Coin Detection and Classification using a Few-Shot Learning method based on Siamese Network

Mahsa Vahed

A Thesis

In the Department of

Electrical & Computer Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Master of Applied Science at

Concordia University

Montreal, Quebec, Canada

August 2024

CONCORDIA UNIVERSITY
School of Graduate Studies

This is to certify that the thesis prepared

By: Mahsa Vahed

Entitled: Coin Detection and Classification using a Few-Shot Learning method based on Siamese Network

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

Master of Applied 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. Chunyan Wang

______________________________ Examiner
Dr. Chunyan Wang

______________________________ Examiner
Dr. Yong Zeng

______________________________ Thesis Supervisor(s)
Dr. Wei-Ping Zhu

______________________________ Thesis Supervisor(s)
Dr. Ching Yee Suen

Approved by ______________________________
Audrey Veileux, Graduate Program Director Chair of Department or Graduate Program Director

____
Dr. Mourad Debbabi Dean of Faculty

August 29, 2024

Abstract

Coin Detection and Classification using a Few-Shot Learning method based on Siamese Network

Mahsa Vahed

Coins are used in our daily lives for a long time with less depreciation than paper currency. Detecting counterfeit coins visually is a challenging way with lots of errors. This thesis investigates advanced machine-learning techniques to differentiate between counterfeit and genuine coins with a small dataset. It focuses on the implementation of few-shot learning. This study is applied to two different types of datasets. The first dataset contains the images converted to grayscale, and the second dataset contains the four slopes images. As the detection of counterfeit coins is challenging due to their high similarity with genuine coins, more features are required before pre-training the neural network.

For this study, 2,474 labeled images from the CENPARMI dataset belonging to 22 different classes were used. To enable experimentation, the dataset was split into two parts: a Main Dataset (D_m) and a Target Dataset (D_t). We used a pre-trained model, which learns from the D_m and adapted it to D_t. The Inception V3 network was fine-tuned in the main dataset to learn general coin characteristics. This knowledge was transferred to the target dataset to learn new coin types from a few images. FSL using Siamese networks and contrastive loss was used. The algorithm performance was evaluated using the total accuracy with different epochs and different batch sizes to earn the optimum of them, and also the precision and recall and F.score per class.

It is shown that the accuracy of our method in epoch 20 is optimal. At this point, the model achieves a high level of accuracy (92.13% for grayscale images and 94.73% for SMMIG images). the model trained with a batch size of 32 achieves the highest accuracy of 92.13% for the grayscale dataset and 94.73% for the SMMIG dataset, indicating that moderate batch sizes contribute to optimal performance.

Acknowledgment

I would like to express my sincere gratitude to the individuals who have played crucial roles in the completion of my thesis.

I would like to express my sincere gratitude to my supervisors, Prof. Ching Yee Suen and Prof. Wei-Ping Zhu, for their continuous support, invaluable guidance, and scholarly mentorship during my academic journey. Their expertise, encouragement, and commitment have played a vital role in shaping the direction of my research. Moreover, their encouragement, constructive feedback, and dedication to my intellectual growth have been truly inspiring and have helped me to become a better researcher. Their caring and supportive demeanor extended beyond academic matters and they have always been there for me whenever I needed their guidance and advice.

I am extremely thankful to all the staff members at CENPARMI, Concordia University for their help and support during my studies. I want to thank Nicola Nobile, the research manager at CENPARMI, for his exceptional technical assistance, cooperative spirit, and unwavering support.

I would like to express my sincere gratitude to the members of my examination committee for dedicating their time, sharing their expertise, and providing me with *invaluable feedback on my thesis. Their constructive comments and scholarly contributions have significantly enhanced the quality of my work.*

I am extremely thankful to my parents and my brother. I owe them an immense debt of gratitude for their unconditional love, unwavering encouragement, and unwavering belief in my abilities. Their sacrifices, guidance, and unwavering support have been the cornerstone of my academic achievements, and I am deeply grateful for their enduring presence in my life.

I would like to express my deepest appreciation to my husband, Saeed, for his unwavering love, encouragement, and support throughout my journey. His patience, understanding, and belief in my dreams have been a constant source of strength and inspiration. I am truly blessed to have him by my side, and I am grateful for his unwavering support and companionship.

Table of contents

List of Figures

List of Tables

List of Abbreviations

BBSC	Binned Borders in Spherical Coordinates
CNN	Convolutional Neural Networks
FN	False Negatives
FP	False Positives
FSL	Few-Shot Learning
MSER	Maximally Stable Extremal Region
PBDA	Precipice Boundary Detection Algorithm
Pi	Precision
RFR	Regional Binary Patterns
Ri	Recall
SIFT	Scale Invariant Feature Transform
SMMIG	Steep, Moderate, MidModerate, and Gentle slope
SVM	Support Vector Machines
TN	True Negatives
TP	True Positives

Chapter 1

Introduction

1.1 Motivation

Coins are generally used in our life such as retail kiosks, supermarket self-checkout machines, arcade gaming machines, payphones, launderette washing machines, car parking meters, automatic fare collection machines, public transport ticket machines, and vending machines for soft drinks, cigarettes, candies, etc. [1]. Coins can be used for a long time with less depreciation than paper currency and can be used for extended periods of time. People enjoy collecting coins not only because they usually have artistic value but also valuable antique coins. Based on the Royal Mint website[2], any amount of counterfeiting is concerning because it has the potential to erode public trust in a nation's money. Furthermore, it's against the law to use fake coins [2]. Based on a paper [3] published by Royal Mint in 2017, there are around 1.6 billion round pound coins in circulation in the UK, and the Royal Mint estimates that 2.55 percent of them about 40 million pounds are fake.

Detecting and eliminating counterfeit coins has several benefits, such as promoting economic stability, building trust in currency, protecting collectors, supporting law enforcement, and maintaining market integrity [4]. The primary idea behind coin detection is to compare the physical attributes of a coin to the accepted standards for legitimate coins. After assessing the coin's weight, diameter, thickness, metal composition, and magnetism, the coin acceptor sends an appropriate electrical signal through its output connection for coin detection [5]. Several researchers have invested significant effort into coin identification and counterfeit coin detection [1, 6-11]. This research focuses on two issues: coin classification and counterfeit coin detection. The detection of counterfeit coins is a current area of study, with various approaches developed by different academics to address this issue.

The primary motivation for switching from physical features to machine learning and computer vision approaches to detect counterfeits is due to the same physical attributes and metal type in coins from different countries. A metal item that precisely matches weight, size, and type of metal can confuse the system, which is the primary disadvantage of the two systems based on physical characteristics. Therefore, by concentrating on design elements, errors can be removed, and a stronger categorization system can be produced. Computer-designed automated solutions are less expensive, more methodical, and able to be implemented remotely without the need for human interaction [12].

Coins can be photographed under various conditions, with different contrasts, lighting, and backgrounds. These factors can significantly interfere with coin recognition. Deep learning models, which require a fixed, relatively small size of the input image, may lose essential coin features due to naive resizing of the input image. Therefore, before feeding the image to the coin recognition model, it's crucial to locate the coin and remove its background [13]. Figure 1.1 (a) shows the samples of Canadian Coins year 1996, (b) shows the samples of Danish Coins year 1990. (c) shows the samples of Chinese Coins – Memento year 1927 and shows (d) the samples of Chinese Coins – China-Year3 1911-G-One-Dollar-Color. All images are captured with Keyence scanner at the CENPARMI lab.

(a)

(b)

(c)

(d)

Figure 1.1: (a) Samples of Canadian coins year 1996, (b) Samples of Danish coins year 1990. (c) Samples of Chinese coins – Memento year 1927, and (d) Samples of Chinese coins – china-year3 1911-G-one-dollar- color. All images were captured by Keyence scanner in the CENPARMI lab.

1.2 Challenges

Counterfeit coin detection, a key component of digital imagery and numismatics, is essential to many areas of economic and historical study.

Various automatic fake machine detectors are used in coin detection as an initial device for assessing a distinct component of the coin's features [14]. However, these technologies cannot distinguish the difference between counterfeit and genuine coins when their physical features are identical. The scientific field has been overflowing with studies on image-based techniques for detecting counterfeit coins in recent years [7, 15, 16].

The widespread use of object identification technology has been driven by the quick advancements in digital image processing, pattern recognition, machine vision, and machine learning. Even with these developments, it is surprisingly difficult to create a system that can accurately identify a specific coin because most coins have a similar appearance. Neural networks are mostly used in coin identification techniques to extract different picture characteristics and categorize the feature vectors using classifiers [17].

1.2.1 Size of the coins

The challenges in distinguishing between genuine and counterfeit coins are worsened by their tiny size, as most coins are similar in size. This intrinsic property makes it difficult for people to detect microscopic differences in minute characteristics that serve as differentiating qualities between genuine and counterfeit coins. In this situation, experts play a critical role, depending on their tactile senses to distinguish minute distinctions in surface, edges, letters, size, and texture—a skill set beyond the common person's grasp. Furthermore, the lack of specialist instruments to help in recognizing these discrepancies adds to the difficulty of differentiating real from counterfeit coins. As a result, expertise in coin authenticity validation remains crucial, especially in the absence of accurate sizing technologies.

1.2.2 High quality of counterfeit coins

When counterfeit coins are meticulously constructed to closely resemble the qualities of genuine coinage, traditional techniques of identification encounter major challenges. Well-made counterfeits frequently mimic the weight, look, and even tactile features of real coins, making it difficult for people and even automated systems to distinguish the tiny variations. Counterfeiters' sophisticated skills contribute to the seamless absorption of these reproductions into circulation, aggravating the difficulties of correct detection. Given these issues, there is a growing need for stronger security measures, innovative technology, and professional expertise to properly identify and struggle with the rise of well-produced counterfeit coins in circulation.

1.2.3 Coin images

Establishing a dataset of counterfeit and genuine coin images posed challenges, primarily centered around the search for a proper scanner. Because of the reflecting nature of the metallic, bright coin surfaces, as well as their small size, necessitated the use of a specific instrument for image scanning. This task proved to be inherently challenging due to the characteristics of coins. To address this, careful consideration of the scanning environment became imperative to minimize light reflections and ensure optimal results. Creating an environment to reduce glare emerged as a critical aspect in overcoming the challenges with scanning coins, ultimately contributing to the successful compilation of a comprehensive dataset for further analysis and research.

1.2.4 Lack of fake coin images

In the world of coin collecting, distinguishing genuine from counterfeit coins can be challenging. Some counterfeit coins are easily identified, others are expertly crafted to mimic the real thing. The possession of counterfeit coins is strictly prohibited by the government, and legal consequences await those found in possession of them. The government does not release images of fake coins due to security concerns, making it challenging for researchers to have meaningful investigations. This lack of data in existing literature presents a significant obstacle, particularly in the realm of counterfeit coins.

In recent coin detection investigations, the Concordia University (CENPARMI) lab in Montreal, Quebec, has developed numerous unique ways for identifying counterfeit coins. They progressed from two-dimensional to three-dimensional image processing for coin recognition [8, 18], capturing height and depth rather than color levels. The authors of [19] focused their research on coin weights and employed an autoencoder to identify coins. It has been studied how to identify fake anomalous coins using an autoencoder.

Large datasets are necessary for most machine learning algorithms for effective training and performance. However, there is a noticeable lack of images, especially for ancient coins, which makes developing robust models for coin recognition difficult [20]. As a result, coming up with a technique to reliably detect counterfeit coins in a small dataset is not only technologically necessary but also extremely useful. In the case of rare and antique coins, when picture resources are few and counterfeiting is a serious concern, this method would be extremely helpful for identification and categorization, and this remains an open area of research.

