National College of Ireland

Euro Coin Recognition using YOLOv7

MSc Research Project
Msc Data Analytics

Syed Ahmed Omer
Student ID:22132295

School of Computing
National College of Ireland

Supervisor: Abdul Shahid

National College of Ireland
Project Submission Sheet
School of Computing

National College of Ireland

Student Name:	Syed Ahmed Omer
Student ID:	22132295
Programme:	Msc Data Analytics
Year:	2023
Module:	MSc Research Project
Supervisor:	Abdul Shahid
Submission Due Date:	14/12/2023
Project Title:	Euro Coin Recognition using YOLOv7
Word Count:	6284
Page Count:	22

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Syed Ahmed Omer
Date:	31st January 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Euro Coin Recognition using YOLOv7

Syed Ahmed Omer
22132295

Abstract

Coins have long been a common medium of exchange for financial, charitable, and retail transactions. Even in the present day, when the majority of transactions take place online, coins remain a vital and trustworthy form of payment. Coins come in different denominations, and counting them is a laborious, error-prone process that takes a lot of time. Conventional coin-counting systems arrange and tally the coins according to their physical attributes, such as weight, size, and form, fake coins with similar physical features can easily fool the conventional system. In this paper, we propose an image-based counting system using the state-of-the-art deep learning object detection model YOLOv7, which is capable of simultaneously detecting multiple coins in an image, 99% of recall and precision was achieved by the YOLOv7 model. It has applications ranging from financial, retail, banking institutions, and charity trusts for coin recognition and counting.

Keywords: coin recognition, YOLOv7, deep learning, object detection, coin counting

1 Introduction

Since ancient times, coins have been utilized as tools for the exchange of goods and services. In the era of digitalization, coins continue to be significant and useful low-denomination units of cash. Coins are a dependable medium of exchange that is utilized in daily transactions. For instance, they are widely used in parking garages, vending machines, public transportation, parks, charitable events, and retail establishments.

Because coins come in a variety of modest denominations, such as 5, 10, and 50 cents, counting them is a time-consuming, labor-intensive process that is always prone to human error. In supermarkets and retail establishments, consumers frequently arrive with a large amount of coins. Accurately counting a large number of coins can be difficult and time-consuming, especially when there is a large line of people waiting. The cashier is under pressure due to a long queue, which frequently causes mistakes or miscalculations.

There are a variety of coin counting systems on the market that can be used to tackle these issues; the most widely used ones fall into the following categories:

1.1 Mechanical coin counter:

One of the first coin-counting systems ever introduced, it sorts and counts coins according to their physical attributes such as size, shape, and weight. Compared to electronic machines, it is an inexpensive, basic machine that is easy to use and mostly

appropriate for small amounts of coins. It has mechanical parts and manual movement. It cannot, however, distinguish between fake coins that have the same size, weight, and shape, nor is it appropriate for huge quantities of coins.

1.2 Electronic counting machines:

Compared to a mechanical counter, this machine's electric parts, sensors, and motors can process a higher volume of coins with more accuracy. However, these devices have a high initial cost of investment because they are costly and require frequent calibration to retain precision.

1.3 Weight counting machine:

These use the weight of every coin to determine the quantity and value of the coins. They are palatable and compact. However, this type of machine can only weigh one currency denomination at a time. currency denominations must be manually separated before being weighed on the machine; it also requires human calibration and is unable to distinguish between fake coins of the same weight. Since these techniques heavily depend on the physical characteristics and dimensions of the coin, its size and weight have a significant impact on accuracy and precision, especially when working with worn-out or damaged coins that may have significantly altered weight or dimensions.

In contrast, computer vision counting systems examine the color, texture, size, form, and symbols of the coins to precisely identify the real ones as well as distinguish false ones. Because they only need an app on a phone or other electronic device to function and don't require any special training, they are more precise accurate, and portable.

1.4 Computer Vision:

A field of AI called Computer Vision deals with the extraction of information from pictures and videos. Computer vision aims to imitate human vision in machines so that they can identify patterns and objects from the input images.

One of the applications of computer vision is object detection, which locates and recognizes objects or things in pictures or videos. The object detection models, which are deep learning-based, can be trained on images of euro coins, and then this model can be used for recognizing the coins in an input image, we can then analyze to what extent does application of this deep learning-based model increases the accuracy and tackle the problem of coin detection.

There are several state-of-the-art object detection models which include YOLO, SSD, Faster R-CNN, Mask R-CNN, and Efficient-DetTan et al. (2020). YOLO (You Only Live Once) Object detection model was selected over others for this research on coin recognition because of its inference speed and also because it detects all the classes together in the image, unlike other approaches where a lot of pre-processing steps are required to detect multiple objects, which makes YOLO more efficient than other models for our use case.

YOLOv7, being the state-of-art for object detection and the latest version by the official authors of YOLO architecture was selected for the research. To facilitate the training process data was divided into train, test, and validation splits. Additionally, training data size was increased using data augmentation techniques to increase the performance of the YOLOv7 model.

YOLOv7 shows very good results in detecting multiple coins in varying lighting conditions, precision of 99% and recall of 99% was obtained which surpasses the results of YOLOv5 by 1%.

The format of this document is as follows: Section II summarizes the related work on coin detection and recognition approaches. Section III describes the proposed research methodology. Section IV summarizes the design specifications of YOLOv7, Section V describes the implementation and configurations, Section VI shows the Evaluations and VII concludes with the conclusion and future work.

2 Research Question:

We also aim to explore how well the YOLOv7 model performs in handling images with multiple coin types in different lighting conditions, and various backgrounds and compare the performance with YOLOv5, which is found to be the best-performing model in related studies that were done for the research.

3 Related Work

The problems related to counting systems like using fake coins with the same dimensions and weights were recognized shortly after the advent of counting machines, to address this issue, several different research approaches were proposed. In the following sub sections, we discuss the different categories of research methodologies tackling these challenges.

3.1 Classic Statistical Approaches:

Several statistical methods were used to extract features from the image that can be further used to correctly classify the coins. Linlin Shen proposed a statistical method for coin classification, the author specifies that current classification is based on physical parameters like dimension, which can be bypassed by fake coins of the same dimensions and weight so suggests an image-based coin classification method.

The author uses the Hough transformation on the image to separate the coin from the background and then to achieve rotational invariance by dividing the coin into concentric circles. Gabor wavelet coefficient is applied, and the statistics obtained from each ring are concatenated to form a feature vector, now the Euclidean distance and nearest neighbour are used to match it with other coins. The accuracy obtained was 74.27, which outperforms other edge-based approaches.

This paper concludes by suggesting additional research to improve the accuracy and combining this method with physical parameters like dimensions and weight Shen et al. (2009).

P. Thumwarin proposed a robust coin recognition method with rotation invariance, the author specifies there are many coins with the same size and shape or similar patterns but differ in values, particularly in amulets and coins of Thailand, to simplify the automatic coin recognition system.

