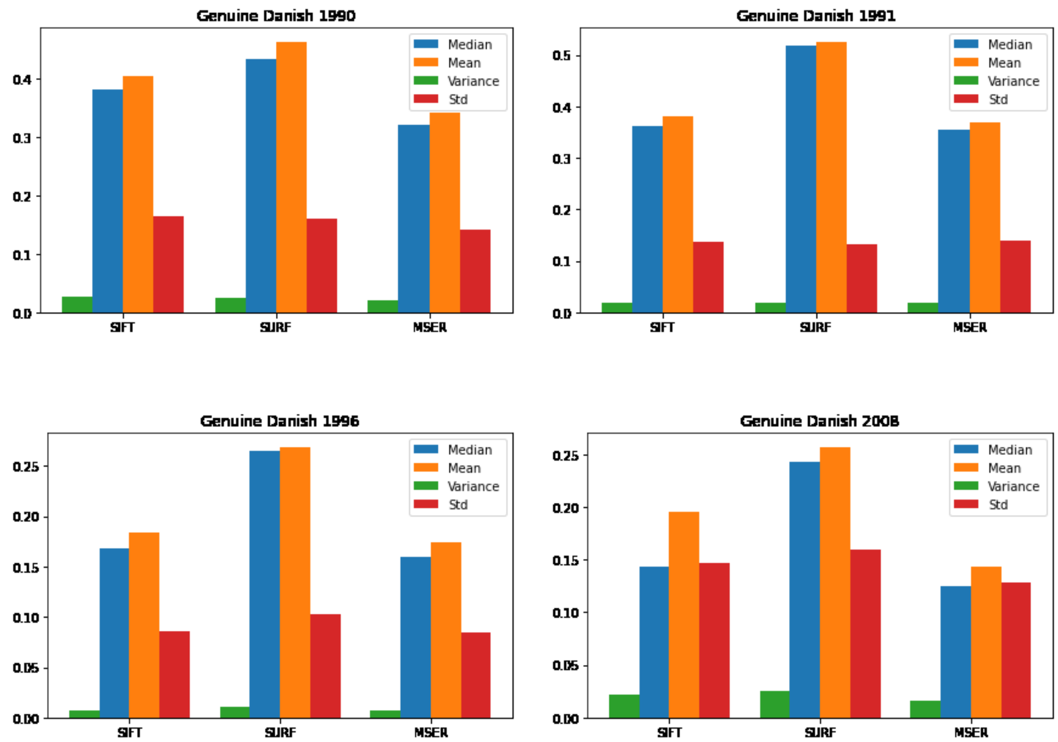


Figure 4.6. Bar chart of mean, median, std, and variance for SIFT, SURF, and MSER resulted from Danish genuine coin dataset in the different years.

After using the mentioned descriptors for image comparison, one descriptor is selected as the most reliable one for the rest of the study. Two types of the Danish coin and

Canadian toonies image datasets are compared. According to the result in Tables 4.8 to 4.12, all of the comparison rates in Canadian toonies are allocated to '1' which means all pairs of images are completely similar in this dataset. This result could not be reliable as there are some comparisons with this rate in the Danish dataset. It is worth mentioning that we apply an equal strength for finding the similarity in all descriptors. In order to compare a pair of images, all descriptors work with the same ability to match the key points. According to our investigations, unlike SURF descriptors, there were no miscalculations for SIFT and MSER in this study. According to statistical analysis, the similarity values calculated by the SIFT is more reliable than MSER in all datasets. Thus, with regard to the effectiveness of SIFT descriptors in the same dataset in the previous studies [29], and our investigation the SIFT is chosen as the similarity criterion for the next step in this research. Besides, an image that has the highest similarity with the other images is selected as a basic or reference image that is supposed to be a good representative of all images. An image in each separate dataset that has the highest rate of similarity with other images is selected using the SIFT descriptor. As a result, an image is selected as the basic coin image.

4.4 Result

In this stage of the study, we were trying to find the most reliable and effective descriptor for image comparison in our specific dataset. We proposed a procedure for selecting similarity criteria in such a way that, all the images in the dataset were compared pairwise by MSER, SURF, and SIFT descriptors. Then, we rated each comparison for a pair of images. We chose SIFT descriptor as the most reliable descriptor for the dataset through statistical analysis. We will employ this descriptor as a criterion for the next stage of the study.

Chapter 5

Autoencoding-Based Counterfeit Coin Detection

An autoencoder is a neural network that has been trained to encode (compress) and decode (reconstruct) unlabeled data so that the most valuable input data is preserved during reconstruction [30, 31]. An autoencoder is a type of unsupervised machine learning technique that converts targets to equivalent inputs using an iteration strategy. It refers to an autoencoder that has been trained to duplicate its input to its output. Figure 5.1 depicts the general structure of an autoencoder in which *h* represents the hidden layers of the autoencoder that maps *x* as the input image to *r* as the output image [29].