1.3 Objective

This research uses machine-learning techniques to differentiate between counterfeit and genuine coins. Its uniqueness lies in its focus on achieving this objective with a smaller dataset. With the increasing sophistication of counterfeit coins, a detection system that can adapt and be precise is crucial. The intentional decision to work with a smaller dataset acknowledges the challenges of not having a vast array of diverse counterfeit coin images.

1.4 Contribution of the thesis

Deep learning approaches need the collection and annotation of enormous image datasets, which is either technically or economically impractical. This research proposes a new ground of counterfeit coin recognition by introducing and implementing few-shot learning techniques, specifically the Siamese algorithm with a small dataset.

This is one of the first studies to investigate the use of Siamese architecture in the context of coin authentication, tackling the inherent challenges.

There are several methods have been developed for coin classification and coin detection in recent years, However, this study is the first research to implement a method to classify and detect counterfeit coins by using Few-shot learning algorithms and specifically the Siamese network with a small dataset. These methods have not been performed on coin detection before.

As part of our research, we collected and scanned new Chinese coins, Danish Coins, and Canadian coins with the Keyence 3D scanner in the CENPARMI Lab at Concordia University. The Keyence system is a 3D measurement system that uses high-intensity LED light and a 4-megapixel monochrome CMOS. This system is specifically designed to provide highly measurements images. A total number of 2,474 images were scanned. All steps to scan the images were completed by CENPARMI students and me. We set up the proper conditions including light reflection and scanner placement to ensure that there are no side effects on the images, and we have the optimal image quality. Each coin in this dataset was carefully scanned from both front and back views, covering both counterfeit and genuine coins. All the coins were captured with two resolutions, 12X and 40X. This is a high resolution to scan the 3D images and address the challenges posed by the small size of coins. This research is the second study using 3D images for detection of counterfeit coins, and all the previous ones were used 2D images.

To conduct our research, we obtained coins from the Law Enforcement Office provided to the CENPARMI students at Concordia University in Montreal, which includes their entire collection. It's important to note, however, that access to additional counterfeit coins is limited, hindering our investigative efforts. To address this limitation, we utilized a common augmentation technique as a compensatory measure. We will discuss this in detail in future chapters.

The data was split into a main, dateset containing 19 classes, and a target one with 3 classes. The Inception V3 network was fine-tuned in the main classes to learn general coin characteristics. In case the number of samples is insufficient for training a deep neural network, severe overfitting, and disappearance of the deep network gradient may happen. To address this issue, the Inception structure uses smaller convolution blocks instead of larger ones, thereby increasing the nonlinear expression ability of the model and making better use of parameters.

InceptionV3 is a powerful CNN architecture that is specifically designed for image classification tasks. It is well known for its ability to capture complex hierarchical features in images. InceptionV3 comes pre-trained on a large dataset, typically ImageNet, which enables the model to learn generic features from a diverse set of images. This pre-training can be advantageous when only a small dataset is available for a specific task [21, 22]. The information was transferred to the target classes. A few-shot learning using Siamese networks and contrastive loss was used. The main and target sets were each divided into a training set of 80% for developing the method and a test set of 20% for obtaining the results.

1.5 Outline

The rest of the thesis is structured as follows:

- Chapter 2 presents a comprehensive discussion of computer vision methods for dealing with various coin research problems and applications, including coin recognition, and counterfeit coin detection.
- Chapter 3 discusses image preprocessing methods and dataset preparation.
- Chapter 4 presents our proposed design and model, feature extraction methods, and the procedure of training the model.
- Chapter 5 provides experimental results.
- Chapter 6 concludes the thesis, summarizing the work completed and providing some insights and suggestions for the future work.

Chapter 2

Literature review

2.1 Coin detection

In recent years, numerous studies have been conducted to determine the difference between counterfeit and genuine coins. Several papers based on image processing techniques and classification algorithms have been published for counterfeit coin detection with different methods such as the Hough Transform, Gabor filter, Heuristics, and Artificial Neural Networks.

Identifying the counterfeit and genuine coins is an important task. However, it can be very challenging for those who are new to the field, especially in the case of rare or damaged coins. Even computer-based methods may face difficulties. Due to practical and inherent issues [23].

A mature recognition system typically consists of four main components: image capturing, pre-processing, feature extraction, and classification. Some prior studies also include a verification phase [24-26].

CENPARMI students at Concordia University in Montreal have developed several methods for detecting counterfeit coins.

The authors of [7] proposed a method for isolating individual letters and numbers on coins to study their features separately. Once the segmentation of letters was completed, four attributes were extracted from them, including letter width, smoothness, height, and width. Additionally, two characteristics between adjacent letters were investigated, such as relative distance and relative angle. The authors conducted experimental tests on two groups of coins to demonstrate the effectiveness of these features. They analyzed the lettering, images, and texture of coin faces to identify potential fake features. To separate the letters and digits from the image background and extract their features, they applied a novel shape feature, and a distinct region feature called the Maximally Stable Extremal Region (MSER) for texture analysis.

After capturing an image of a coin [7], the next step is segmentation. This involves separating the image into two parts. The background and the foreground. The background is usually a dark area that does not contain any useful information. To extract the foreground from the background, the image is detected and segmented using a technique called Hough Transform. This is the most used technique for this purpose.

• The steps of the hough transform method are as follows:

1. Obtain a binary edge image.
2. Specify the sub-divisions in x-y-r-plane.
3. Examine the counts of accumulator cells for high pixel concentration.
4. Search for the local maxima cells.

After segmentation, it is necessary to binarize it. All letters and digits are distributed in a circle in the center of the coin. To separate the letters, it is better to limit the image of the coin to the size of the ring, which contains only the letters. They tested sixteen coins. Eight coins are from the year 1990, and the others are from the year 1996. They faced some challenges during this project. The project posed several challenges, including detecting subtle patterns that differentiate similar objects. This is particularly difficult in pattern recognition applications where there are many similarities and differences between classes. The size of the coin was another challenge. Counterfeit coins are often indistinguishable from genuine ones based on size alone. Even visual inspections may not reveal fine details in texture and design, and there are no direct measurement tools for these features. Different forging techniques also create unique challenges. Fake coins do not have uniform features that differentiate them from genuine coins. However, counterfeit coins from the same source tend to share similar characteristics, whereas those from different sources do not.

Finally, the development of advanced counterfeiting technologies and the lack of expert knowledge make it more difficult to distinguish between counterfeit and genuine coins.

The method proposed in the paper [18] worked with a 3D approach. Most current methods for detecting fake coins are based on 2D images, which only provide statistical information about length and width and lose important characteristics like height and depth. Therefore, 3D techniques have become more popular in recognition, biometrics, security, and image processing. In this paper, the authors suggested a 3D method to detect and analyze the coin surface and extract important features. They introduced the Precipice Border Detection Algorithm (PBDA), which is not considered in previous methods. The authors extracted effective features based on the depth and height of a coin. For detecting the border of the coins, they used the Fuzzy C-Means algorithm [8, 27].

The major advantages of the method in [18] as follows:

- Suggesting a 3D Precipice Boundary Detection Algorithm (PBDA), Instead of the normal edge detection in 2D methods that can detect the precipice border of the coin's surface and be used for the technique of feature extraction.
- Degraded images don't require image enhancement or restoration.
- Binned Borders in Spherical Coordinates (BBSC) includes analyzing the direction and area of curved precipice borders. This method uses triangulation and fuzzy clustering to examine different border parts. By triangulating coin height images and extracting features from these triangles, a matrix of triangle samples is created.

The authors of [28] aimed to enhance the detection of counterfeit coins by utilizing deep learning techniques. They used a Generative Adversarial Network to generate fake coins for training purposes. To make the height-map images compatible with pre-trained networks, they proposed representing relief maps with three channels: Steep, Moderate, and Gentle slope (SMG). This generated a new channel for height-map images that can be utilized to train the pre-trained network. To increase the accuracy of the system, they proposed a hybrid method that combines fine-tuning pre-trained deep neural networks with a rejection option. The system delivered impressive results in coin classification. Additionally, the method can be utilized to detect coins that have not been previously seen by the model, whether they are genuine or counterfeit.

The authors of [14] proposed a technique for recognizing counterfeit 2-Euro coins that utilizes an optical mouse to take images. By comparing these images with a set of reference coins, the researchers were able to successfully identify counterfeit coins. The authors noted that the use of an optical mouse has many benefits, including its small size, affordability, and user-friendliness, which do not require specialized expertise. Nevertheless, it should be noted that the optical mouse only captures a portion of the coin's image, which may impact the accuracy rate and lead to misclassification.

The authors in [19] propose a method for detecting counterfeit coins based on image content and evaluate the effectiveness of different descriptors such as SIFT, SURF, and MSER. The study used the CENPARMI Danish coin dataset for experimentation and preprocessing was done to create counterfeit coins with slight shape differences from the original coins. The paper addressed some challenges, such as the complexity of processing color images and the need to convert RGB coin images to grayscale for certain steps. The authors proposed an autoencoding-based anomaly method that eliminates the need for fake data in training counterfeit coin detection models [29]. An autoencoder was trained to find anomalies in the coin images. The trained autoencoder received a coin image as input and generated a new image, which was compared with a basic image using the selected criterion.

Coins can be photographed under various conditions, with different contrasts, lighting, and backgrounds. These factors can significantly interfere with coin recognition. Deep learning models, which require a fixed, relatively small size of the input image, may lose essential coin features due to unskilled resizing of the input image. Therefore, before feeding the image to the coin recognition model, it's crucial to locate the coin and remove its background [13].

The authors of [1] proposed a method for detecting counterfeit coins using image-based techniques. The approach uses the dissimilarity space to represent the images of coins. This space is constructed by comparing the image with a set of prototypes. Local key points on each coin image are detected and described to measure the dissimilarity between the two images. Matched key points between the two images are identified based on the characteristics of the coin, enabling efficient detection. A post-processing procedure is used to eliminate mismatched key points. The proposed method uses only genuine coins for one-class learning, making it effective for fake coin detection. Extensive experiments have been conducted to evaluate the proposed approach on various datasets, demonstrating its validity and effectiveness. The paper also compares the clustering-based prototype selection with the random selection method and the RBF kernel with the linear kernel for one-class SVM. Additionally, experiments with different values of RBF kernel width and training error rate are presented.

The authors of [30] proposed a mechanism for detecting counterfeit coins, utilizing two distinct feature extraction techniques - Scale Invariant Feature Transform (SIFT) and Rotation and Flipping invariant Regional Binary Patterns (RFR). Additionally, the researcher developed an

Automatic Coin Grading system to identify and eliminate low-quality coins from the dataset. The approach involved acquiring digital images and applying computer vision and machine learning algorithms to analyze them.

In [30], the RGB images were first converted to grayscale, as grayscale images contain more information than black-and-white images. Subsequently, the images were segmented using hough transform to separate the foreground from the background. Then, some preprocessing was done to eliminate any redundant information. The author uses SIFT algorithm with a four-stage filtering process. Scale-space extrema detection stage obtains the location and scale of the object. In key point localization stage, any keypoint that has a low contrast from the extracted keypoints is removed. Orientation assignment, This stage considers the local image properties, assigns consistent orientation to keypoints, and represents each key point relative to it, making it rotationally invariant. Then key point descriptor creates keypoint descriptors using local gradient data, which are rotated and weighted by a Gaussian to align with the keypoint's orientation.