The author proposed a statistical method of using Fourier coefficients, the coin can be recognized by the distance between the Fourier coefficient value of the reference coin and the coin to be recognized. This paper however does not show any comparative evaluation of the methodThumwarin et al. (2006).

Anupa Kavale created a prototype for automatic coin sorting and counting using an Arduino board. The system consists of a coin acceptor to identify the coin's denomination, a DC motor to distribute the matching coin, and a display panel to indicate the total. The system has a blower to blow the appropriate coins into their container, a load cell to weigh the coins, and an infrared sensor to count the coins.

This research effectively implemented the hardware-based system to accurately perform the coin counting mechanism and it also proposed an image-based counting system for future research to increase accuracy even more Kavale et al. (2019).

3.2 Traditional Machine Learning Approach:

Machine learning has advanced incredibly in the last few years and several research have been conducted as a result of today's increased computational efficiency. The following studies highlight key contributions in the field of coin detection using traditional machine learning models and techniques.

Nur Nadirah Zainuddin proposed a machine learning approach that involves the images going through several preprocessing stages to improve their quality and prepare them for further tasks like edge recognition and coin segmentation. The stages include converting to grayscale, filtering, eliminating specks, filling in gaps, blending borders, and ultimately segmenting the coin area. These modifications are essential for accurate and efficient deep learning-based coin counting and detection.

Before training the data, two methods were used to minimize the image's dimensions and extract the most important attributes from the images of coins. The first method retrieved 64 features from each image using the grey-level co-occurrence matrix (GLCM). Histogram-oriented gradients (HOG) is the second feature extraction approach which can extract up to 324 features, which is more than the first method. Next, these two feature extraction techniques were used to train six different machine learning models, and the outcomes of all the models were compared. This study concludes that the AdaBoost classifier has the highest accuracy for GLCM feature extraction, while the KNN classifier has the lowest accuracy. The best accuracy for GLCM feature extraction is achieved with the AdaBoost classifier and HOG feature extraction with the ANN classifier

This study helps future researchers and developers in the field of coin recognition by proposing benefits and drawbacks of various approaches Zainuddin et al. (2021).

3.3 Deep Learning Based Approach:

Conventional machine learning algorithms only accept text as input, like words or tabular data. However, images contain several important details that can be used to address various issues in the data industry. Convolutional Neural Network (CNN). is a deep learning-based algorithm that extracts patterns from pictures and videos, following are some deep learning-based approaches for coin recognition tasks.

Shatrughan Modi and Dr. Seema Bawa proposed a deep learning-based approach that uses an artificial neural network (ANN) to recognize Indian coins of different values while maintaining rotation invariance, to ensure that the model can identify the coin regardless of its side, images of both sides of the coin are taken.

To prepare the image for feature extraction and classification using a neural network, several preprocessing processes were carried out. These included transforming the RGB images into greyscale for computational efficiency, the coin's shadows are eliminated from

the greyscale image using the Hough Transform, and then the image is cropped so that just the coin is visible, making it into a square with 100x100 pixel size. By dividing the image into 5x5 pixel segments, the image is further reduced to 20x20 pixels, which further reduces calculation and complexity. The average of each segment is then calculated to provide a pattern average image. These are then transformed into a 400x1 vector and supplied as input to the artificial neural network. This research exhibits the usefulness of the neural network and its application in the coin identification domain. The ANN gets remarkable results with a staggering 97.74 percent correct recognition.

However, the paper does not compare this object detection technique to any other object detection methods and it has a lot of pre-processing which makes it computationally expensive and slow Modi and Bawa (2013).

Nikolay Fonov and Urkaeva Ksenia proposed a study on coin identification and classification, with a focus on USSR currencies. The authors hope to create a classification model that can accurately predict the coins by utilizing ML-CPNN (Multi-Layer Counter Neural Network) Data, which includes 4000 photos divided into 26 categories. The study contrasts two distinct pre-processing steps: in the first, the algorithm is applied to the image which results in the image to transformed into binary form, and the other method is where the image is first converted to grayscale and then the algorithm is applied to the image. The second method smoothed the image and highlighted the corners by applying a Laplacian Gaussian to it. This method was appropriate because it works in all lighting conditions. The first method produced good results in well-lit conditions, but when images with low light were passed, the images had noise. The images were passed to the ML-CPNN network, a neural network with one input layer, one hidden layer, and one output layer

The test dataset yielded a 98 percent accuracy rate, and the research indicates that this can be applied to a phone app that can be used for coin recognition as well.

Although this study advances the use of deep learning for coin recognition and application in mobile applications, it has complex pre-processing steps Fonov and Ksenia (2021).

Alexandra Fanca suggested a method for coin recognition using TensorFlow and Keras, which is based on a Custom CNN architecture. The image is first scaled to contain fewer pixels, which increases the coin's edge detection accuracy. To make the image noise-free Gaussian smoothing/blur is applied to the images, then a "Canny edge detector" is used to identify the coin's edges. A minimum threshold of 30 and a maximum threshold of 150 gradient magnitude were chosen to remove the noise that is not the coin's edges. Then each coin is cropped and saved as a file which is given as input to the CNN architecture which classifies the coin. The CNN cannot detect multiple coins at once so each coin in the image is cropped and then sent to CNN for detection which increases the complexity of preprocessing and this research is limited to identifying only 10 and 50 beans.

The research not only performs coin identification but also developed a method for total value calculation in the image, the research provided a noteworthy contribution to the coin recognition topic Fanca et al. (2022).

Bulbul Bamne proposed a CNN (Convolutional Neural Network) strategy based on deep learning, using a variety of CNN models, including AlexNet, VGG-16, VGG-19, Google Net, and ResNet-50. The results were obtained using a majority voting strategy, which takes the output from all the models and predicts the value of the majority winning class. This approach builds an ensemble model by combining the power of numerous models. The results of the individual CNN model and the ensembled model are compared

in this research.

The CUB 200-2011 dataset, which contains bird photos, was utilized. The ensemble model yielded an accuracy of 97.43 percent, surpassing that of the individual CNN models. It is possible to apply the suggested technique to more real-world object identification scenarios and get higher accuracy. However, using several different models might increase computation and complexity while slowing down the model's inference time Bamne et al. (2020).

A.U. Tajane presents a deep-learning approach for automatic coin recognition and identification. The proposed solution uses a Convolutional Neural Network (CNN) based Alex Net, which was trained to separate Indian coins into four categories: 1, 2, 5, and 10 rupees.

The CNN was trained on a dataset with 1600 coin images, The results show that the deep learning-based approach performs better than conventional coin identification systems, offering faster response times and more accuracyTajane et al. (2018).