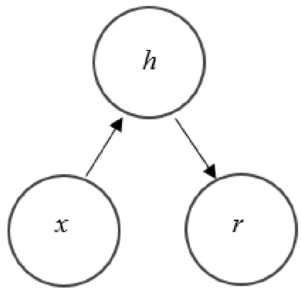


Figure 5.1. The general structure of an autoencoder.

At first glance, the procedure of copying the input to the output appears to be pointless. This assignment makes sense when we train the autoencoder to encode the most valuable aspects of input data to a latent representation. We preserve the most essential feature of the input in the *h* with a smaller dimension than before by using autoencoder [30]. An autoencoder learns how to duplicate input to itself by utilizing encoding and decoding layers as a map, according to [32].

$$\overline{X} = D(E(X))$$

That X is input data, E is an encoding map from the hidden layer to the input data, D is a decoding map from the hidden layer to the output layer, and $\overline{X}$ is reconstructed input data. A deep autoencoder is a type of autoencoder with multiple hidden layers [33], and this kind of autoencoder with convolutional layers is used in this investigation.

5.1 Autoencoder architecture

Encoding and decoding stages allow an autoencoder to reconstruct its input as output, as stated in the previous section. During the decoding phase, the input should be reconstructed with the least amount of reconstruction error based on the autoencoder's training. In other words, the disparity between input and output should be minimal. We intend to utilize autoencoder capabilities to identify counterfeit coins in the form of a model. Here, the difference is referred to as the loss function. This section constructs a model that leads to the minimum loss function after reconstruction. In fact, when the system is trained with our dataset, the loss during the compress and decompress representation operations should be minimized. In subsequent steps, the loss value is used to evaluate our suggested model. We created the autoencoder utilizing Convolutional Neural Network (CNN) layers in order to achieve the best possible outcome. We utilized a dimension reduction procedure on high-dimension data to train the autoencoder. As the encoding process progresses, the image size of the coin gradually shrinks until it is as small as latent space. When decoding, this process is reversed, going from low-dimension to actual image size in reverse order (input image size). In this study, we show that an autoencoder works as a filter and helps to emboss the dissimilarity between genuine and fake coin images. Here, we will go into greater detail about our autoencoder design.

5.2 Training the Autoencoder

Our goal is to find a way to detect counterfeit coins without the use of fake ones, as we stated previously. We use the autoencoder for this purpose. Genuine coins are used to create the initial training dataset. Our trainset contains solely genuine coins for training, and we save the fakes for testing the model. We primarily use the Danish dataset to demonstrate our suggested solution because it provides a fake sample, which we require to test the system using counterfeit coin images. In the same way, we utilize the GAN-generated image of a Canadian toonies to demonstrate the system's performance on this type of image. According to this research, the proposed method's accuracy hinges on how much data can be used to train the model. A smaller loss function (reconstruction error) is achieved by using a large number of coin pictures in the system's training. We use data augmentation to improve the dataset because there are not enough real coin images. We could lessen the decoding phase's reconstruction error by employing this method. In order to increase the amount of genuine coin images, we use the Flipping, Brightness, and Rotating method. As the research progresses, we will create the autoencoder layers. Using a four-layer convolutional neural network, we design an autoencoder in which the image size is lowered at each layer. Before feeding coin images into the autoencoder, we first scale all of the coin images to a resolution of 96*96 pixels so that they are compatible with the required specifications. It is worth mentioning that we tried the other sizes as input. We got the same result in a bigger size. Thus to avoid computational complexity and network overweight, this size is selected. The encoding phase of image compression involves dividing all of the image sizes by two at each convolutional layer. This is done in such a way that the size of the images in the fourth layer is equal to 6*6. Only then the image can be compressed into a latent space. After the image has been compressed into the latent space for the reconstruction phase, we apply this procedure in the same way but on the reverse side by transposed convolutional layer in order to obtain back the original coin images. This means that during the decoding step, we convert coin images from a low-dimensional format to a high-dimensional format (actual size 96 by 96). The size of the image increases proportionally in each successive layer in such a way that the dimension is increased by a factor of two in each successive layer. In every layer, we utilize the linear activation method, and we set the padding to "*same*." In addition, we maintain the stride at a constant value of 2 across all layers. It is important to point out that the design of our CNN layers in this investigation is based on a process of trial and error so that we can get the best possible results from the model. Figure 5.2 illustrates the design of the autoencoder for our proposed method.