The author utilized a pattern recognition approach to identify the authenticity of coins based on their wear and tear over time. However, there were several challenges they encountered during this process. These included the small size of some coins, which made it difficult to differentiate genuine coins from counterfeit ones with precision. In addition, inconsistencies in the design of counterfeit coins posed another challenge, as different manufacturers use various methods to produce them. Furthermore, the quality of counterfeit coins and the advancement in counterfeit technologies are increasing every day, making it harder to identify them. Lastly, due to government restrictions, there is insufficient data available on fake coins, which makes research in this area more challenging for researchers.

The authors in [21] delved into the implementation of deep learning to classify plant leaves and emphasized the importance of a substantial number of samples for supervised training. The proposed approach employed the Siamese network framework, utilizing a parallel two-way convolutional neural network with weight sharing to extract features from distinct images. By training the network with a loss function, it learned a metric space where similar leaf samples are clustered together while dissimilar ones are separated. Through experimentation, the results demonstrate remarkable classification accuracy despite the limited number of supervised samples.

The process involved extracting features from two distinct images utilizing a parallel two-way convolutional neural network with weight sharing.

2.2 Classification

Coin classification is a crucial task in the field of numismatics with applications in historical research, coin grading, and automated coin sorting. Traditional methods relied on manual inspection and expert knowledge, but recent advances in computer vision and machine learning have enabled automation. The field has made significant progress through various machine learning and deep learning techniques. especially, siamese networks have emerged as an effective method for fine-grained visual recognition tasks, particularly when faced with limited data availability.

Siamese networks, first introduced by [31], refer to neural networks specifically engineered for the identification of similar or dissimilar image pairs. This is accomplished through a distinct architecture featuring twin networks with shared weights. Their efficacy lies in metric learning, rendering them particularly advantageous in discriminating between coins exhibiting subtle differences in coin classification. An increasingly promising approach to coin classification involves the utilization of siamese neural networks. Siamese neural networks represent a type of deep learning architecture that is well-suited for tasks centered on image similarity and comparison[32]. Siamese networks are characterized by a specialized architecture that enables them to acquire a representation of the input data that is sensitive to the intrinsic features of the images, as opposed to solely their superficial characteristics. This characteristic makes them well-suited for tasks such as coin classification, where the subtle distinctions between different types of coins pose a challenge for traditional classifiers[33].

Guo et al.[23] employed siamese networks in the classification of ancient coins, achieving high accuracy through the focused analysis of distinctive features in coin images. Their method involved establishing a feature embedding space to optimize the distances between similar and dissimilar coin images. Furthermore, research by Lorente et al.[34] provides additional evidence of the efficacy of siamese networks in coin classification. Collectively, these studies highlight the capability of siamese networks to achieve high levels of accuracy in coin classification, even when facing with challenging conditions such as low-resolution or noisy input images. The capacity of siamese networks to discern subtle differences in coin images makes them particularly suitable for this task, ensuring reliable classification across a diverse array of coin types and conditions.

The authors in [35] proposed the topic of Few-Shot Learning (FSL) algorithms applied to plant leaf classification using deep learning with small datasets. Through comparison with classical fine-tuning transfer learning, the paper concluded that FSL outperforms traditional methods when dealing with small training sets. To achieve this, the study employed the Inception V3 network, which is fine-tuned in the source domain to gain a better understanding of general plant leaf characteristics. This understanding is then carried over to the target domain to learn new leaf types from only a few images. The plant leaf image classification algorithm's architecture involves a general-purpose CNN image classification network that is fine-tuned to extract leaf image features or image embeddings. Following this, a shallow SVM classifier was trained to identify differences between the feature mappings for various plant leaf classes.

Chapter 3

Dataset preparation and image preprocessing

3.1 Dataset preparation

In this chapter, we will discuss how to create our datasets, the preprocessing steps involved, and the process of converting images into four slope images.

3.1.1 Keyence 3-D scanner

The Keyence VR-5000 is a 3D Scanner system developed by Keyence Corporation, a Japanese manufacturer of automation and inspection equipment. The VR-5000 Series is a 3D measurement system that uses high-intensity LED light and a 4-megapixel monochrome CMOS to capture a single fringe projection image of a wide area. This system is specifically designed to provide highly precise 3D measurements in various industrial applications. It uses laser technology to capture detailed three-dimensional data of objects, allowing for accurate measurements and inspections. Figure 3.1 shows the Keyense Scaner.

The VR-5000 Series uses scan optics to create fringe projection light via high-intensity LEDs built into the projection units. The structured (fringe projection) light passes through the telecentric projection lens and hits the object diagonally from above. When there are differences in height on the object's surface and when light is applied diagonally, the fringe projection image becomes distorted. The VR-5000 Series captures the distorted fringe projection image from directly above using the camera and measures the object's height from the distortion. To minimize the impact of the object's shape and orientation, the VR-5000 Series applies light from both the right and left projection units. Figure 3.2 shows the VR-5000 Series applies the light from both right and left projection units.

Figure 3.1: Keyence VR-5000 Scanner

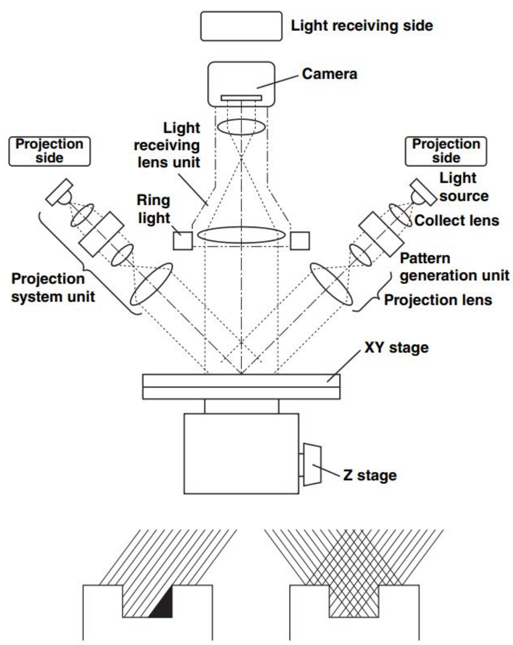


Figure 3.2: The VR-5000 Series applies the light from both right and left projection units. This reduces the impact of the shape and orientation of the object.

This study relied on both the dataset obtained using the Keyence 3-D scanner and the dataset based on IBIS TRAX scanner provided by CENPARMI.

3.1.2 Dataset with Keyence 3-D scanner

For the dataset captured with the Keyence 3-D scanner, we used the VR-5000 scanner from KEYENCE CANADA INC. This dataset includes a wide range of coins, such as Chinese coins like Dr. Sun Yat-sen-Memento 1927, Fat Man 1914, Phoenix and Dragon 1923, and One Dollar-Dragon 1911, Danish coins from 1996, and Canadian toonies coins from 1996. Table 3.1 shows the coins' dataset captured with the Keyence 3D. All steps to scan the images were completed by CENPARMI students and myself in the CENPARMI lab. We set up the proper conditions for light reflection and the scanner placement.

All the coins in this dataset were carefully scanned from both front and back views, covering both counterfeit and genuine coins.

Table 3.1: Coins' dataset captured with the keyence 3D

Coin	**Type\Name**	**Year**	**Number of Coins**
Canadian	Toonie	1996	57
Chinese	Dr. Sun-Yat-sen-Memento	1927	14
Chinese	Fat Man	1914	35
Chinese	One Dollar-Dragon	1911	6
Chinese	Phoenix and Dragon	1923	8
Danish	20 Kroner	1996	132

3.1.3 Dataset with IBIS TRAX

In this study, we also used the CENPARMI coin dataset [18] to investigate the authenticity of Danish coins from 1990, and 1996, as well as Canadian Toonies from 1996, and Half Yuan Chinese coins from 1942. The dataset was sourced from the Danish police, who provided both genuine and counterfeit coins to a local company, which subsequently submitted them to the CENPARMI laboratory for analysis. The images of the coins were captured by a very precise 3-D scanner in the name of IBIS TRAX, equipped with a built-in microscope and five groups of adjustable LEDs to facilitate the acquisition of high-quality images from various perspectives. The patent for this device is held by Ultra Electronics Forensic Technology Ltd. Company in Montreal. Table 3.2 shows a selection of coin images captured using the IBIS TRAX.

Table 3.2: CENPARMI coins' dataset captured with the IBIS TRAX scanner.

Coin	Type\Name	Year	Number of Coins
Canadian	Toonies	1996	75
Danish	20 Kroner	1990	125
Danish	20 Kroner	1996	109
Chinese	Half Yuan	1942	10

3.2 Image preprocessing

Data preprocessing, also known as data cleansing, is an essential phase in the machine learning process, and most ML engineers spend a significant amount of time on it before developing a model. Outlier detection, missing value treatments, and removing undesired or noisy data are a few examples of data preprocessing.

Image preprocessing refers to the processing of images that are performed at the most basic level of abstraction. If entropy is used as a measure of information, then these actions diminish rather than increase the information content of the image. Pre-processing aims to improve the picture data by reducing unwanted distortions or enhancing specific visual properties that are important for subsequent processing and analysis tasks [36].

Our dataset contains a lot of degraded and noisy coins, Therefore, to enhance the image quality and ensure better representation in our dataset, we implemented the following preprocessing techniques.

3.2.1 Data cleaning and resizing

It is common to encounter inaccuracies, defects, and errors that lead to inconsistencies when dealing with datasets. Therefore, a dataset is never entirely ready for processing. To achieve a perfect dataset, some action is necessary. In this study, none of the scanned coins were in perfect condition. Some coins were entirely damaged and had degraded to a point where they were difficult to process. For instance, some coins had a completely worn-out edge, and the image was not in perfect condition for preprocessing techniques and feature extraction in the upcoming steps.

We decided to remove some corrupted coin images from our dataset to ensure that we have an ideal training set for our proposed method. In our dataset, all coin images in the provided dataset were captured at high resolution. The original size of the Danish coins was 3550x3550 pixels and 1991x1982 pixels, the original size of the Canadian coins was 2976x2976 pixels and 1600x1274 pixels, and the original size of Chinese coins varied from 1755x1748 pixels for One Dollar Dragon to 1383x1373 pixels for Memento, and 1878x1803 pixels for Fatman. However, working with these large dimensions requires a lot of memory and is time-consuming. Therefore, to improve processing time and avoid memory issues, we reduced the size of the coin images to 128x128 pixels.

3.2.2 Transformation of the RGB images to grayscale and hough transform

The purpose of using filters is to change or improve the qualities of the images and to extract important data from the images, such as edges, corners, and blobs. A kernel, which is a tiny array applied to each pixel and its neighbors inside a picture, defines a filter. Brightness transformations improve pixel brightness, and it depends on the properties of a pixel, and it is important for both human and computer vision. Brightness corrections and grayscale transformations are two types of brightness transformations and in most of the recognition systems, gray or binary images are used because processing color images is computationally high. Also, some images contain backgrounds and watermarks that could make the recognition process difficult [37].

All the coins used in this study have circular shapes. Therefore, the Hough transform for circle recognition proposed by Reisert, et al. [38] is employed to segment round of coin images with the gray level of 0 to 255. Figure 3.3 shows the images of a Danish coin from 1996 and a fake ONE DOLLAR Dragon and the converted images to grayscale.