Yufeng Xiang and Wei Qi Yan researched the use of LSTM (Long Short-Term Memory) and CNN (Convolutional Neural Network) to identify fast-moving coins using slow-motion recording on a mobile phone.

CNN is used to extract information from images, while LSTM is utilized for sequential data such as words. In the paper, LSTM and CNN are combined. The fast-moving coin is captured using the phone's slow camera, and each frame is then passed to CNN, whose output is subsequently utilized as an input for LSTM.

The proposed model in the paper produced an accuracy of 90%, surpassing the 80% accuracy achieved by the Faster R-CNN model. This work demonstrates the effectiveness of the suggested method in precisely identifying coins from video clips, significantly advancing the detection of fast-moving coinsXiang and Yan (2021).

Multiple pre-processing steps and inference speed presented challenges for these techniques, which were overcome by following works using YOLO architecture. A real-time object detection architecture called You Only Live Once (YOLO) can identify multiple objects in an image or video at the same time.

Danielle M. Dumaliang proposed utilizing the CNN-based YOLO v3 architecture to identify and recognize coins. It attempts to address the issue that travelers encounter while trying to distinguish between different coin values in various nations. The model was trained using four different currencies. This article also aimed to determine the ideal distance for taking photos, and it was found that six centimeters was the ideal distance. The coin's angular rotation test results showed the highest 98.1% accuracy was achieved at 0° and 90° angles. A graphical user interface (GUI) was created to accurately translate the coin into its equivalent in other currencies.

This research contributes to the field by investigating not only the model building but also the rotation angle and distance for maximum accuracy Dumaliang et al. (2021).

S. Prabu discusses the identification and detection of Indian coins in different values using physical characteristics such as size, shape, and pattern in the images and compares the outcome with other state-of-the-art object detection models in the research.

YOLO V5 (You Only Live Once) architecture based on Convolution Neural Network (CNN) was utilized to train the model. The dataset consisted of 656 images representing different coin values, such as 1, 2, 5, and 10 rupees. The outcomes were compared against the results of other widely used object detection algorithms, such as Fast R-CNN. SSD and R-CNN faster. With an impressive 98% F1-Score, YOLO V5 surpasses other object detection methods for the dataset of Indian coins.

This work demonstrates the effective use of deep learning-based YOLO for coin recognition along with the evaluation and comparison of alternative methods. It highlights the possibility of coin identification in real-world situations and opens new opportunities for further research and improvements in the field Prabu et al. (2022).

Table 1: Related Work Comparison

S.No	Paper	Technique	Metrics	Year	Comment
1	Statistics of Gabor Features for Coin Recognition	Statistics of Gabor feature	74.27%	2009	Low Accuracy, Additional Research required
2	"A Robust Coin Recognition Method with Rotation Invariance"	Fourier coefficient	N/A	2006	Classic Stats Approach, No comparative evaluation
3	Coin Counting And Sorting Machine	ARDUINO	N/A	2019	Hardware approach, Image-based counting system for future work to improve further accuracy.
4	Malaysian Coins Recognition Using Machine Learning Methods	Machine Learning Models	98%	2021	Compares different machine learning models. Deep learning models are more appropriate for the task
5	Automated Coin Recognition System using ANN	ANN	97.74%	2011	Computationally intensive preprocessing
6	Development of an Accurate Neural Network for Coin Recognition	Counter Neural Network	98%	2021	Deep Learning-based coin app, complex preprocessing

Table 1 (continued): Coin Recognition Papers and Techniques

S.No	Paper	Technique	Metrics	Year	Comment
7	Romanian Coins Recognition and Sum Counting System From Image Using TensorFlow and Keras	Custom CNN using Keras and TensorFlow	94%	2022	Lacks Multi-Coin recognition
8	Transfer learning-based Object Detection by using Convolutional Neural Networks	CNN	97%	2020	Using Multiple Models impacts Inference Speed
9	"Deep Learning Based Indian Currency Coin Recognition"	CNN	N/A	2018	Deep Learning-based, outperforms traditional methods
10	Fast-moving coin recognition using deep learning	LSTM & CNN	90%	2021	Significant contribution in fast coin detection from video sequences but limited to just identifying the moving coin
11	Coin Identification and Conversion System using Image Processing	YOLO v3 and CNN	98.1% for 0° and 90° angles	2021	Focused not only on Accuracy but distance from camera and rotation analysis
12	Indian Coin Detection and Recognition Using Deep Learning Algorithm	YOLO V5	98%	2022	YOLOv5 implementation but latest state-of-the-art version available

Table 1, shows the comprehensive comparison of various studies in the related works section, where the columns labeled "Paper" denote the title of the paper, "Techniques" denote the approach or technique used in that specific research, "Metrics" indicates the

evaluation metrics obtained from the approach, "Year" is the year of publication and "Comments" indicates all the advantages or limitations, the limitations stated were addressed in this research.

Notably, we identified research gaps related to coin detection and recognition, that motivated us to investigate the effectiveness of YOLOv7. we also aim to explore how well YOLOv7 performs in handling variations in multiple coin types, different lighting conditions, and various backgrounds.

4 Methodology

The proposed approach follows a systematic flow, Figure 1 shows the architectural flow of the proposed approach. The input images are prepared for model training by applying preprocessing techniques which include data augmentation and splitting then the training and validation data are sent for model training where the model is trained based on the parameters provided and then we use the trained model to get the predictions of the test data which is compared with ground truth and the model performance is calculated.