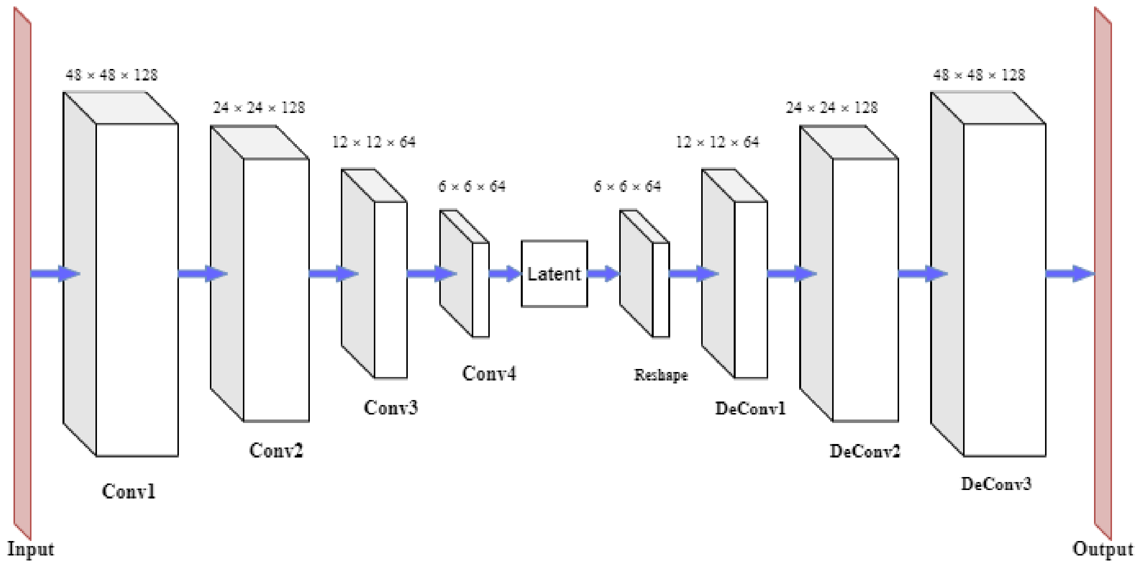
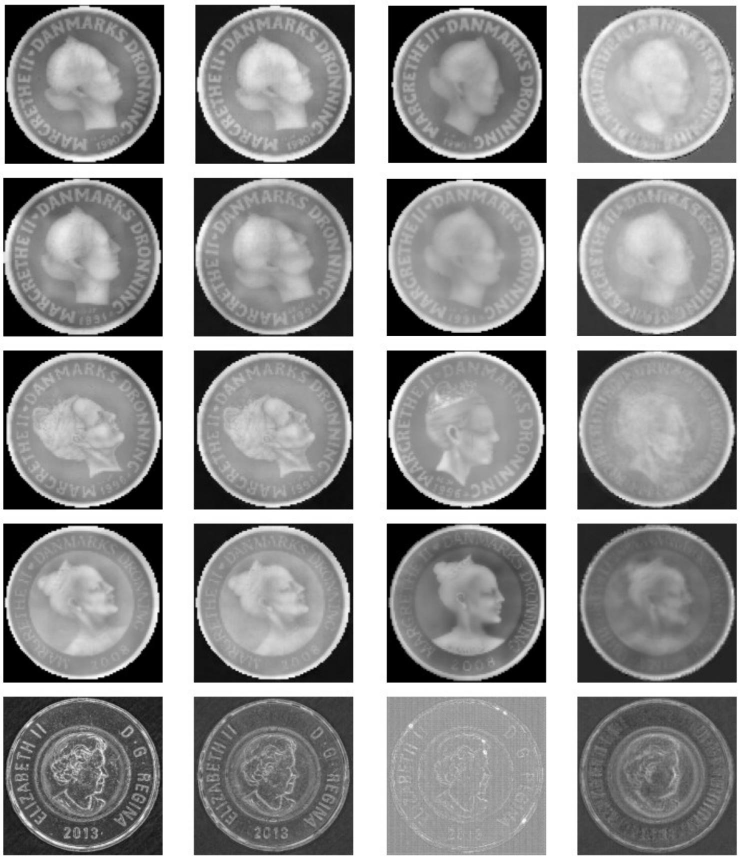


Figure 5.2. Structure of the autoencoder in the proposed method.