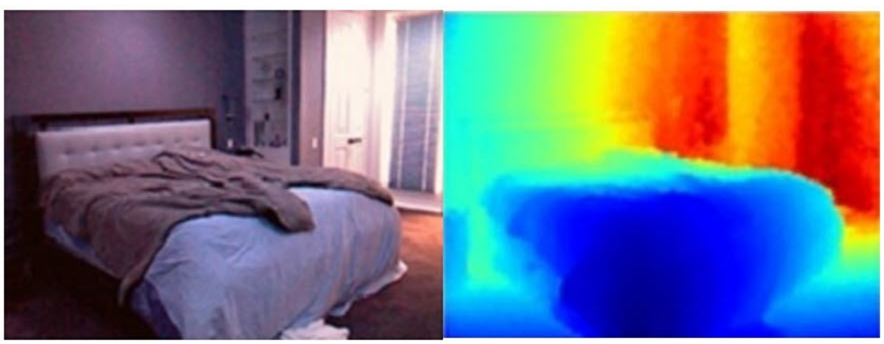
(a) (b)

Figure 3.3: (a) Sample of colored coin images, and (b) gray images

3.2.3 Transformation of the grayscale images to four slopes

3.2.3.1 Background and preliminary concepts

A height-map is a grayscale image that stores information about the distance or height of a surface from its background. The darker shades in the image represent shorter heights, while brighter shades represent longer heights. The minimum and maximum heights are represented in black and white respectively. A depth map is an image that shows the distance of objects in a scene from the camera's view and shows varying intensities that indicate the distance of each pixel. On the other hand, a surface normal is a set of three channels that show the orientation of each pixel in a scene. Each channel represents a direction cosine of the orientation vector for that pixel. Obtaining these characteristics from a single RGB image is a challenging task in computer vision, but it is essential [39]. Figure 3.4, shows a RGB color image and the corresponding depth colormap image, where blue indicates closer objects and red indicates farther objects [40].



(a) (b)

Figure 3.4: (a) RGB image, and (b) depth map image[40]

3.2.4 Transformation of the grayscale images to four slopes

As the detection of counterfeit coins is challenging due to their high similarity with genuine coins, more features are required before pre-training the neural network for classification purposes. To use the pre-trained models and grayscale images for fine-tuning, we converted grayscale images to four slope images. Therefore, having four channels carrying significant information can enhance the capability of networks for classification.

Based on the method used in [28] to convert the grayscale images to three slopes, we proposed a method to convert the grayscale images to four slopes images.

In this study with our image processing technique, we implement a procedure to analyze and categorize the slopes present within scanned images. The process applies through each pixel of the image repeatedly and forms triangles using neighboring pixels. By calculating the normal vector of these triangles, we determine the angle (θ) between the normal vector and a reference vector, allowing us to quantify the slope at each pixel location. Based on this angle (θ), pixels are categorized into specific slopes, including Steep slope displays with red color, Moderate slope displays with blue color, MId-moderate slope displays with light blue color, and Gentle slope displays with green color (SMMIG). According to the specifications listed in Table 3.3, a Gentle slope refers to any part of the coin surface where the angle is smaller than a threshold value called T1. On the other hand, a Moderate slope is defined by an angle between T1 and T2, while Mid-Moderate is a slope where the angle is between T2 and T3. Finally, the Steep slope is a section where the angle exceeds T3. These slopes provide a valuable understanding of the characteristics captured within the images. Furthermore, we use color representations to visually represent each slope category. The objective is to make it simple to understand and analyze the features that are visible in the images.

Table 3.3: Specification of the four classes

Angles	**Slope**
θ < T1	Gentle
T1 ≤ θ < T2	Moderate
T2 ≤ θ < T3	Mid-Moderate
θ ≥T3	Steep

In figure 3.5 the structure of the converting grayscale images to SMMIG images is displayed.

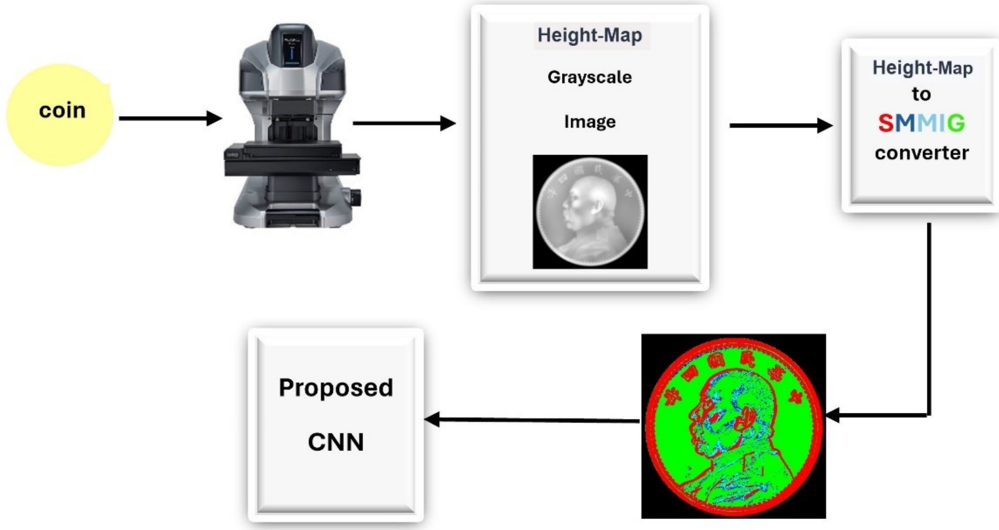


Figure 3.5: Structure of the converting grayscale images to SMMIG images

Figure 3.6 displays grayscale samples of both genuine and fake Danish 1996 coins, as well as the SMMIG result. This image clearly illustrates the distinct slopes present in different parts of the coin and shows that S, M, MI, and G matrices have no overlap in their elements. It means that each pixel in the image is uniquely categorized into one of these slope categories and is assigned to only one slope category, therefore avoiding confusion in the classification process.

(a) (b)

(c) (d)

Figure 3.6: Fake and genuine Danish 1996 coins. (a) grayscale height-map image of a genuine Danish 1996 coin, (b) an SMMIG image for the genuine coin. (c) grayscale height-map image of a fake Danish 1996 coin, and (d) an SMMIG image of the fake coin.

Figure 3.7 analyzes the normal vectors present on the surface of a coin to understand the variations in slope and categorize the slopes observed on the coin's surface into four types based on their steepness. Steep slopes are sharply inclined, moderate slopes have a moderate incline, mid-moderate slopes have a mid-moderate incline, and gentle slopes have a slight incline.

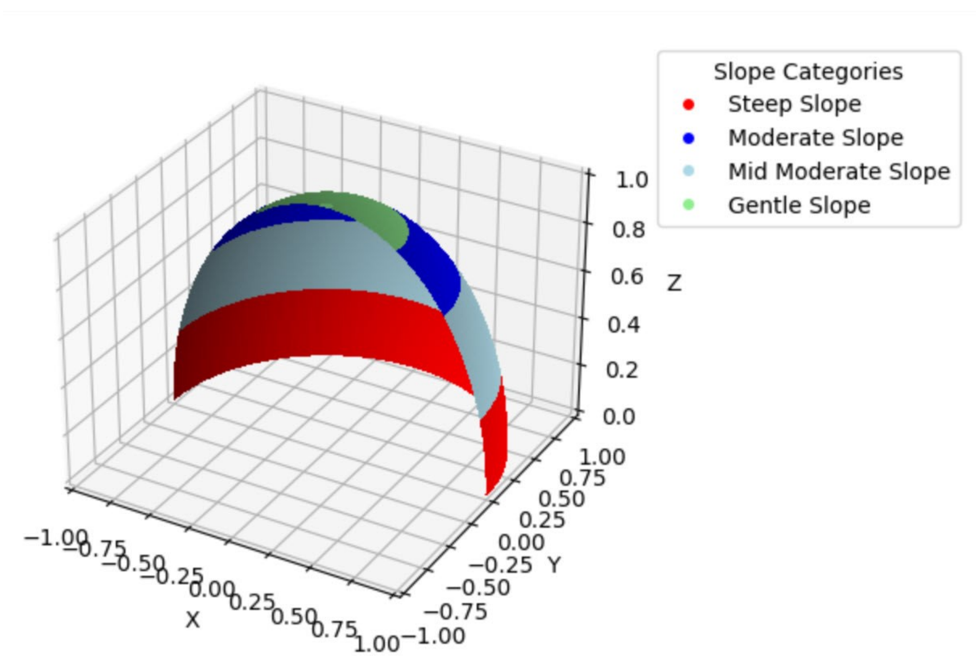


Figure 3.7: Normal vector distribution for the SMMIG channels

3.2.5 Augmentation

As previously discussed, we are facing with a lack of coins specially in Chinese coins and some degraded coin images in our Danish dataset, and obtaining more fake coins is restricted, which has imposed a significant challenge for this study. To compensate for this issue, we used a standard augmentation technique in our study.

Image augmentation is a technique widely used in computer vision tasks, especially for training deep learning models in tasks like image classification, object detection, segmentation, and more. This technique involves applying various transformations to the original images to create new training examples. Augmentation enriches the dataset and improves the model's ability to generalize [41, 42].

We applied various transformations such as random rotating an image by a certain degree angle of 20 to help the model learn to recognize objects from different viewpoints. Flipping the images horizontally and vertically to create mirror images, to help the model with learning symmetrical patterns. Shearing is applied to the input images with the range of between -0.2 and 0.2, and zooming involves cropping and resizing the original image to focus on specific regions. These techniques are applied randomly and in combination to ensure that the model receives a diverse range of images. However, it's crucial to ensure that the augmented images retain their semantic content and do not introduce unrealistic variations that could potentially confuse the model. Table 3.4 indicates six types of coins scanned by the Keyence 3-D scanner and IBIS TRAX scanner used in this model. Figure 3.8 shows the sample of input coin images in different types and Figure 3.9 shows the generated images by Augmentation.

Table 3.4: The specification of coins used in this study.

Coin	Type\Name	Number of Augmented
Chinese	Half Yuan-1942	92
Chinese	Dr. Sun-Yat-sen-Memento	469
Chinese	Fat Man	431
Chinese	One Dollar-Dragon	250
Chinese	Phoenix and Dragon	209
Danish	20 Kroner - 1996	27
Danish	20 Kroner - 1990	158

Figure 3.8: A selection of coin images which are provided as input to the augmentation.

Figure 3.9: Coin images generated by augmentation.

Chapter 4

A few-shot learning-based counterfeit coin detection method

4.1 Few-shot learning and siamese network

Few-shot learning (FSL) is one of the important topics in machine learning for training and developing a network with a few samples. In few-Shot Learning, a similarity score between input data and examples from each class is commonly employed for object classification. Metric-based networks, such as matching networks, and siamese networks, are frequently used in different classification tasks[43, 44]. The Siamese network structure can map the similarity relationship between different images into a metric method so that the samples related to the same category can be as close as possible, and the samples related to different categories can be as far as possible[31].

4.2 Proposed method

The method used in this project is trained in a supervised way, and the samples are extracted by a convolution neural network. Then, the Euclidean distance between features is calculated by a metric-based method. It means when the samples are more similar to each other the distance is closer. However, errors may occur in the formation of the measurement. The reason is that there are several similar types of coins in the training dataset, which makes it difficult to form a stable measurement space. For example, three similar samples will be in three different classes, and in the first stages of network training, a large number of samples match the requirements of acceptable samples. We will discuss more Euclidean distance in this project.