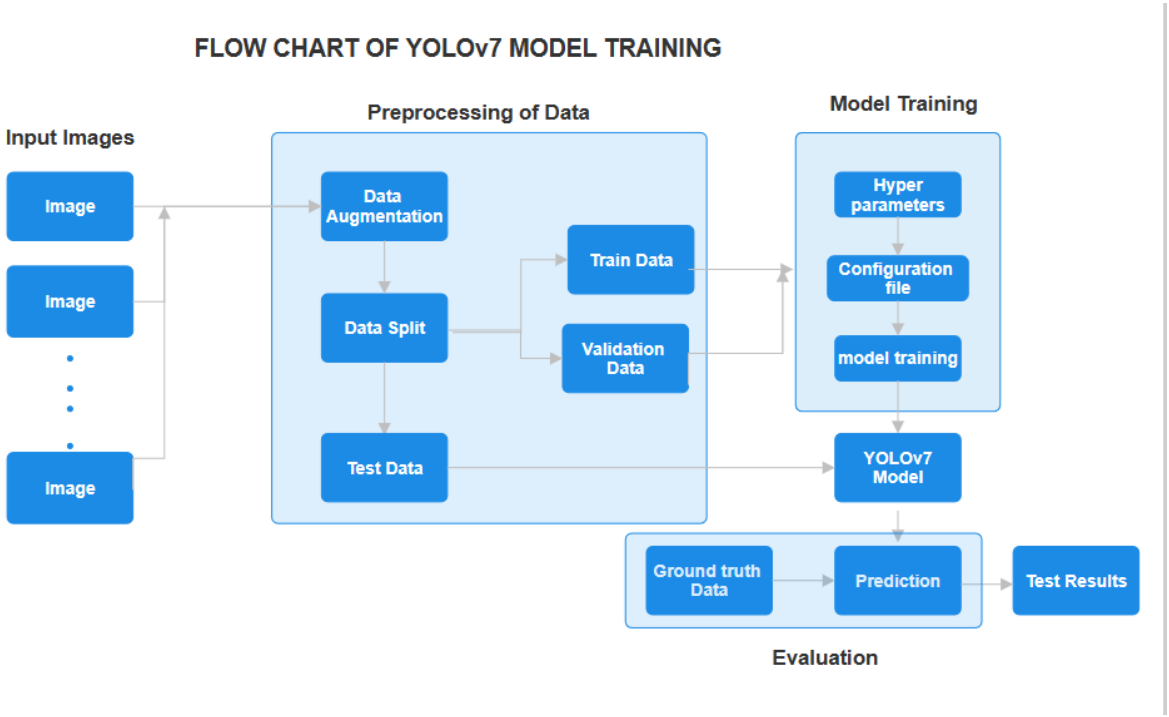


Figure 1: Architecture diagram of the proposed approach

Detailed steps and insights of the architectural diagram of the proposed system are discussed in the next sections.

4.1 3.1 Dataset:

The Dataset used was obtained from the open-source Githib repository, Images of the 1, 2, 5, 10, 20, 50 cent, and 1, 2 euro coins in different folders are included in this dataset. Bounding boxes are accurately labeled around the coins in every image in the dataset. In this dataset each image may have multiple classes of coins, for example, as seen in the

table 2 the 5 cents class has a total of 71 images in the dataset, but each image may have a single or multiple 5-cent coins in it. The mixed coin is a dataset that has images where each image has a different denomination of multiple coins or classes present in it. Figure 2b shows an example of multiple same coin images and figure 2a shows multiple different classes of coins.

Coin	Number of Images
5 cent	71
10 cent	73
20 cent	44
50 cent	44
1 euro	43
2 euro	47
Mixed coins	55

Table 2: Number of Images for Each Coin

(a) Image with mixed coins

(b) Image with multiple same coins

Figure 2: Example Images in the Dataset

This Dataset can be found on the follwoing Github repository, https://github.com/SuperDiodo/euro-coin-dataset

4.2 Preprocessing:

Pre-processing is an important part of model building; it gives insights into the data and makes sure that the data is fully prepared for the model-building step. Below are the steps that were used in the pre-processing of the data.

4.2.1 Data split:

The Dataset was uploaded to the Roboflow and then and then the images were randomly classified into train, test, and valid datasets. It was made sure that the training, testing, and validation datasets have the images of all the classes. The dataset was split into the

train, test, and valid with below proportions. 70 % - Train dataset 20 % - test dataset 10 % - test dataset

The 70-20-10 split is widely accepted and is a standard split used for training the model, which is also a well-balanced split that gives a maximum percentage for training data and then test and valid data. This split is not used in certain cases where the data is very small, but in our case, data is sufficient to train the model so, we stick with the standard split.

4.2.2 Data Augmentation:

Data Augmentation is a technique that is generally used for image datasets. It artificially increases the size of the training data by applying various augmentation techniques on the images like rotation, brightness change, and others, which helps the model to be more generalized and avoids over-fitting, it also makes the model more robust by creating possible real-time scenarios, table 3 shows all the augmentation techniques and their corresponding ranges which were used in the experiment, following are the detailed explanation of each technique used.

1. Flip: This Flip the image horizontally and vertically making the model robust to different arrangement of the coins in the image.

2. 90-degree rotation: It rotates the image clockwise and counterclockwise which again creates a different arrangement of the coin in the images,

3. Rotation: It rotates the image in the given range, for this experiment the range given is +15 to -15 degrees. coins appear in a different orientation in the real world, this makes the model more robust against any coin orientation in the images.

4. Saturation: Saturation increases or decreases the intensity of the color in an image, for this experiment the range of change in saturation is set to -25 and +25 percentage. This makes the model more robust to different lighting conditions in real-time.

5. Brightness: This randomly increases or decreases the brightness in an image within the set range of +25 to -25 percent. This also helps the model to perform better in different lighting conditions.

6. Noise: This randomly adds noise to the images which makes the model more robust to the noise that may be present in a real-time world.

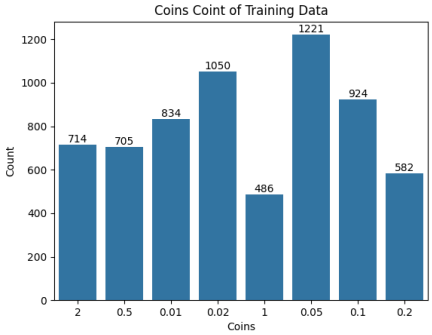
7. Bounding box brightness: This increases or decreases the brightness randomly in the bounding box areas only, which helps prevent overfitting and performs better in distorted images as well.

4.2.3 Data Distribution:

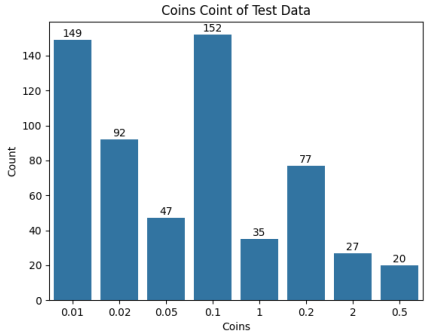
Data Distribution is the spread of data or the number of images present in the dataset for each class. Since the dataset has images that have multiple classes of coins, all the classes in every image were counted and plotted on the graphs for train, test, and validation datasets, Figure 3 shows the distribution of data where the x-axis represents the class and the y-axis represents the count of class in the dataset.

S.no	Augmentation	Range
1	Flip	Horizontal, Vertical
2	90 degree Rotate	Clockwise, Counter-Clockwise, Upside Down
3	Rotation	Between -15 and +15 Degree
4	Saturation	Between -25% and +25%
5	Brightness	Between -25% and +25%
6	Noise	Upto 3% of pixels
7	Bounding Boxes - Brightness	Between -25% and +25%

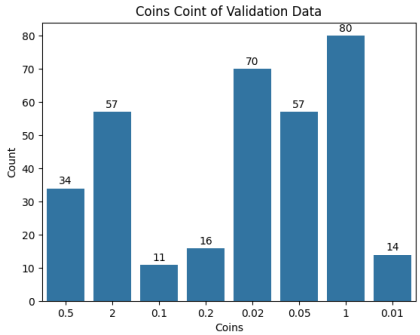
Table 3: Data Augmentation Techniques



(a) train data distribution



(b) test data distribution



(c) valid data distribution

Figure 3: Data Distribution

5 Design Specification

YOLO is an object identification algorithm that detects objects in photos and videos in real time. It completes the detection process in a single forward pass, which makes it incredibly quick and efficient. In contrast to other object recognition models, YOLO has the ability to identify multiple objects in the image at once, reducing the need for pre-processing processes like cropping each object separately and then feeding the model.