5.3 Custom Classifier

This study trains a specific dataset with an autoencoder to have the minimum loss function during the reconstruction. Thus, the measure of loss function is placed in a specific range with a particular threshold. The intention is to create an autoencoder that learns to reconstruct an image as similar as possible to the original input image Figure 5.3 (a) in this case. The autoencoder is fed by genuine coin images in the first step. After autoencoder reconstruction, a new image is generated that is expected to be very similar to the original one (input) Figure 5.3 (b). The similarity of the selected coin image and the image generated by the autoencoder is measured in this step. As previously mentioned, an autoencoder creates a new coin image after being trained on the genuine coin images. SIFT descriptor is used to calculate the similarity of these two images. According to our prior findings in this study, the SIFT descriptor produces the best result between the images in our dataset. A result with a specified threshold is obtained after measuring these two images (selected basic coin image and autoencoder output coin image). To show how the proposed model works, the previously trained autoencoder is tested with a counterfeit coin image Figure 5.3 (c). For example, when the model is trained with the genuine Danish coins using the counterfeit Danish coins in the same year, the model reconstructs another output Figure 5.3 (d). The similarity of this output is compared with the selected basic image same as the first step. This similarity is lower than the previous step. Therefore, a defective result is achieved when testing the system with a counterfeit coin. It means that the reconstruction loss is pretty high when testing the system with a counterfeit image coin. The loss function for genuine coins should be in a specific threshold. Thus, when finding any other loss function outside the specified threshold between the output and the selected coin image, the outcome might be known as anomalous or counterfeit. Figure 5.3 illustrates the input and output coin images used in the proposed method.



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

Figure 5.3. Examples of input and output of autoencoder for genuine and fake coins.

Images in the (a) column represent a sample of input genuine coin images (trained data) to the system for each separate year. (b) Illustrates the output (reconstructed) coin images by the system when using genuine coins as input. (c) Demonstrates the counterfeit coin images as input for testing the system. (d) Shows the output (reconstructed) coin images by the system when using counterfeit coins as input. Table 5.1 indicates four types of Danish coins and one type of Canadian toonies that were used in the study for model training and evaluation.

Table 5.1. The properties of coins and augmented images used in this research.

Dataset	Number of images		Number of augmented images for training	Test set	
	Genuine	Fake	Genuine	Genuine	Fake
Danish 1990	51	25	204	40	10
Danish 1991	51	14	204	41	12
Danish 1996	90	10	360	72	9
Danish 2008	22	114	88	20	30
Canadian 2013	40	15	160	32	13
All coins	254	178	1016	205	74

5.4 Experimental Results

In this section, we conducted experiments to evaluate the performance of the proposed method. The hardware that has been used during the test was an iIntel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90 GHz, DDR4 16GB RAM, NVIDIA GeForce 940MX with 4 GB Dedicated VRAM. The operating system used was Windows 10-64 bit, and the programming environment was MATLAB 2014 and Google Colaboratory.

5.4.1 Results

To evaluate the system, the system receives the images as input from the test set that contains previously unseen images of the fake and genuine coins. Then, the SIFT descriptor is used to measure similarity between the input and selected basic image. In order to implement the system, the system is first fed with 2008 year genuine coin images for reconstruction by the autoencoder. Then, the similarity between the input and a selected basic coin image is measured by SIFT. As mentioned before, we selected a genuine high-quality coin image as a selected basic image. According to our investigation, the average similarity between the selected basic coin image and the generated ones calculated by SIFT is 0.93. It is worth mentioning that, all coin images stated in this section are meant to be 2008 year Danish coin images. A counterfeit coin is used for the same year as input to test the system, and then the degree of resemblance is measured. The SIFT algorithm found an average of 0.73 similarity between generated Danish counterfeit coin image and the selected basic coin image. The significant difference between fake and genuine coins in terms of the similarity criteria in different inputs indicates that the system was successful in revealing the dissimilarity between counterfeit and genuine coins. While the amount of dissimilarity was low before using the autoencoder. Table 5.2 indicates the average similarity between the basic image and input images for both genuine and fake coin images. Here, we need a threshold to determine when a coin is abnormal and classify it as a fake coin. If the similarity between the test input and our basic coin is less than the threshold, the input is classified as a counterfeit coin. This threshold has been selected manually 0.81, 0.82, 0.76, 0.81 for Danish 1990, Danish 1991, Danish 1996, and Danish 2008 respectively. As a result, with the performed model, the genuine coin images could be detected from the counterfeit coin images. Table 5.3 demonstrates the average similarity between the image and output image when the system is fed with genuine coin images for the first time. Table 5.4 illustrates the average similarity between the basic image and output image when the system is tested with counterfeit coin images for the first time.

Table 5.2. Average similarity between the basic image and input images for both genuine and fake coin images.

Dataset	Genuine with basic	Fake with basic
Danish 1990	0.98	0.86
Danish 1991	0.97	0.84
Danish 1996	0.98	0.71
Danish 2008	0.96	0.84

Table 5.3. Average of comparison of the generated coin image by autoencoder with the selected basic coin image when we use genuine coin images as input.

Dataset	Similarity by SIFT
Danish 1990	0.91
Danish 1991	0.90
Danish 1996	0.91
Danish 2008	0.93
Canadian 2013	0.94

Table 5.4. Average of comparison of the generated coin image by autoencoder with the selected basic coin image when we use counterfeit coin image as input.