This study used a dataset consisting of 2,474 coin images belonging to 22 different classes. The dataset includes both genuine and counterfeit coin images on both obverse and reverse sides, The images were resized to dimensions of 128 × 128 pixels to standardize the experimental conditions. A representative illustration of the 22 classes is provided in Figure 4.1 for grayscale images, and

Figure 4.2 for SMMIG images. Table 4.1 and Table 4.2 offer a comprehensive overview of the coin types included in the dataset.

To enable experimentation, the dataset was split into two domains: a Main Dataset (D_m) and a Target Dataset (D_t). To reduce the complexity of training, we used a pre-trained model, which learned from the larger dataset (D_m) and obtained a better understanding of general coin characteristics. This understanding is then transferred to the target domain (D_t) to learn new coin types from a few images. The main dataset, which aimed to establish a baseline coin classification algorithm, included nineteen classes, totaling 2,143 images. In contrast, the target dataset comprised the remaining three classes, which include 331 images and served as the dataset for the development and evaluation of Few-Shot Learning (FSL) algorithms.

Each dataset was randomly partitioned into the training set (80%) and validation set (20%). To prevent data leakage, images originating from the same coin but captured at different orientations and/or conditions were consolidated into the same partition. This approach aimed to ensure the integrity of the experimental design and the effectiveness of the developed algorithms.

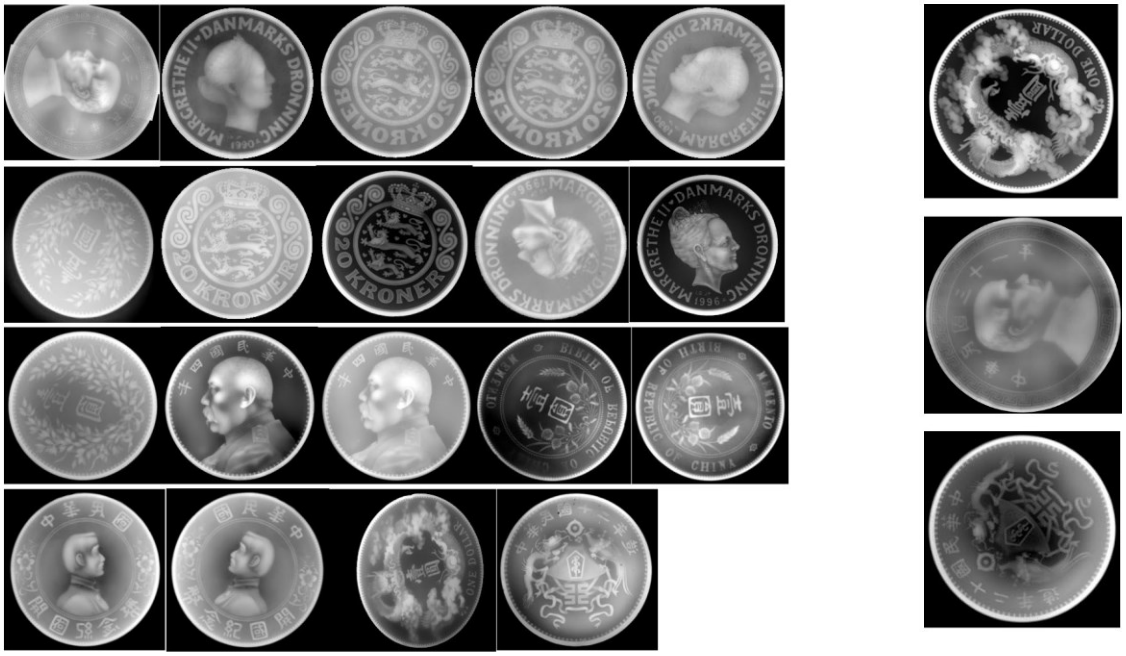
Table 4.1: The properties of the main dataset

No.	Coin	Type/Name	Dataset	Images
1	Chinese	Half Yuan-1942-(Genuine)	D_m	48
2	Danish	20 Kroner 1990-Queen-(Fake)	D_m	143
3	Danish	20 Kroner 1990-Back-(Fake)	D_m	100
4	Danish	20 Kroner 1990-Back-(Genuine)	D_m	64
5	Danish	20 Kroner 1990-Queen-(Genuine)	D_m	101
6	Danish	20 Kroner 1996-Back-(Fake)	D_m	191
7	Danish	20 Kroner 1996-Back-(Genuine)	D_m	57
8	Danish	20 Kroner 1996-Queen-(Fake)	D_m	190
9	Danish	20 Kroner 1996-Queen-(Genuine)	D_m	55
10	Chinese	Fat Man-Back-(Fake)	D_m	171
11	Chinese	Fat Man-Back-(Genuine)	D_m	74
12	Chinese	Fat Man-(Fake)	D_m	167

13	Chinese	Fat Man-(Genuine)	D_m	89
14	Chinese	Dr. Sun-Yat-sen-Memento-Back-(Fake)	D_m	182
15	Chinese	Dr. Sun-Yat-sen-Memento-Back-(Genuine)	D_m	83
16	Chinese	Dr. Sun-Yat-sen-Memento -(Fake)	D_m	113
17	Chinese	Dr. Sun-Yat-sen-Memento-(Genuine)	D_m	119
18	Chinese	One Dollar-Dragon-(Genuine)	D_m	109
19	Chinese	Phoenix and Dragon-(Genuine)	D_m	87

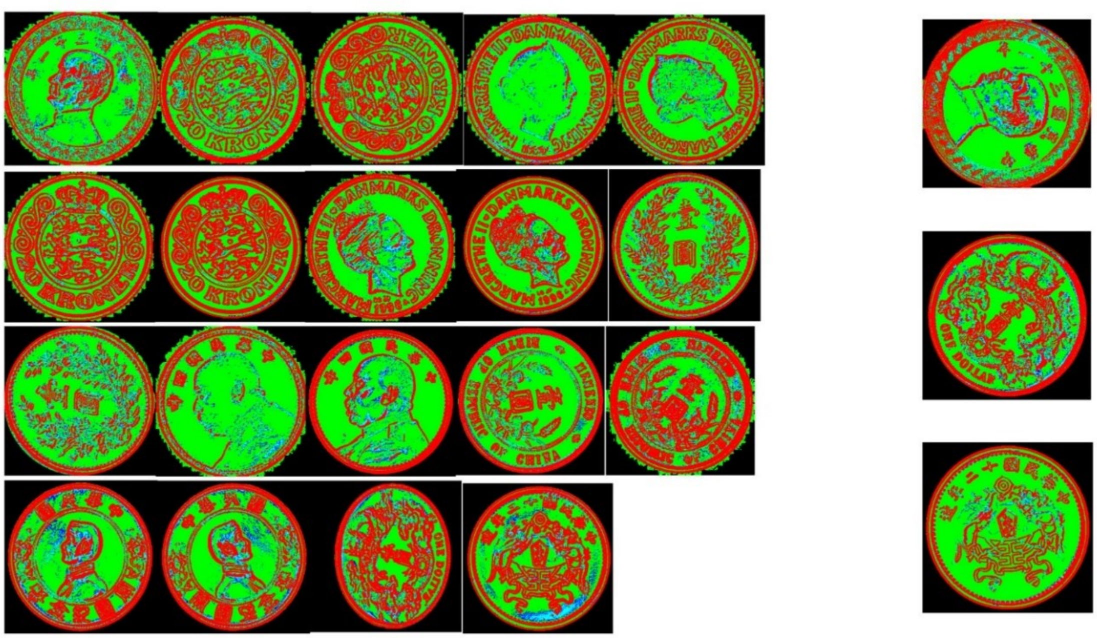
Table 4.2: The properties of the target dataset

No.	**Coin**	**Type/Name**	**Dataset**	**Image**
20	Chinese	Half Yuan-1942-(Fake)	D_t	54
21	Chinese	One Dollar-Dragon-(Fake)	D_t	147
22	Chinese	Phoenix and Dragon-(Fake)	D_t	130



(a) Main Dataset, D_m (b) Target Dataset, D_t

Figure 4.1: Examples from the 22 classes in our grayscale images dataset



(a) Main Dataset, D_m (b) Target Dataset, D_t

Figure 4.2: Examples from the 22 classes in our SMMIG images dataset.

4.2.1 Baseline Convolutional Neural Networks (CNNs)

The most effective image classification results are achieved through deep learning algorithms based on CNNs, which may contain thousands or even millions of tunable parameters [35]. CNNs have become a crucial component of modern image analysis, representing the essence of artificial intelligence in interpreting visual content. Resembling the intricate processes of the human visual system, CNNs operate through a series of layers that progressively extract and interpret hierarchical features from images. Starting with rudimentary elements such as edges and colors, CNNs navigate deeper layers to identify complex structures such as shapes, textures, and objects. By using convolutional operations and learned parameters, these networks encode elaborate representations of visual data, facilitating tasks such as image classification, object detection, and semantic segmentation [45].

We used CNN with the InceptionV3 model pre-trained on the ImageNet dataset [46] with the model architecture excluding fully connected layers and specifying input shape for images of 128x128 pixels. InceptionV3 is known for its high performance in image classification tasks and models like inceptionV3 which has been pre-trained on large datasets like ImageNet, are often a good choice for small datasets [21, 35]. A global average pooling layer is added to the base InceptionV3 model output, creating a feature extractor model using the Keras Model. This is followed by another dense layer with ReLU activation and a predictions layer with SoftMax activation, producing predictions for 19 classes. The final model is constructed using the Keras Model, specifying the inputs and outputs. To retain learned representations, the layers of the base InceptionV3 model are frozen during the fine-tuning process. The network contains 311 layers and to fine-tune the network for Coins detection the first 249 layers were frozen, while 62 layers were set to be trainable. This decision arises from the specific architecture and configuration of the InceptionV3 model. This approach helps fine-tune the model for a specific classification task while preserving the learned representations. Generally, using neural network classifiers for optimization leads to serious overfitting because the number of features is insufficient and because the neural network classifier has a large number of parameters to be optimized. It is noteworthy to acknowledge that creating a suitable classifier is necessary.

Deep neural networks often contain many parameters. When a pre-trained network is fine-tuned or used as a feature extractor on a different but related task, it often outperforms an initialized network trained from scratch and the performance is particularly noticeable. Our pre-trained model (ImageNet) is a dataset with a large-scale computer vision benchmark that includes hundreds and thousands of images for the visual recognition challenge. According to the WordNet hierarchy, the ImageNet dataset contains 14,197,122 images. Pre-training is absolutely useful in the few-shot learning method due to the small size of the dataset.

4.2.2 Siamese network architectures

This study uses a training architecture that focuses on Siamese networks, a specialized neural network configuration designed for comparing and matching pairs of inputs that consist of two identical branches that share weights and parameters. The architecture comprises two identical subnets, drawn from the Inception V3 architecture, featuring 62 tunable layers, and constitutes the backbone of the Siamese network. The weight-sharing mechanism between these subnets facilitates a collaborative learning process, enabling the network to discern meaningful patterns and representations within the input images. Figure 4.3 is the architecture of the generic Siamese model.