5.1 YOLO (You Only Live Once) V7:

Object Detection fields advanced by the release of the YOLOv7 object detection model, it is the latest YOLO version created by the official authors of YOLO architecture. YOLOv7 is more accurate and faster than its previous version of YOLO which is YOLOv5. YOLO models are single-stage object detector models. In YOLO models the image frames or pixels are featured in the backbone which is then passed to the neck where it is combined and mixed and passed along the head of the network then YOLO predicts the location of the object along with the class in an image.

The YOLOv7Wang et al. (2023) authors thought to create a model that set the state of the art in object detection by creating an architecture that predicts the bounding boxes more accurately with the same inference time as its other peers. To achieve that, the authors made several changes to the YOLO network and training routine, below are some notable changes that were made in the YOLOv7.

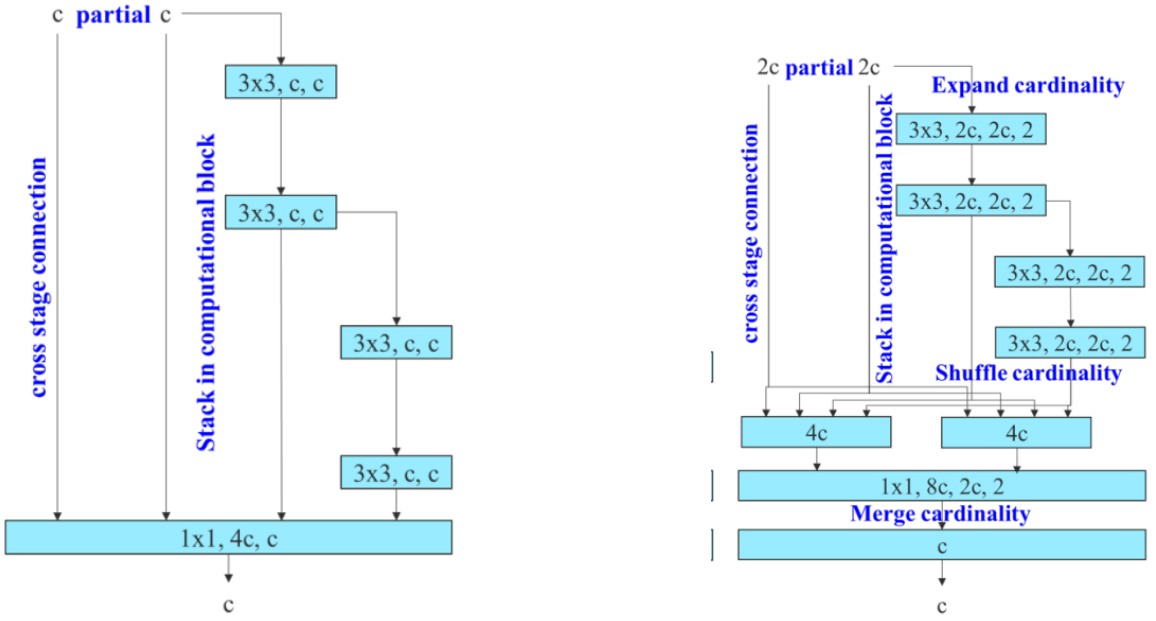
5.1.1 Extended Efficient Layer Aggregation (E-ELAN):

The efficiency of the convolutional layers in the backbone of the YOLO model is very important in achieving a higher inference speed. The author built on the research happening on this topic, E-ELAN (Extended Efficient Layer Aggregation) was built on top of the architecture of ELANWang et al. (n.d.). In ELAN stacking more computational blocks leads to a decrease in parameter utilization, the new Extended ELAN proposed in the paper addressed this instability issue. This approach not only maintained the efficiency of ELAN but also made it possible to add more computational blocks to learn more features, figure 4a shows the architecture of ELAN, and figure 4b shows the architecture of E-ELAN.

5.1.2 Model Scaling Techniques:

Since different applications require different levels of accuracy and speed, model scaling plays a crucial role in obtaining models with various scales and various inference speeds by adjusting specific attributes of the model. Generally, object detection models consider the network's breadth, depth, and training resolution.

In this paper, the author proposed the compound model scaling method shown in figure 5, in which by coordinating width scaling in transmission layers with depth scaling in computing blocks, this technique preserves the original design features while optimizing the structure. By doing this, the model is guaranteed to retain hardware efficiency even when the scale changes.



(a) ELANWang et al. (n.d.)

(b) E-ELAN Wang et al. (2023)

Figure 4: Extended Efficient Layer Aggregation

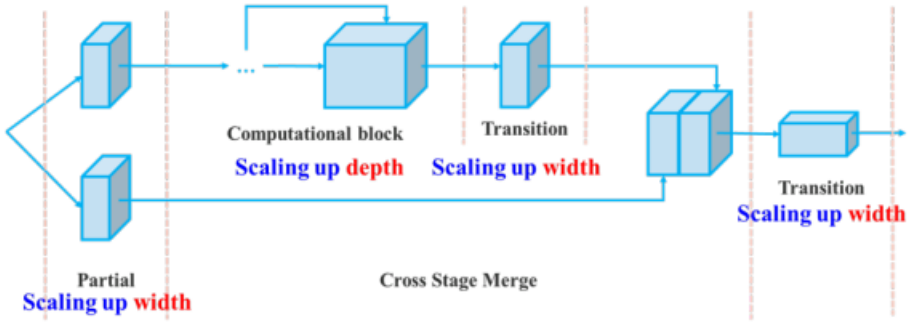


Figure 5: Compound scaling up depth and width for concatenation-based model Wang et al. (2023)

5.1.3 Planned re-parameterized convolution:

In re-parameterization procedures, a set of model weights are averaged to produce a model that is more robust to the general patterns it is trying to represent, the YOLOv7 authors use gradient flow propagation path to analyze how re-parameterized convolution should be combined with different networks. The authors introduced "planned re-parameterized convolution" by using RepConvDing et al. (2021) without identity connection to design the architecture.

5.1.4 Auxiliary Head Coarse-to-Fine:

In YOLOv7 authors introduced an approach called "Coarse for auxiliary and fine for lead loss" in the context of deep supervision while training the deep networks. Since the final predictions are made by the head, who is far away in the network, it can be advantageous to add an auxiliary head somewhere in the middle of the network, by doing this when training you are not only supervising the final head but also the auxiliary head. The auxiliary head does not train as efficiently as the final head, so the author experimented with different supervisions settling down to Coarse to fine lead guided assigner where supervision is passed from the final head at different granularity. figure 6 shows different auxiliary head supervisions.