Dataset	Similarity by SIFT
Danish 1990	0.70
Danish 1991	0.77
Danish 1996	0.55
Danish 2008	0.73
Canadian 2013	0.12

The proposed method is compared with four recent studies published in the field of counterfeit coin detection. It should be noted that the data for this comparison was exactly the same as the data used for training and evaluating our proposed method. To this end, the accuracy of methods trained by the Danish coins dataset is computed and compared. According to Table 5.5, the proposed method using autoencoder outperformed other methods in some cases. Although the accuracy of the proposed model is a bit lower than [11], our proposed model is faster than others in detecting counterfeit coins. It is worth mentioning that our proposed model is trained without the necessity of counterfeit coins.

Table 5.5. Comparing the proposed method with several different previous methods in terms of accuracy

Dataset	[9]	[34]	[1]	[11]	Proposed
Danish 1990	NA	87.2%	93.2%	98.6%	98.4%
Danish 1991	92.1%	86.4%	96.6%	98.0%	97.8%
Danish 1996	96.8%	98.0%	100%	99.8%	99.8%
Danish 2008	95.5%	92.%	99.6%	99.9%	99.6%

Chapter 6

Fake Canadian Mint Mark Detection

6.1 Selecting the Mint Mark

Fake Canadian toonies are circulating in the market, as mentioned earlier. As a result, we decided to propose a method for detecting counterfeit Canadian toonies using our previously designed system. Mint Marks, plays an important role in coin detection. The dies used to strike the Mint Marks on coin surfaces are elegantly designed. As a result, forging a coin with special Mint Marks for forgers is difficult, and this part of the fake coin's surface is always the most vulnerable. Figure 6.1 depicts a sample of genuine and counterfeit maple leaf Mint Marks on Canadian toonies.

(a) (b) (c)

Figure 6.1. (a) A genuine Maple leaf Mint Mark in 1996 Canadian toonies in, (b) A genuine Crown Mint Mark in 2011 Canadian toonies, and (c) fake maple leaf

Furthermore, training the system on a specific feature of coin images rather than the entire coin image surface could speed up the detection process. As a result, we are focusing on this specific symbol of Canadian toonies in this section. First, we must remove the selected Mint Mark from the coin's surface. We trained the system with coin images and used the selected Mint Marks images as objects for labeling in this work. The YOLO algorithm [35] is then used in this study to detect and recognize the Mint Marks.

6.2 Edge Detection

We will use the detected Mint Marks from Canadian toonies in this section. To better access the border of these symbols, we use the edge detection technique. Edge detection can be used in image processing to extract features from important parts of objects. By detecting edges, a set of curves indicating object contours, surface component borders, and changes in surface direction is generated. Furthermore, edge detection techniques reduce data volume by retaining important image features while removing unnecessary information. In this experiment, we use the Canny edge detector to gain easier access to the Mint Marks borders on Canadian toonies. The majority of the coins in the dataset were degraded with numerous scratches on the coin surface. Considering we determined the values of *minVal* and *maxVal* through trial and error in order to have the best access to the borders of coin images, we set the *L2gradient* equal to *True* to calculate the image gradient magnitude more accurately. The results of the Canny edge detection technique at various values of *minVal* and *maxVal* are shown in Figure 6.2.

(a) (b) (c)

Figure 6.2. Applying the edge detection technique for Canadian toonies Mint Marks.
(a) Selected Mint Marks of Canadian toonies before preprocessing. (b) Selected Mint Marks after Canny edge detection *minVal=100* and *maxVal=200*. (c) Selected Mint Marks after Canny edge detection without noise *minVal=200* and *maxVal=500.*

6.3 Generating Fake Mint Marks

Deep learning algorithms are used in this section of the study to design our proposed method. To generate fake coin images, we use the same Generative Adversarial Network (GAN) that was used previously. During this research, we aimed to propose a method that does not require the use of fake coin images during the training process. In order to test the model, counterfeit coin images are required. Despite the fact that counterfeit Canadian toonies are widely available, we do not have access to the counterfeit coin images for our research. As a result, we use GAN to compensate for the lack of fake coin images. We generate fake Mint Marks images from the symbols we chose in the previous step to use as a test set for our model. Figure 6.3 depicts the GAN-generated fake Mint Marks images.

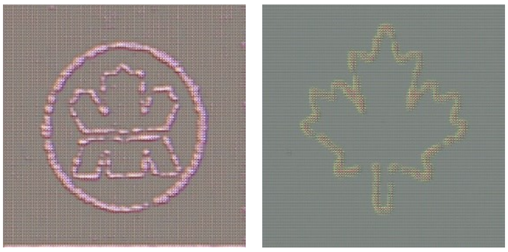


Figure 6.3. Generated fake Mint Mark images by GAN.