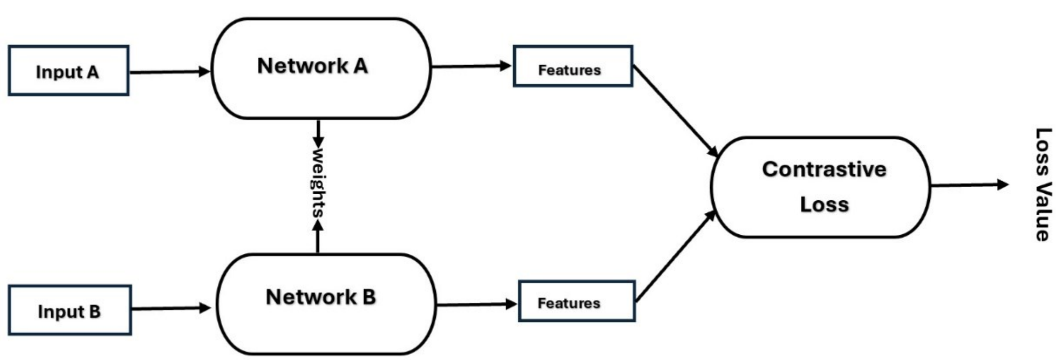


Figure 4.3: The architecture of the generic siamese model.

In order to improve the performance of the network, we use a cost function that can differentiate between pairs. This function encourages similar examples to be close to each other, and dissimilar ones to be placed at least a certain distance apart from each other, as measured by Euclidean distance. During training, the adoption of contrastive loss serves as a pivotal component, guiding the network to minimize the embedding distances between similar image pairs and maximize those between dissimilar pairs. This approach aims to enhance the siamese network with a discriminative ability crucial for tasks such as image similarity and matching. Figure 4.4 shows the architecture of the siamese network for the two input images based on the contrastive loss function.

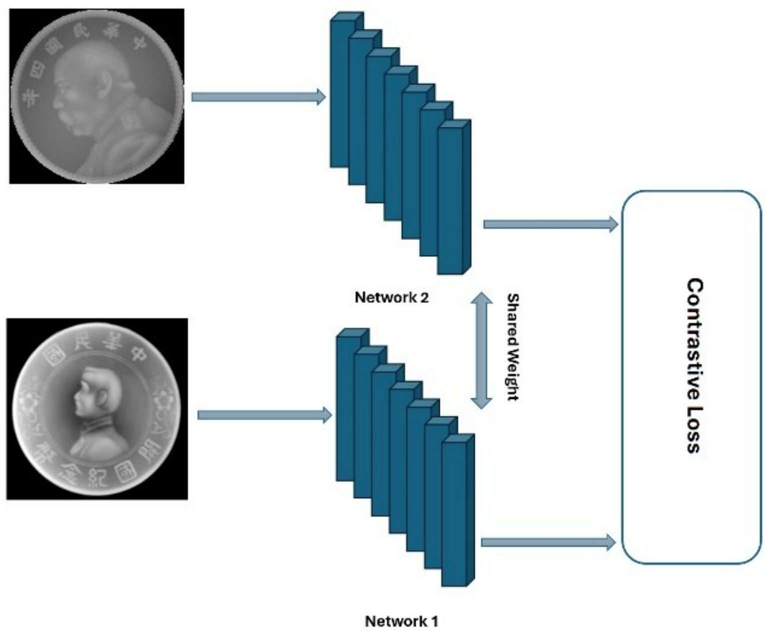


Figure 4.4: Architecture of the siamese network for the two input images based on the contrastive loss function.

The chosen methodology does not only align with the baseline fine-tuning model but also provides a framework for a comparative analysis to evaluate the efficacy of siamese networks in learning image embedding. Notably, the methodology used in this study aligns with current research and extends the literature by demonstrating the effectiveness of siamese networks in learning image embeddings.

In a subnet siamese network, during the training phase, a pair of images X_i and X_j are fed to the network, and the contrastive loss function for the pair is calculated using the euclidean distance represented as $\|\cdot\|^2$. The CNN learns how to map a coin image X_i to $f_i = f(X_i)$. The margin m is set to differentiate between the same class (y = 1) and different classes (y = 0). By minimizing the loss function, the network learns to reduce the distance between embeddings for similar classes and increase the distance between embeddings for different classes up to the margin m. To implement this, we use the margin-based contrastive loss function proposed in [47] which is defined in Eq (4.1) as follows:

$$\varphi_c(X_i, X_j) = y \cdot \|f_i - f_j\|^2 + (1 - y) \cdot \max(0, m - \|f_i - f_j\|^2) \qquad \text{Eq (4.1)}$$

4.2.3 Support Vector Machines (SVM) classifier

Support vector machines are a collection of supervised learning techniques used for classification. SVMs select the decision boundary by maximizing the distance from the nearest data points of all classes. This decision is known as the maximum margin or gap classifier.

One of the reasons why Support Vector Machines (SVMs) are used in our project is because they can identify and find complicated relationships within data without the need for a lot of data transformations. This makes SVM an excellent option especially when working with smaller datasets like our datasets that have several features, providing more precise results compared to other algorithms [48].

In this project, the SVM (Support Vector Machine) classifier is used as a binary classifier for the features extracted by a pre-trained neural network and is initialized with a linear kernel to find the linear decision boundary that best separates the classes in the feature space.

4.2.4 Few-shots learning architecture and transfer learning

Determining the number of trainable layers and the frozen layers in fine-tuning our neural network was one of the important steps and it depended on the architecture of our model. Typically, our goal was to fine-tune the later layers which are closer to the output, and the layers that extract higher-level features related to the project and did not keep the earlier layers, which extract more general features. Due to the small size of our dataset, we first started by freezing more layers to prevent overfitting and help our model to learn specific features and not the general features and kept the late layers which allowed us to update during the training. In this case, the late layers are only the trainable layers and will update during computation, the frozen layers are set as untrainable and remained as fixed to prevent their weights from being updated. The second reason to freeze more layers was due to limited computational resources. Since fine-tuning more layers requires more computations and has a bad effect on the training speed. We experimented with the different balances between the frozen layers and trainable layers to find the optimal value and monitored the performance of the fine-tuned model on the validation set.

The SVM classifier is designed to maximize the gap between the features of different classes [48]. For the proposed method, SVM with multiclass classification is used. Each SVM is trained to one particular class or positive class and the other classes are considered as negative classes. Each class is trained individually. Each classifier was assigned a decision value when all classes were trained. Then the final prediction is determined by selecting the class with the highest value [49, 50].

To transfer the knowledge from the main dataset D_m to the target dataset D_t, the SVM classifier was re-trained with the images of the target dataset. Each class includes a different number of images, and a typical fine-tuning was used. Transfer learning refers to training the pre-train model and using the knowledge gained in the smaller dataset or target dataset (D_t). Pre-training is essential for small datasets to prevent overfitting, and the network can be optimized quickly with less training. In the proposed method, the last frozen layers were trained on the D_m and then fine tuned with the D_t dataset. Using transfer learning techniques like gradually unfreezing, focusing on the particular layers and contrastive learning rates can help fine-tuning the model more efficiently.

Chapter 5

Experimental result and discussion

5.1 Performance metrics

In the proposed algorithm, we implement the training of the network by transfer learning. The traditional methods suggest larger batch sizes and more epochs during model training, however, large batch sizes and more epochs are demanding computational resources more efficiently, particularly when training on GPUs. On the other hand, extremely large batch sizes may lead to memory restrictions. Batch sizes between 16 and 128 are commonly used. Smaller batch sizes can introduce more noise, while larger batch sizes may provide more stable gradients. During the process, monitoring the training and validation loss is important. If the model training and validation loss is high that means underfitting, training with more epochs may be beneficial, while if the model training and validation loss continues to decrease, overfitting may happen, so reducing the number of epochs is useful. Another technique we used to prevent overfitting is early stopping. It means that stop training when the validation loss stops improving or starts to degrade. Larger datasets may require more epochs for the model, while smaller datasets may converge faster and require less epochs. In general, finding the proper batch size and number of epochs requires balancing computational efficiency, convergence behavior, and generalization performance. Monitoring key metrics during training is essential for determining the optimal values for our scenario.

5.2 Experimental setup

The experiment is performed on the Concordia GPU Cluster environment. The cluster uses SLURM resource management and a job scheduling engine. We interacted with SLURM from the Submit node used to prepare and submit jobs. Jobs get executed on Compute nodes according to available resources. The hardware configuration is as follows: Seven nodes with 4x 80GB A100 GPUs, sliced into 4x 20GB MIGs. Twenty-four, 32-core nodes, each with 512 GB of memory and approximately 10 TB of volatile-scratch disk space. Twelve NVIDIA Tesla P6 GPUs, with 16 GB of memory. One AMD FirePro S7150 GPUs, with 8 GB of memory. One node with six (6) V100 GPUs. Job Management is handled by the Slurm Workload Manager. The software platform is the Linux operating system, CUDA version 12.1.1. The programming language is Python version 3.11, and the deep learning framework is Keras with a TensorFlow backend.

5.3 Network optimization

5.3.1 Impact of epoch

In this study, we conducted training experiments with different numbers of epochs to investigate the impact of epoch count on the performance of our counterfeit coin detection model. Specifically, we trained our model using five different epoch settings: 10, 15, 20, 25, and 30. The choice of these epoch values was motivated by the need to explore a range of training durations and assess how longer training periods affect the model's ability to learn discriminative features for counterfeit coin detection. By systematically varying the number of epochs, our goal was to identify the optimal training duration that maximizes model performance while avoiding overfitting the training data. The accuracy and loss calculated during the model's training and validation are used to evaluate the model. To evaluate the effectiveness of different epoch settings, we analyzed the performance metrics of our counterfeit coin detection model on both grayscale and SMMIG images across each epoch on the validation set. A summary of the performance metrics obtained for each epoch setting on grayscale and SMMIG (Steep, Moderate, Mid-Moderate, Gentle) images are displayed in Table 5.1 and Table 5.2.

Table 5.1: Comparison of training epochs vs. accuracy for grayscale images.

	Trained with GrayScale Images				
Epoch	10	15	20	25	30
Accuracy (%)	51.18	79.54	92.13	92.31	92.55

Table 5.2: Comparison of training epochs vs. accuracy for SMMIG images.

	Trained with SMMIG Images				
Epoch	10	15	20	25	30
Accuracy (%)	59.23	82.66	94.73	94.80	94.91

In Tables 5.1 and 5.2 we demonstrated that the comparison highlights the effect of training epochs and dataset characteristics on model accuracy. For the model trained on grayscale images, there is a noticeable improvement in accuracy as the number of epochs increases. Particularly, the accuracy increases from 51.18% at epoch 10 to 92.13% at epoch 20. Beyond epoch 20, the accuracy remains relatively stable between 92.31% to 92.55%. Similarly, for the model trained on SMMIG images, there is a consistent improvement in accuracy with increasing epochs. The accuracy rises from 59.23% at epoch 10 to 94.91% at epoch 30. Notably, accuracy surpasses 90% at epoch 15 and continues to improve, indicating the effectiveness of longer training durations. Across all epoch settings, the model trained on the SMMIG images consistently outperforms the model trained on grayscale images in terms of accuracy. This suggests that the SMMIG images, with their color information and additional features, provide more information for the model to learn from compared to grayscale images.