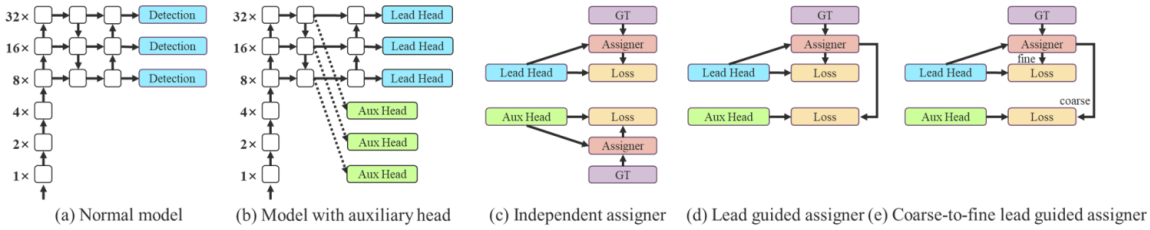


Figure 6: Wang et al. (2023)

The YOLO-based architecture was selected for this research because this task requires speed and the ability to detect multiple coins in the image at the same time. Many versions of the YOLO design are available, since YOLO v7 is the most recent state-of-the-art release from the official authors and performs better than other models, it was selected for the research.

6 Implementation

The Model was trained in Google Colab, which gives access to free GPU resources which is very important as the deep learning model takes huge time to run on CPU's. Below are the configurations of the machine and GPU used on Colab.

Machine Configurations:

GPU Model: Tesla T4 Processor: Intel(R) Xeon(R) CPU @ 2.20GHz CPU Cores: 2

Software Environment:

Operating System: Linux 90fbe043f659 5.15.120+ Python Version: 3.10.12

6.1 Hyper-parameters:

Hyper-parameters are the configuration settings that are used to train the machine-learning model. Default hyper-parameters were used to train the model, figure-6 shows all the hyper-parameters used in the training of the model.

```
lr0: 0.01  # initial learning rate (SGD=1E-2, Adam=1E-3)
lrf: 0.1  # final OneCycleLR learning rate (lr0 * lrf)
momentum: 0.937  # SGD momentum/Adam beta1
weight_decay: 0.0005  # optimizer weight decay 5e-4
warmup_epochs: 3.0  # warmup epochs (fractions ok)
warmup_momentum: 0.8  # warmup initial momentum
warmup_bias_lr: 0.1  # warmup initial bias lr
box: 0.05  # box loss gain
cls: 0.3  # cls loss gain
cls_pw: 1.0  # cls BCELoss positive_weight
obj: 0.7  # obj loss gain (scale with pixels)
obj_pw: 1.0  # obj BCELoss positive_weight
iou_t: 0.20  # IoU training threshold
anchor_t: 4.0  # anchor-multiple threshold
# anchors: 3  # anchors per output layer (0 to ignore)
fl_gamma: 0.0  # focal loss gamma (efficientDet default gamma=1.5)
hsv_h: 0.015  # image HSV-Hue augmentation (fraction)
hsv_s: 0.7  # image HSV-Saturation augmentation (fraction)
hsv_v: 0.4  # image HSV-Value augmentation (fraction)
degrees: 0.0  # image rotation (+/- deg)
translate: 0.2  # image translation (+/- fraction)
scale: 0.9  # image scale (+/- gain)
shear: 0.0  # image shear (+/- deg)
perspective: 0.0  # image perspective (+/- fraction), range 0-0.001
flipud: 0.0  # image flip up-down (probability)
fliplr: 0.5  # image flip left-right (probability)
mosaic: 1.0  # image mosaic (probability)
mixup: 0.15  # image mixup (probability)
copy_paste: 0.0  # image copy paste (probability)
paste_in: 0.15  # image copy paste (probability), use 0 for faster training
loss_ota: 1 # use ComputeLossOTA, use 0 for faster training
```

Figure 7: Hyper-parameters

Learning rate:learning rate is a hyper-parameter, which decides the size of steps taken during the optimization of a machine learning model, the greater the size of the learning rate larger the steps taken during optimization and vice-versa. The value of 0.01 was the default value used for the research.

weight decay: Weight decay is a regularization technique that prevents overfitting of the model by penalizing the large weights in the model during the training. The default value of 0.0005 was used for the research.

warmup_epochs: Warmup epochs refer to initial epochs during the training phase where a lower learning rate is used for the initial few epochs and then converted into the planned learning rate, this helps the model stabilize and avoid large weight updates. The default value of 3 is used for the research.

Training epochs: Iteration over the entire training dataset is called an epoch during the training of the machine learning model. Choosing the right number of epochs is important because to less epochs result in low learning and low accuracy whereas a very

high number of epochs results in overfitting on training data which means data performs well on training data but is poor on test data. Considering the complexity of the problem to be solved moderate level "100 epochs" were selected.

Batch size: Training data is not trained all at a time in an epoch, the data is divided into batches and then trained, a batch size of 16 was selected for the experiment as GPU memory is limited smaller batch size was selected.

Weights used: Transfer learning was used for the model training so, the weight $yolov7_training.pt$ was downloaded from the repository.

Configuration File: The Default configuration file has 80 classes from COCO dataset, this file was changed to 8 classes with the respective names of each class that is value of coins in cents and the path to the data set was updated. Figure [] shows the configuration file used.

```
1
2   train: D:\NCI\Thesis_code\data_aug_exp\data\train
3   val: D:\NCI\Thesis_code\data_aug_exp\data\valid
4   val: D:\NCI\Thesis_code\data_aug_exp\data\test
5
6
7   nc: 8
8   names: ['1', '10', '100', '2', '20', '200', '5', '50']
```

Figure 8: Configuration file

7 Evaluation

The Investigation in the related studies section revealed that the YOLOv5 object detection model, which is a deep learning-based technique achieved the best accuracy. To do a fair comparison, the YOLOv5 model was trained with the same data, the same preprocessing, split, augmentation, and the same epochs that were used in training the YOLOv7 model.

This ensures a consistent benchmark to evaluate and compare both the model performance on the same dataset and task. Detailed Evaluations and comparisons for both the model YOLOv7 and YOLOv5 are discussed in the next sections.

7.1 YOLOv7 Evaluation

Metrics used:

Since the data was imbalanced, Accuracy was not used in the evaluation, as accuracy can be skewed to the dominant class thus making it a misleading metric to be used in such a scenario. Metrics like precision, recall, and F1 score provide a more balanced evaluation when dealing with imbalanced datasets.