6.4 Autoencoder for Mint Marks

In this section, we design the autoencoder in the same manner as in the previous chapter. We train the autoencoder with four convolutional layers. This time we use the preprocessed Mint Marks images as input to the system. The system compresses and decompresses the images. To test the system we use GAN-generated images as input to the system to receive another output. After reconstruction, we use the output images to evaluate the study in the next section. It is worth mentioning that we train the system with two kinds of Mint Marks in two different steps. The system output images are demonstrated in Figure 6.3.

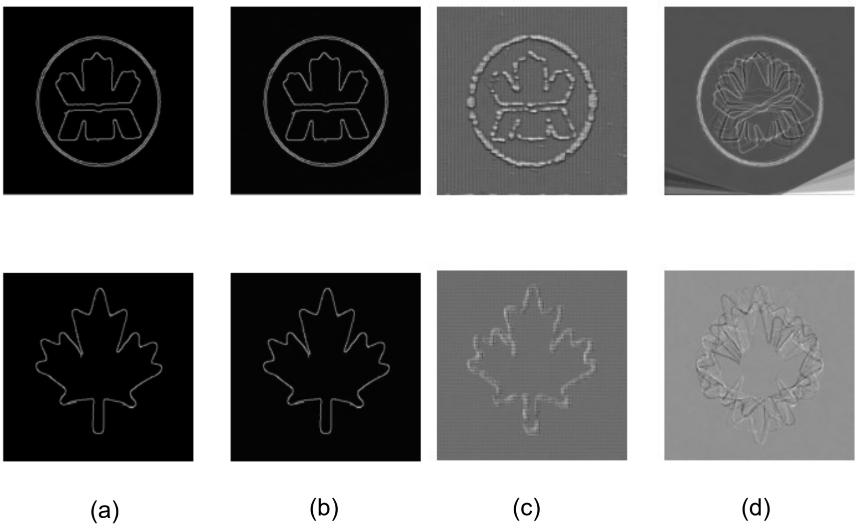


Figure 6.4. Examples of input and output of autoencoder for genuine and fake Mint Marks.

Images in the Figure 6.4 (a) column illustrates a sample of input genuine Mint Mark images (trained data) to the system for each separate year, (b) represents the output (reconstructed) Mint Mark images by the system when using genuine coins as input, (c) demonstrates the counterfeit Min Mark (GAN-generated) images as input for testing the system and (d) shows the output (reconstructed) Mint Mark images by the system when using counterfeit coins as input.

6.5 Result

In this section, we measure the dissimilarity between output images and their inputs in each feeding step to achieve the amount of difference between them. This amount leads us to evaluate the reconstruction error (loss function) in each step. As we mentioned in chapter 5, our goal is to train this system so that the system will make errors in some situations. To compare the output images in each separate feeding, we employed two metrics in this step. First, we use Mean Squared Error (MSE). This metric indicates the average difference between pixel intensities of images across the entire image. It is important to note that a higher value of MSE conveys a greater discrepancy between the input image and the reconstructed image. Second, the Peak Signal-to-Noise Ratio (PSNR) is used to compare the quality of reconstruction images. Table 6.1 and Table 6.2 illustrate the comparison between the Mint Mark image and output images in both feeding steps (training and testing) by MSE and PSNR respectively. For example, Table 6.1 indicates the average dissimilarity measured by MIC metric between Crown Mint Mark images and their output images (the outputs when training the system with the genuine sample) is equal to 0.521739, and the dissimilarity between the counterfeit Crown Mint Mark images and their output images (the output when testing the system with the GAN-generated sample) is equal 17652.32. This huge amount of difference demonstrates the reconstruction error is relatively high when we use counterfeit Mint Marks as input to the system. Considering the system is unable to reconstruct the counterfeit Mint Mark properly. This input can be detected as a counterfeit coin. Table 6.2 represents the difference in the quality of reconstruction. As seen, the quality of reconstruction for genuine Mint Marks images as input is significantly bigger than the counterfeit input ones. Thus with this amount of difference, the system can conclude the input sample is a counterfeit coin.

Table 6.1. Average of comparison of the generated Mint Marks image by autoencoder with both genuine and counterfeit inputs using MSE metric.

Input	Output
Crown Mint Mark	0.521739
Counterfeit Crown Mint Mark	17652.32
Maple Mint Mark	0.625637
Counterfeit Maple Mint Mark	17406.37

Table 6.2. Average of comparison of the generated Mint Marks image by autoencoder with both genuine and counterfeit inputs using PSNR metric.