Based on the observed accuracy improvement and computational efficiency, after analyzing the performance of our counterfeit coin detection model across different epoch settings, we observed that while there is a consistent improvement in accuracy with increasing epochs, the rate of improvement decreases significantly after epoch 20. Beyond this point, the incremental gains in accuracy become marginal, indicating decreasing returns in performance despite longer training durations. Furthermore, it's important to consider the computational resources required for training the model. As the number of epochs increases, so does the processing time and computational complexity. Given the trade-off between accuracy improvement and computational cost, we deemed epoch 20 as the optimal training duration. At this point, the model achieves a high level of accuracy (92.13% for grayscale images and 94.73% for SMMIG images), while minimizing the computational burden compared to training for longer durations. Therefore, by selecting epoch 20 as the optimal training duration, we achieved a balance between model performance and computational efficiency. This ensures that our proposed model achieves satisfactory accuracy without incurring excessive processing overhead. Figure 5.1 shows the impact of different epoch settings on the model performance.

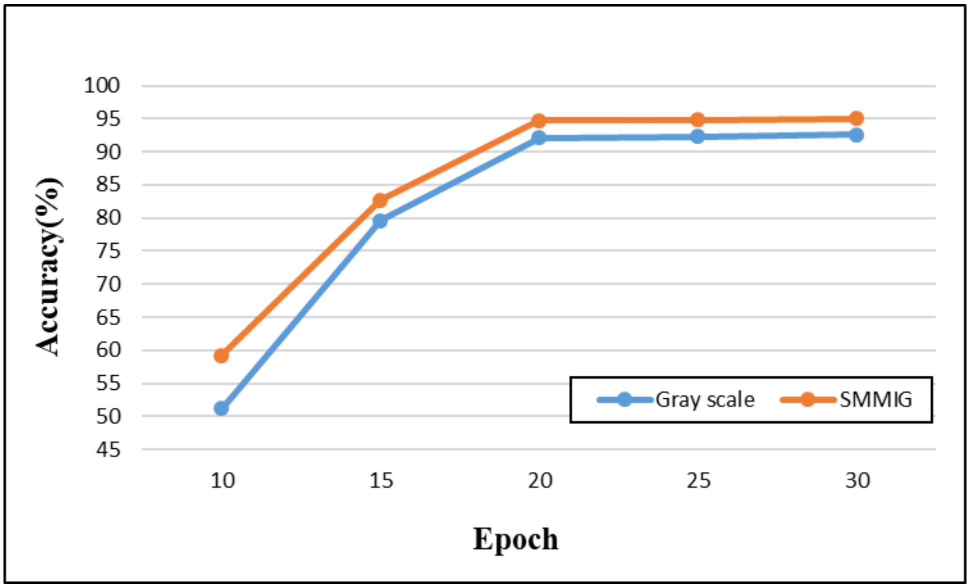


Figure 5.1: Impact of different epoch settings on the model performance.

5.3.2 Impact of batch size

Tables 5.3 and 5.4 present the accuracy (%) achieved by the proposed method when trained with grayscale and SMMIG images using different batch sizes. Across different batch sizes, we observed variations in accuracy. Notably, the model trained with a batch size of 32 achieves the highest accuracy of 92.13%, indicating that moderate batch sizes contribute to optimal performance. Batch sizes of 16 and 64 also yield relatively high accuracies of 69.19% and 86.92%, respectively, however a larger batch size of 128 results in a decrease in accuracy to 80.04%. Similarly, for the model trained with SMMIG images, we observed variations in accuracy across different batch sizes. The highest accuracy of 94.73% is achieved with a batch size of 32, indicating that moderate batch sizes are conducive to optimal performance. Batch sizes of 16, 64, and 128 also yield relatively high accuracies of 72.49%, 88.20%, and 81.71%, respectively. Across all batch sizes, the model trained on the SMMIG dataset consistently outperforms the model trained on grayscale images in terms of accuracy. This suggests that the SMMIG images provide richer information for the model to learn from compared to grayscale images. The difference in accuracy between grayscale and SMMIG models highlights the importance of dataset characteristics in model training. For both grayscale and SMMIG images, the highest accuracies are achieved with moderate batch sizes (32). This shows that a batch size of 32 achieves a balance between computational efficiency and model performance, making it possible for the model to effectively learn from the training data without very large computational resources. Figure 5.2 shows the impact of different epoch settings on the model performance.

Table 5.3: Comparison of training batch sizes vs. accuracy for grayscale images.

	Trained with GrayScale Images			
Batch Size	16	32	64	128
Accuracy (%)	69.19	92.13	86.92	80.04

Table 5.4: Comparison of training batch sizes vs. accuracy for SMMIG images.

	Trained with SMMIG Images			
Batch Size	16	32	64	128
Accuracy (%)	72.49	94.73	88.20	81.71

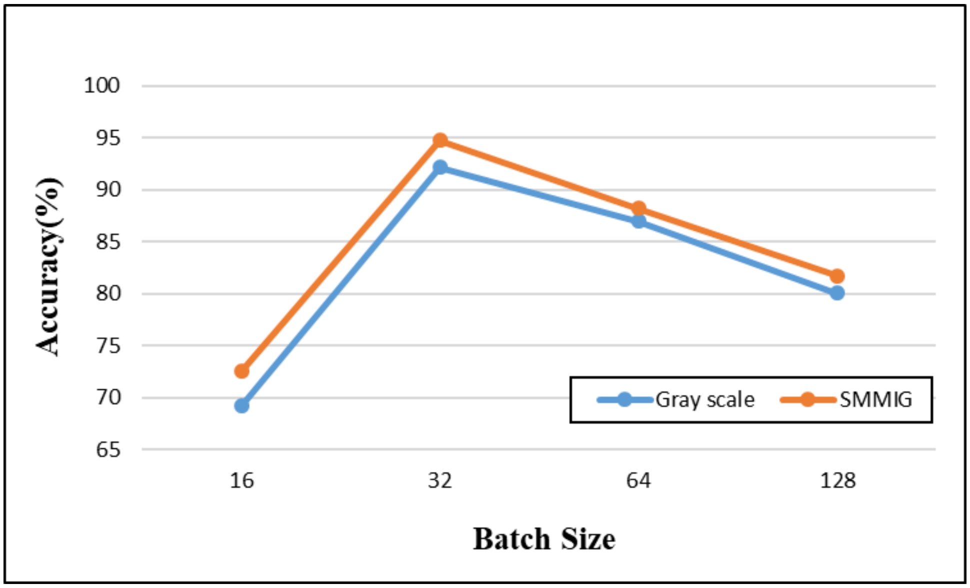


Figure 5.2: Impact of different batch size settings on the model performance.

5.4 Comparison with other methods

Table 5.5 shows the comparison between our proposed method and recent studies published in counterfeit coin detection methods. 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 five datasets and at the end the case of all our datasets. In this comparison, we trained the machine with grayscale, two slopes, three slopes[28], and four slope images. The SMMIG method has higher accuracy than other methods.

Table 5.5: Comparison of the proposed method and some other methods in terms of accuracy

Dataset	Grayscale images	2 Slopes images	[28] SMG images	SMMIG images
Half Yuan-1942	87.13	87.25	88.1	89.92
20 Kroner 1990	98.31	98.44	98.5	99.25
20 Kroner 1996	97.57	97.89	98.05	98.11
Fat Man	92.78	93.1	93.9	95.73
Dr. Sun-Yat-sen-Memento	89.45	89.11	90.8	94.28
All Dataset	92.13	92.2	93.49	94.73

5.5 Performances of the proposed model

To assess a classification system, there are several standard metrics. Precision measures the quality of model predictions for one specific class, and it focuses on the accuracy of positive predictions. On the other hand, Recall measures the model's performance for the actual observations of a specific class, and it focuses on the ability of the classifier to find all positive instances. These metrics are defined for one class at a time and emphasize a specific class. To calculate Recall (Ri), Precision (Pi), and F-score for each class, we followed these steps: first, we calculated True Positives (TP) number of samples correctly predicted as positive (belonging to the class), False Positives (FP) number of samples incorrectly predicted as positive (predicted to belong to the class, but actually belong to a different class), False Negatives (FN) number of samples incorrectly predicted as negative (predicted not to belong to the class, but actually belong to the class), and True Negatives (TN) the number of samples that are correctly predicted as negative (not belonging to the class). For our counterfeit coin detector, a TP shows that the system detects a genuine coin correctly. A TN shows that the counterfeit coin detector has correctly detected a counterfeit coin. The FP is when a counterfeit coin is falsely classified as a genuine one. The FN shows that when a genuine coin is classified as a counterfeit one incorrectly. F-Score provides a balance between precision and recall, considering both false positives and false negatives. Precision, Recall, and F-Score can be computed as Eq (5.1), Eq (5.2), and Eq (5.3) respectively:

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive+False Positive}} \qquad \text{Eq (5.1)}$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive+False Negative}} \qquad \text{Eq (5.2)}$$

$$F - \text{Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \qquad \text{Eq (5.3)}$$

We evaluated our counterfeit coin detection methods by testing a different test set with the grayscale images and SMMIG images. in our test, each class includes 10 images. We repeated the process 10 times to ensure statistical robustness. For each test, we calculated recall, precision, and F-score to assess the model's performance. In this scenario, we compared our method using grayscale images with SMMIG. The results for precision, recall, and F-score are displayed in Table 5.6.

Figure 5.3 shows the precision, recall, and F-score for 19-class models in our test subset of the main dataset. The classes are sorted by ascending F-score values. The number of classes with F-scores below 70% was 4 and it is shown in Figure 5.3 that recall and precision are above 50% for all classes.

Table 5.6: Comparing the results of grayscale images and SMMIG images in terms of precision, recall, and f-score

Class number	Grayscale			SMMIG		
	Recall%	Precision%	F-Score%	Recall%	Precision%	F-Score%
2	63.45	52.65	57.55	61.35	51	57.7
15	58.39	61.7	60	59.9	61.89	60.88
14	71.45	61.29	65.98	72.49	60.54	65.98
13	68.09	65.08	66.55	73.45	65	68.97
11	65.06	74.29	69.37	65.84	75.12	70.18
1	71	74.47	72.69	71.85	75.2	73.49
9	74.53	75.78	75.15	77.2	73.77	75.45
18	88.13	68.35	76.99	88.09	68.53	77.09
7	68.87	88	77.27	70.02	92.13	79.57
3	83.92	77.13	80.38	84.62	78.38	81.38
4	80.22	81.83	81.02	81.34	83.53	82.42
10	80	83.88	81.89	84.06	83.8	83.93
17	87.18	80.89	83.92	89.78	80.02	84.62
16	84.18	90.03	87.01	83.65	92.82	88
12	92.96	89.1	90.99	89.82	89	89.41
6	94.07	95.62	94.84	93.48	95.62	94.54
19	93.38	93.23	93.8	97.03	92.93	94.94
8	96.33	95.79	96.06	96.13	98.01	97.06
5	96.84	97	96.92	97.24	97.7	97.47