Precision: Precision is defined mathematically as True Positive divided by (True Positive + False Negative)

Precision = TP / (TP + FP)

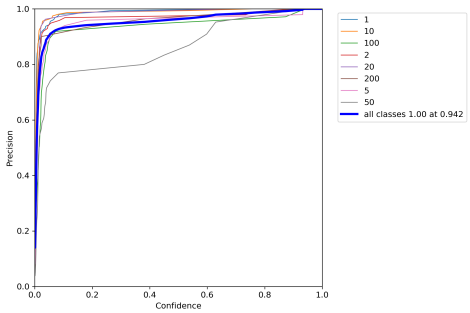
It indicates the number of positive predictions well made, The higher it is, the less the False Positive predicted by the model.

Recall: Recall shows the percentage of correctly predicted positives by the model Mathematically represented as Recall = TP / (TP + FN) High recall indicates the model will not miss any positives.

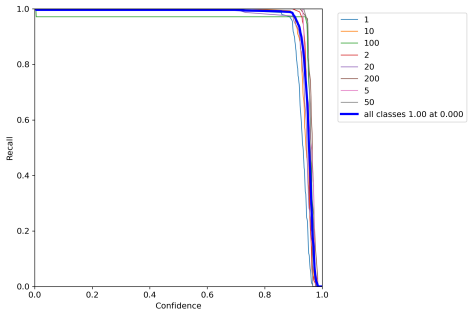
F1 Score: Precision and Recall do not give any information about the quality of all predictions so, they stand alone and cannot be used to evaluate the model. F1 Score is a combination of both these metrics which provides a good evaluation of the performance of the model.

F1 = 2 * Precision * Recall / (Precision + Recall)

Figure 9a and figure 9b show the precision and recall curves, the x-axis indicates the confidence, and the y-axis indicates precision and recall values respectively. Figure 10a and figure 10b indicate the F1 score of the model and the Precision-Recall curve of the model on the test data set using YOLOv7. Figure 11 shows the confusion matrix for the YOLOv7 model.

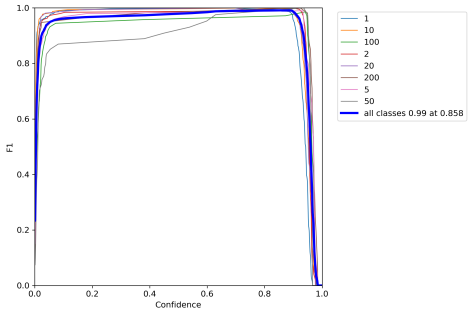


(a) P Curve for Test Data using YOLOv7

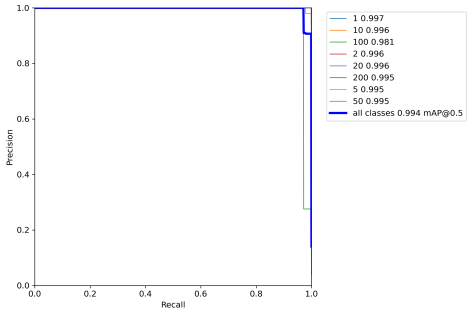


(b) R Curve for Test Data using YOLOv7

Figure 9: Precision, Recall Curves for YOLOv7 on test data set



(a) F1 score for Test Data using YOLOv7



(b) PR Curve for Test Data using YOLOv7

Figure 10: F1 Score, PR Curve for YOLOv7 on Test Data

It is observed that the F1-score for the coin 50 cents is low compared to the rest, the coins 20 cents and 50 cents have a lot of similarities in terms of the same pattern and designs on both coins, which could be the possible reason for the low F1-score.

7.2 Experiment 2, YOLOv5 Evaluation

The YOLOv5 model was trained on the same dataset for 100 epochs with default hyper-parameters and the same configuration file. Following are the results obtained from the

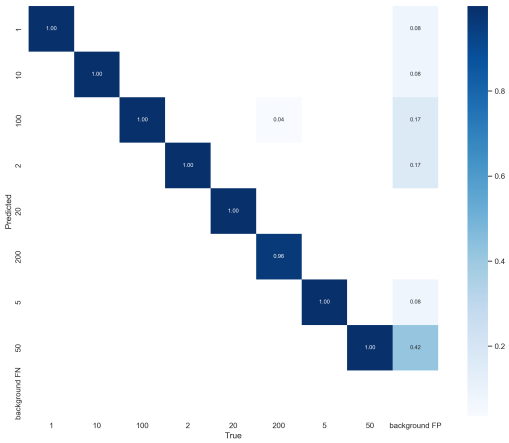
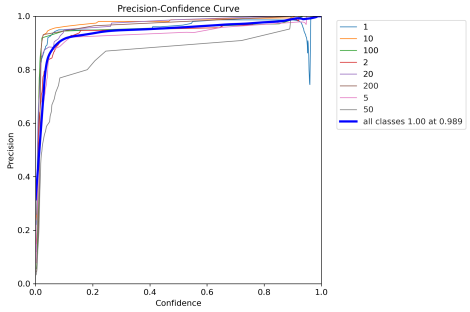
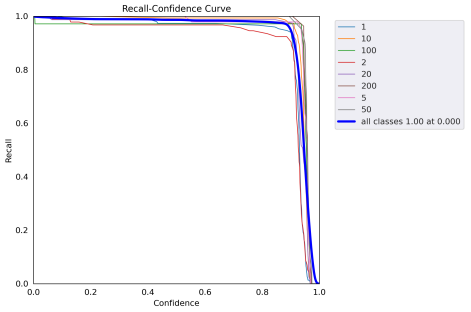


Figure 11: Confusion matrix for Test Data using YOLOv7

model on the test dataset, Figure 12a and 12b show the Precision and Recall Curves, Figure 13a and 13b shows the F1 score and Precision-Recall curves obtained from the test data. The confusion matrix is shown in figure 14



(a) P Curve for Test Data using YOLOv5



(b) R Curve for Test Data using YOLOv5

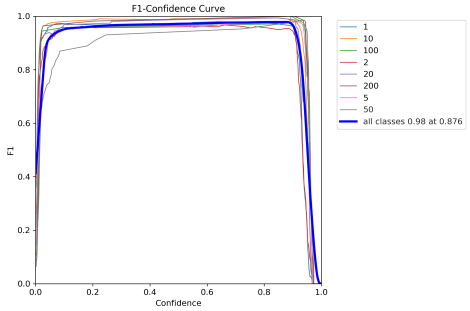
Figure 12: Precision, Recall Curves for YOLOv5 on test data set

7.3 Discussion

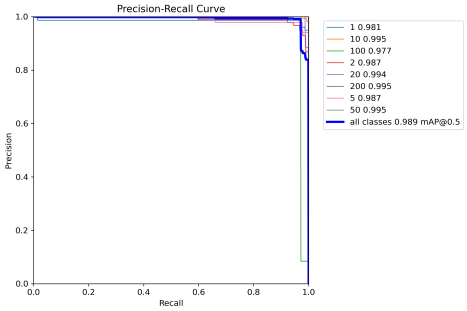
Images with various backgrounds and multiple classes in different lighting conditions were given to both the models, YOLOv7 the current state of the art, and YOLOv5 which got the best metrics for the deep learning-based approaches in the related work section.