Input	Output
Crown Mint Mark	50.95
Counterfeit Crown Mint Mark	5.61
Maple Mint Mark	49.91
Counterfeit Maple Mint Mark	4.2

Chapter 7

Conclusion and Future Work

7.1 Conclusion

In fake coin detection studies, accessing fake coins for training is difficult. So, we proposed a counterfeit coin detection method that does not require the use of fake coins to train the models. The results of MSER, SIFT, and SURF in Danish coin images were examined in the first stage of the study. The images from each year were then compared to the descriptors listed above. Statistical analysis was used to determine the best descriptor for generating more reliable similarity between two coin images in our dataset. As a result, SIFT descriptors were chosen for our datasets. In the following stage of research, we proposed our model by creating an autoencoder with layers of Convolutional Neural Networks, encoded genuine coin images, and then reconstructed them from latent images. We trained the model to reconstruct the input genuine coin images with the highest accuracy and lowest possible loss-function. We calculated the similarity degree between the model's output and the input images of genuine coins. As a criterion for the final stage of the research, we used a specific range. Finally, we fed the previously trained model counterfeit coin images. As a result, we determined that the range of difference exceeds the specified criteria.

We also used edge detection techniques to locate the border of Mint Marks on Canadian toonies. We proposed the use of an autoencoder and a Generative Adversarial Network (GAN) to detect fake Mint Marks on Canadian coins. Our method is capable of detecting counterfeit Danish coins with 98.4%, 97.8%, 99.8%, and 99.6% for Danish 1990, Danish 1991, Danish 1996, and Danish 2008 respectively. In addition, this research allows us to train a system to detect counterfeit coins without the use of fake coins. Since there are various forgery sources in the production of imitation coins, this study helps reduce the need to access all types of fake coins for training in the same year. However, the method misclassifies poor quality coin images in some cases and detects them as fake samples.

7.2 Future Work

The accuracy of fake coin generation is increasing as technology advances. Since future counterfeit coins may appear to look like genuine coins from a distance the proposed method may be vulnerable. To address these potential issues, the proposed model should be trained in a variety of areas. In this manner, we should pay more attention to all individual features of coin images separately in order to create a robust system against precise fake coins. This accomplishment could be attained through different methods. For example:

1. We can access more accurate and perfect fake samples by using the new generation of Generative Adversarial Networks, such as E-GAN. Despite the fact that in this thesis we proposed a model that no longer requires fake samples to train the system. It is unavoidable that fake samples will be used to test the system. With the immaculate fake generated coins by the last generation of the GAN, we could test the system with high accuracy and reduce the Error coefficient.

2. We could access each part of a coin's surface separately by using high-resolution coin images. By training the system with each individual element of coin images and combining them, we could have a well-trained and more robust system against proper fake coins.

References

[1] Liu, L., Yue, L., Suen, C. Y.: An Image-Based Approach to Detection of Fake Coins. In: Transactions on Information Forensics and Security, Vol. 12, pp. 1227–1239. IEEE, (2017).

[2] The protection of Euro coins https://ec.europa.eu/info/index_en.

[3] Fake toonies discovered in Hawkesbury, Ont. https://www.cbc.ca/news/canada/ottawa/fake-toonies-hawkesbury-1.6313064.

[4] Khazaee, Saeed, Maryam Sharifi Rad, and Ching Y. Suen. "Detection of counterfeit coins based on modeling and restoration of 3D images." In International Symposium Computational Modeling of Objects Represented in Images, pp. 178-193. Springer, Cham, 2016.

[5] 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, p. 383-387, 2018.

[6] R´acz, A., H´eberger, K., Rajk´o, R., Elek, J.: Classification of counterfeit coins using multivariate analysis with X-ray diffraction and X-ray fluorescence methods. Forensic Sci. Int., vol. 115, pp. 129–134, https://doi.org/10.1016/S0379-0738 (00)00309-1, 2001.

[7] Tresanchez, M., Pallej`a, T., Teixid´o, M., Palac´ın, J.: Using the Optical Mouse Sensor as a Two-Euro Counterfeit Coin Detector. Sensors (Basel, Switzerland). 9(9):7083-7096, 2009.

[8] Jain, N., Jain, Neha. Coin Recognition Using Circular Hough Transform. International Journal of Electronics Communication and Computer Technology (IJECCT) 2(3), 2249-7838 (2012).

[9] Sun, K., Feng, BY. Atighechian, P., Levesque, S., Sinnott, B., Suen, C. Y.: 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 165-170, (2015).

[10] Huber, R., Ramoser, H., Mayer, K., Penz, H., Rubik, M.: Classification of coins using an Eigen space approach. Pattern Recognit. Lett. 26 (1) 61–75 (2005).

[11] Khazaee, Saeed, Maryam Sharifi Rad, and Ching Y. Suen. "Detection of counterfeit coins based on 3D height-map image analysis." Expert Systems with Applications 174: 114801 (2021).