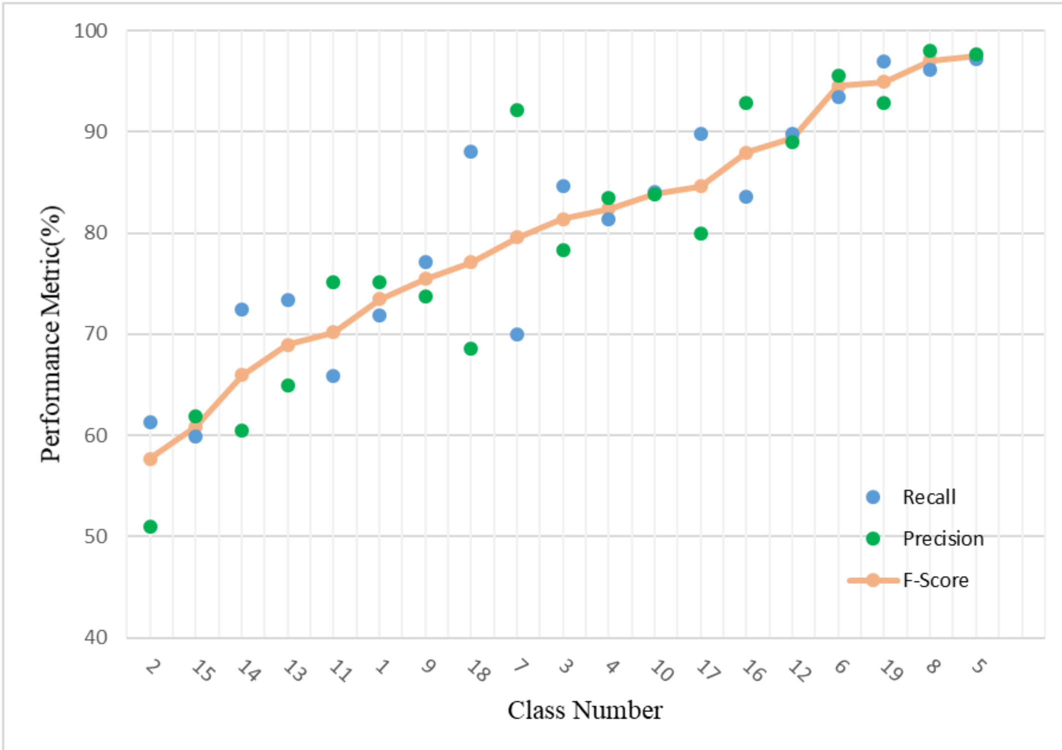


Figure 5.3: Performance metric per class for SMMIG dataset in the test set of the D_m. The classes are sorted by ascending F-score values.

Chapter 6

Conclusion and future work

6.1 Conclusion

In this study, an improved network structure is proposed to solve the problem of coin detection and classification in the case of small samples. This is of great significance especially when addressing the challenge of sparse data samples or various classification tasks. This study is applied to two different images. The first set consists of the images converted to grayscale, and the second set consists of the SMMIG (Steep, Moderate, MidModerate, and Gentle) images. We converted the grayscale images to four slope images using the proposed method to provide the SMMIG dataset. This study shows that the few-shot learning method by using contrastive loss and efficient class boundary has improved the method for counterfeit coin detection. The few-shot learning architecture designed for this study is based on the InceptionV3 network. Other embedding extraction networks could also be used such as Resnet or VGG. In fact, the author of [51] compared the InceptionV3, ResNet, VGG, and MobileNet architectures for a coin classification, and showed that the InceptionV3 network produced the highest classification accuracies and the computationally efficient and higher-performance architecture was Inception V3. After training a general CNN to learn and extract coin characteristics, our method showed an accuracy above 92%. The machine that is trained with SMMIG images has a higher accuracy than grayscale images. This project demonstrates that it is possible to develop accurate new algorithms to identify coin detection methods with very few annotated training images. These few-shot learning methods can substantially reduce the cost of new developing methods for counterfeit coin detection. The proposed method is capable performing of coin detection and coin classification of any type of coin and has remarkable accuracy. Although deep learning techniques are growing so fast, they can be very effective in the proposed method. However, fake and genuine samples for each type of coin are required to train the model in this method.

6.2 Future work

Although this method has achieved good results, for future work, it could be interesting and beneficial to extend the SMMIG images to more slopes, especially in medical images and face recognition projects. We can divide the coin images into several circular/annular sectors, and try with two or three rings to access more features.

References

Uncategorized References

[1] L. Liu, Y. Lu, and C. Y. Suen, "An image-based approach to detection of fake coins," *IEEE Transactions on Information forensics and security,* vol. 12, no. 5, pp. 1227-1239, 2017.

[2] https://www.royalmint.com/discover/uk-coins/counterfeit-one-pound-coins/ (accessed.

[3] J. Harris, *An essay upon money and coins*. Cambridge University Press, 2017.

[4] J. Snell and J. Theeuwes, "Finding counterfeited banknotes: the roles of vision and touch," *Cognitive Research,* vol. 5, no. 1, 2020.

[5] W. Herbert and G. Maier, "Magnetic ink composition, magnetic ink character recognition process, and magnetically readable structures," *EP2390292 B1,* 2013.

[6] A. Gavrijaseva, O. Martens, and R. Land, "Acoustic spectrum analysis of genuine and counterfeit euro coins," *Elektronika ir Elektrotechnika,* vol. 21, no. 3, pp. 59-57, 2015.

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

[8] S. Khazaee, M. Sharifi Rad, and C. Y. Suen, "Detection of counterfeit coins based on modeling and restoration of 3D images," in *Computational Modeling of Objects Presented in Images. Fundamentals, Methods, and Applications: 5th International Symposium, CompIMAGE 2016, Niagara Falls, NY, USA, September 21-23, 2016, Revised Selected Papers 5*, 2017: Springer, pp. 178-193.

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

[10] H. Anwar, S. Zambanini, and M. Kampel, "Supporting ancient coin classification by image-based reverse side symbol recognition," in *Computer Analysis of Images and Patterns: 15th International Conference, CAIP 2013, York, UK, August 27-29, 2013, Proceedings, Part II 15*, 2013: Springer, pp. 17-25.

[11] S. Zambanini and M. Kampel, "Automatic coin classification by image matching," in *Proceedings of the 12th International conference on Virtual Reality, Archaeology and Cultural Heritage*, 2011, pp. 65-72.

[12] C. Gagg and P. Lewis, "Counterfeit coin of the realm–Review and case study analysis," *Engineering Failure Analysis,* vol. 14, no. 6, pp. 1144-1152, 2007.

[13] M. D. Eugene Steinberg, Nikita Kaptsove, Timofey Emelyanov. "How to recognize coins with deep learning visual model." https://blog.griddynamics.com/how-to-recognize-coins-with-deep-learning-visual-model/ (accessed.

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

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

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

[17] X. Jin, X. Wang, X. Cao, and C. Xue, "Construction and recognition of acoustic ID of ancient coins based on deep learning of artificial intelligence for audio signals," *Heritage Science,* vol. 11, no. 1, p. 46, 2023.

[18] S. Khazaee, M. S. Rad, and C. Y. Suen, "Detection of counterfeit coins based on 3D height-map image analysis," *Expert Systems with Applications,* vol. 174, p. 114801, 2021.

[19] I. Bavandsavadkouhi, S. Khazaee, and C. Y. Suen, "An Autoencoding Method for Detecting Counterfeit Coins," in *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, 2022: Springer, pp. 292-301.

[20] B. Liu, X. Yu, A. Yu, P. Zhang, G. Wan, and R. Wang, "Deep few-shot learning for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 57, no. 4, pp. 2290-2304, 2018.

[21] B. Wang and D. Wang, "Plant leaves classification: A few-shot learning method based on siamese network," *Ieee Access,* vol. 7, pp. 151754-151763, 2019.

[22] X. Xia, C. Xu, and B. Nan, "Inception-v3 for flower classification," in *2017 2nd international conference on image, vision and computing (ICIVC)*, 2017: IEEE, pp. 783-787.

[23] Z. Guo, O. Arandjelović, D. Reid, and Y. Lei, "A Siamese Transformer Network for Zero-Shot Ancient Coin Classification," *Journal of Imaging,* vol. 9, no. 6, p. 107, 2023.

[24] L. J. Van Der Maaten and P. Poon, "Coin-o-matic: A fast system for reliable coin classification," in *Proc. of the Muscle CIS Coin Competition Workshop, Berlin, Germany*, 2006, pp. 7-18.

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

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

[27] J. C. Bezdek, *Pattern recognition with fuzzy objective function algorithms*. Springer Science & Business Media, 2013.

[28] S. Khazaee, M. S. Rad, and C. Y. Suen, "Decomposing relief maps to detect counterfeit coins using a hybrid deep learning method," *Available at SSRN 3990636,* 2022.

[29] J. Bucki. www.thesprucecrafts.com (accessed.

[30] S. Gakhar, "Local Image Patterns for Counterfeit Coin Detection and Automatic Coin Grading," Concordia University, 2020.

[31] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a" siamese" time delay neural network," *Advances in neural information processing systems,* vol. 6, 1993.

[32] A. A. Soofi and A. Awan, "Classification techniques in machine learning: applications and issues," *J. Basic Appl. Sci,* vol. 13, no. 1, pp. 459-465, 2017.

[33] O. Russakovsky *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision,* vol. 115, pp. 211-252, 2015.

[34] Ò. Lorente, I. Riera, and A. Rana, "Image classification with classic and deep learning techniques," *arXiv preprint arXiv:2105.04895,* 2021.

[35] D. Argüeso *et al.*, "Few-Shot Learning approach for plant disease classification using images taken in the field," *Computers and Electronics in Agriculture,* vol. 175, p. 105542, 2020.

[36] S. Zambanini and M. Kampel, "Robust automatic segmentation of ancient coins," in *International Conference on Computer Vision Theory and Applications*, 2009, vol. 1: SCITEPRESS, pp. 273-276.

[37] G. L. Team. "Introduction to Image Pre-processing | What is Image Pre-processing." https://www.mygreatlearning.com/blog/introduction-to-image-pre-processing/ (accessed.

[38] M. Reisert, O. Ronneberger, and H. Burkhardt, "A fast and reliable coin recognition system," in *Joint Pattern Recognition Symposium*, 2007: Springer, pp. 415-424.

[39] M. A. U. Khan *et al.*, "A comprehensive survey of depth completion approaches," *Sensors,* vol. 22, no. 18, p. 6969, 2022.

[40] 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 Recognition Letters,* vol. 127, pp. 165-173, 2019.

[41] H. Wang, Q. Wang, F. Yang, W. Zhang, and W. Zuo, "Data augmentation for object detection via progressive and selective instance-switching," *arXiv preprint arXiv:1906.00358,* 2019.

[42] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM,* vol. 60, no. 6, pp. 84-90, 2017.

[43] O. Vinyals, C. Blundell, T. Lillicrap, and D. Wierstra, "Matching networks for one shot learning," *Advances in neural information processing systems,* vol. 29, 2016.

[44] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *ICML deep learning workshop*, 2015, vol. 2, no. 1: Lille.

[45] L. Chen, S. Li, Q. Bai, J. Yang, S. Jiang, and Y. Miao, "Review of image classification algorithms based on convolutional neural networks," *Remote Sensing,* vol. 13, no. 22, p. 4712, 2021.

[46] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, 2009: Ieee, pp. 248-255.

[47] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, 2006, vol. 2: IEEE, pp. 1735-1742.

[48] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning,* vol. 20, pp. 273-297, 1995.

[49] A. Medela and A. Picon, "Constellation loss: Improving the efficiency of deep metric learning loss functions for the optimal embedding of histopathological images," *Journal of Pathology Informatics,* vol. 11, no. 1, p. 38, 2020.

[50] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *the Journal of machine Learning research,* vol. 12, pp. 2825-2830, 2011.

[51] K. D. Joshi, D. Shah, V. Shah, N. Gandhi, S. J. Shah, and S. B. Shah, "Machine vision using cellphone camera: A comparison of deep networks for classifying three challenging denominations of indian coins," in *2022 28th International Conference on Mechatronics and Machine Vision in Practice (M2VIP)*, 2022: IEEE, pp. 1-6.