The comparison was done on both the models YOLOv7 and VOLOv5 on the same dataset, table 4 shows the comparison between the models. YOLOv7 achieved the best results compared to other approaches in the related work section, it even outperformed YOLOv5 by a margin of 1% in Precision, Recall, and mAP@0.5

The limitations of the model arise from changes in new coin patterns. If the coin pattern for a specific class is absent in the training data, the model may struggle to recognize those particular coins. Coins are to be spread out in the image as the model is also incapable of handling instances where the coins overlap with each other. The performance of the model may also be compromised when dealing with images that at captured at long distances from the coins. To address these limitations, the model should



(a) F1 score for Test Data using YOLOv5



(b) PR Curve for Test Data using YOLOv5

Figure 13: F1 Score, PR Curve for YOLOv7 on Test Data

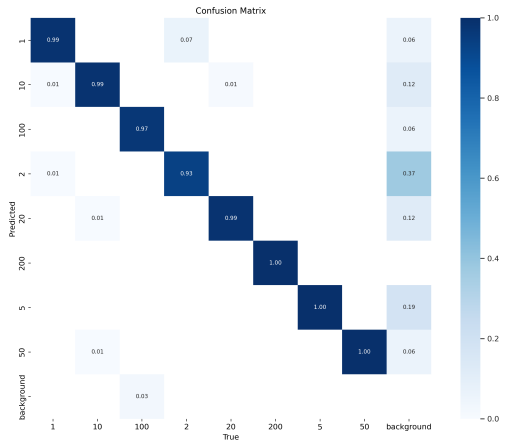


Figure 14: Confusion matrix for Test Data using YOLOv5

be trained on more diverse data containing various patterns of new coins, overlapping scenarios, and images captured from various angles and distances.

The predictions obtained from the model can be used to calculate the total amount of coins present in an image. However, the absence of labeled ground truth data in the current dataset prevents the evaluation of this aspect. Exploring this potential application could serve as valuable future work.

Model	Class	Images	Labels	Precision	Recall	mAP@0.5	mAP@0.5:0.95
YOLOv7	all	105	599	0.991	0.991	0.994	0.891
YOLOv5	all	105	599	0.98	0.976	0.989	0.895

Table 4: Performance Metrics of YOLOv7 and YOLOv5. Green cells indicate better-performing metrics.

8 Conclusion and Future Work

YOLOv7 outperforms other models and achieves the best results for coin recognition tasks compared to other approaches that were discussed in the related work section. We

conclude by saying YOLOv7 is a very effective object detection algorithm that handles variation and multiple coin types in different lighting conditions and backgrounds.

The predictions obtained from the model can be used to calculate the total value of coins in a given input image, This model has the potential to be deployed as a mobile app, making it usable in financial, retail, and banking institutions for efficient coin recognition and counting. Additionally, there is an opportunity to integrate fake coin detection alongside coin recognition, offering a portable and cost-effective solution for coin counting in many industries. This suggests promising avenues for future developments.

As part of future work, extending this implementation to support multiple currencies could also be explored. This would broaden the applicability of the model, allowing it to recognize and calculate the total value of coins across various currency types.

References

Bamne, B., Shrivastava, N., Parashar, L. and Singh, U. (2020). Transfer learning-based object detection by using convolutional neural networks, *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*, IEEE, pp. 328–332.

Ding, X., Zhang, X., Ma, N., Han, J., Ding, G. and Sun, J. (2021). Repvgg: Making vgg-style convnets great again, *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13733–13742.

Dumaliang, D. M., Rigor, J. M. Q., Garcia, R. G., Villaverde, J. F. and Cuñado, J. R. (2021). Coin identification and conversion system using image processing, *2021 IEEE 13th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, IEEE, pp. 1–6.

Fanca, A., Puscasiu, A. P., Gota, D. I., Giurgiu, F. M., Santa, M. M., Valean, H., Domuta, C. and Miclea, L. (2022). Romanian coins recognition and sum counting system from image using tensorflow and keras, *2022 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*, IEEE, pp. 1–7.

Fonov, N. and Ksenia, U. (2021). Development an accurate neural network for coin recognition, *2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus)*, IEEE, pp. 337–341.

Kavale, A., Shukla, S. and Bramhe, P. (2019). Coin counting and sorting machine, *2019 9th International Conference on Emerging Trends in Engineering and Technology - Signal and Information Processing (ICETET-SIP-19)*, pp. 1–4.

Modi, S. and Bawa, D. S. (2013). Automated coin recognition system using ann, *arXiv preprint arXiv:1312.6615* .

Prabu, S., Jawali, N., Sundar, K. J. A., Sharvani, K., Shanmukhanjali, G. and Nirmala, V. (2022). Indian coin detection and recognition using deep learning algorithm, *2022 6th Asian Conference on Artificial Intelligence Technology (ACAIT)*, IEEE, pp. 1–6.

Shen, L., Jia, S., Ji, Z. and Chen, W.-S. (2009). Statictics of gabor features for coin recognition, *2009 IEEE International Workshop on Imaging Systems and Techniques*, IEEE, pp. 295–298.

Tajane, A., Patil, J., Shahane, A., Dhulekar, P., Gandhe, S. and Phade, G. (2018). Deep learning based indian currency coin recognition, *2018 International Conference On Advances in Communication and Computing Technology (ICACCT)*, IEEE, pp. 130–134.

Tan, M., Pang, R. and Le, Q. V. (2020). Efficientdet: Scalable and efficient object detection, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Thumwarin, P., Malila, S., Janthawong, P., Pibulwej, W. and Matsuura, T. (2006). A robust coin recognition method with rotation invariance, *2006 International Conference on Communications, Circuits and Systems*, Vol. 1, IEEE, pp. 520–523.

Wang, C., Liao, H. and Yeh, I. (n.d.). Designing network design strategies through gradient path analysis. arxiv 2022, *arXiv preprint arXiv:2211.04800* .

Wang, C.-Y., Bochkovskiy, A. and Liao, H.-Y. M. (2023). Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7464–7475.

Xiang, Y. and Yan, W. Q. (2021). Fast-moving coin recognition using deep learning, *Multimedia Tools and Applications* **80**: 24111–24120.

Zainuddin, N. N., Azhari, M. S. N. B. N., Hashim, W., Alkahtani, A. A., Mustafa, A. S., Alkawsi, G. and Noman, F. (2021). Malaysian coins recognition using machine learning methods, *2021 2nd International Conference on Artificial Intelligence and Data Sciences (AiDAS)*, IEEE, pp. 1–5.