[12] Chakraborty, Arpan, Joydeb Ghosh, Piyali Datta, Ankita Nandy, and Rajat Kumar Pal. "Anomaly Detection and Three Anomalous Coins Problem." In Advanced Computing and Systems for Security, pp. 303-320. Springer, New Delhi, 2016.

[13] K. Okada, "Method and apparatus for discriminating coins or bank notes," U.S. Patent 4556140, 1985.

[14] Y. Furuya, T. Ishida, I. Fukuda, and G. Yoshizawa, "Method and apparatus for sorting coins utilizing coin-derived signals containing different harmonic components," U.S. Patent 4971187, 1990.

[15] Rad, M. S., Khazaee, S., Liu, L., & Suen, C. Y. A Blob Detector Images-Based Method for Counterfeit Coin Detection by Fuzzy Association Rules Mining. In International Conference on Pattern Recognition and Artificial Intelligence (pp. 669-684). Springer, Cham, 2020.

[16] Khazaee, S., Rad, M.S. and Suen, C.Y., Decomposing Relief Maps to Detect Counterfeit Coins Using a Hybrid Deep Learning Method. Available at SSRN 3990636, 2020.

[17] Reisert, M., Ronneberger, O., Burkhardt, H. A Fast and Reliable Coin Recognition System. Proceedings of 29th DAGM Symposium, vol. 4731, pp. 415-424, Heidelberg, Germany, Sep.12-14, 2007.

[18] Shen, L., Jia, S., Ji, Z., Chen, W.S. Extracting Local Texture Features for Image-based Coin Recognition. IET Image Processing, vol. 5, pp. 394-401, Aug. 2011.

[19] Zaharieva, Maia, Martin Kampel, and Sebastian Zambanini. "Image based recognition of ancient coins." In International Conference on Computer Analysis of Images and Patterns, pp. 547-554. Springer, Berlin, Heidelberg, 2007.

[20] Wei, K.P., He, B., Wang, F., Zhang, T., & Ding Q.J. A Novel Method for Classification of Ancient Coins Based on Image Textures. Proceedings of the Second Workshop on Digital Media and Its Application in Museum & Heritage, pp. 63-66, 2007.

[21] H. Bay, T. Tuytelaars and L. Van Gool, "SURF: Speeded up robust features", Proc. European Conference Computer Vision, pp. 404-417, May 2006.

[22] Munipalle, Vamshi Krishna, and Usha Rani Nelakuditi. "Performance Analysis of Coin Matching Using Local Texture Features." In 2019 International Conference on Cutting-edge Technologies in Engineering (ICon-CuTE), pp. 69-74. IEEE, 2019.

[23] Matas, J., Chum, O., Urban, M., Pajdla, T. Robust Wide Baseline Stereo from Maximally Stable Extremal Regions. Image and Vision Computing, vol. 22, pp. 761-767, Sep. 2004.

[24] David G. Lowe, "Distinctive image features from scale-invariant keypoints," International Journal of Computer Vision, vol. 60, no. 2, pp. 91-110, 2004.

[25] M. Kampel, R. Huber-Mork, and M. Zaharieva, "Image-based retrieval and identification of ancient coins," IEEE Intelligent Systems, vol. 24, no. 2, pp. 26-34, 2009.

[26] 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., pp. 2672–2680, 2014.

[27] Extremal, Maximally Stable, J. Matas, O. Chum, M. Urban, and T. Pajdla. "Robust wide baseline stereo from." In In British Machine Vision Conference. 2002.

[28] Pearson, K.: Contributions to the Mathematical Theory of Evolution. II. Skew Variation in Homogeneous Material. Philosophical Transactions of the Royal Society of London. A, Vol. 186 pp. 343-414, (1895).

[29] Gakhar, Sofia. "Local Image Patterns for Counterfeit Coin Detection and Automatic Coin Grading." PhD diss., Concordia University, 2020.

[30] Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. Deep learning. MIT press, 2016.

[31] Kramer, Mark A. "Nonlinear principal component analysis using autoassociative neural networks." AIChE journal 37, no. 2: 233-243 (1991).

[32] Zhou, Chong, and Randy C. Paffenroth. "Anomaly detection with robust deep autoencoders." In Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining, pp. 665-674. 2017.

[33] LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. "Deep learning." nature 521, no. 7553: 436-444 (2015).

[34] Semin, K., Lee, S. H., Ro, Y. M.: Image-based coin recognition using rotationinvariant region binary patterns based on gradient magnitudes. J. Vis. Commun. Image Represent., vol. 32, pp. 217–223, 2015.

[35] Redmon, Joseph, Santosh Divvala, Ross Girshick, and Ali Farhadi. "You only look once: Unified, real-time object detection." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 779-788. 2016.