Classifying New Zealand Coins using YOLO Machine Learning

Cody Wech
Computer Science and Software Engineering
University of Canterbury
Christchurch, New Zealand
cwe104@uclive.ac.nz

Richard Green
Computer Science and Software Engineering
University of Canterbury
Christchurch, New Zealand
richard.green@canterbury.ac.nz

***Abstract*—This paper presents a method for the automated detection and classification of New Zealand coins using the YOLOv8m deep learning model. Related works showed that gathering a large dataset is a common issue and is crucial to a model's performance. This proposed method addresses the challenge of needing a large dataset for robust performance through the use of automated annotation with HSV color segmentation, Hough Circle transform, and Canny edge detection. A dataset of 42,154 annotations was generated and augmented to simulate real-world conditions. The YOLOv8m model achieved a precision of 0.995, recall of 0.995, Mean Average Precision threshold at 0.5 (mAP@50) of 0.994, and a detection accuracy of 98.3% on 50 unseen images (heads side) and 98.5% (tails side). Although the model shows high performance, there are still challenges due to the 20 cent and 50 cent coins being prone to glare, and their similar compositions. There is also a significant decrease in performance when classifying coins on the tails side, as compared to the strong performance on the heads side. The model demonstrates strong potential for real-time applications, with additional work needing focus on enhancing the robustness in various lighting conditions and orientations.**

I. Introduction

Coins are used all over the world as a universal currency. There are many practical uses for automating the process of counting the amount of coins, through banking, vending machines, and retail roles. Automating the counting of coins improves efficiency, precision and reduces the likelihood of human error where accuracy is critical.

This paper proposes a method for detecting and classifying New Zealand coins from an image, summing up the denomination for each detected coin to produce the total amount in New Zealand currency. This proposed method aims to develop an efficient and accurate method to classify New Zealand coins from images that may have various lighting conditions, backgrounds, and overlapping coins.

II. Background

A. New Zealand Coins

New Zealand coins are currently available in 10 cents, 20 cents, 50 cents, 1 dollar, and 2 dollar denominations. New Zealand coins are unique, causing special challenges to be created when classifying them due to their similarity between denominations. These challenges include similar sizes, colours, designs, and a general lack of distinguishing features. For example, the 20 cent and 50 cent coins are both silver, composed of plated steel, and vary by 3mm with diameters of 21.75mm and 24.75mm respectively. Their plated steel composition makes them hard to tell apart, especially as they are prone to glare, causing visual detection to be purely based on diameter when glare occurs. Additionally, neither coin's reverse (tails) side of both coins have any easily identifiable characteristics. Both 1 dollar and 2 dollar coins are composed of aluminium-bronze, and vary by 3.5mm with diameters of 23mm and 26.5mm respectively, again creating challenges for classification on geometric features. However, the 10 cent coin is unique, composed of copper-coated plated steel with a diameter of 20.5mm. [1] (Fig. 1).

Fig. 1. An image of current New Zealand coins [2]

The current 10-cent, 20-cent and 50-cent coins have been in circulation since 31 July 2006, while the 1-dollar and 2-dollar coins have been in circulation since 11 February 1991 [3]. This creates wear and tear from years in circulation, degrading coin features and a clean visual input that deep learning models rely on. Coins may also be scratched, covered in dirt, and unpolished, causing features to be hard to classify, especially with poor lighting.

B. Related Works

Van Der Maaten and Postma [4] proposed an automatic coin classification for old European coins. The proposed method used contour features, texture features, and edge histograms (Fig. 2) to analyze shapes and textures to differentiate coin types.

(a) Original coin image. (b) Edge angle histogram. (c) Edge distance histogram.

Fig. 2. A diagram of the edge histograms used in the paper [4]

Their approach lacked accuracy when the model was presented with worn or damaged coins due to their irregularities and texture variations. The reliance on manual feature calculations limited its ability to classify coins in darker lighting conditions, as not all edge pixels could be found.

Nolle, Penz, Rubix, Mayer, Hollander and Granec [5] proposed Dagobert, a pattern recognition method to identify high volumes of coins. This method relied on methods such as Canny edge detection and the Laplacian operator. Their approach struggled with overlapping coins and cluttered backgrounds, as edge detection could not distinguish individual coins by itself in complex environments.

Kim and Pablovic [6] utilized convolutional neural networks (CNNs) to find characteristic landmarks on Roman coins, focusing on features such as engravings for classification. They proposed training their dataset on ImageNet, a pre-trained model due to CNNs requiring large datasets because there are millions of parameters to be estimated. Their approach achieved robust performance, and out performed previous approaches, however did not achieve high accuracy due to a lack of data to train on. Capece, Erra, and Ciliberto [7] also developed a coin recognition system using a CNN to achieve high classification accuracy. AlexNet was utilised as a pre-trained model, which includes local feature extraction, feature coding, and learning (Fig. 3)

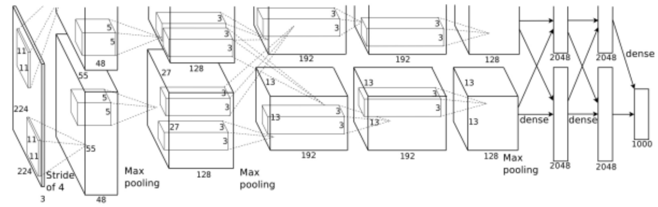

Fig. 3. A diagram of AlexNet model used in the paper [7]

However, this proposed method also faced the challenge of a large dataset of training images needed to achieve robust performance.

Prabu, Sundar, Jawali, Sharvani, Shanmukhanjali, and Veeramani [8] proposed using YOLOv5 to detect and recognize various denominations of Indian coins. This technique allowed accurate predictions, with the precision of 98.6% and and recall of 97.4%. In the proposed method, a comparison between other detection models was made, finding that the YOLOv5 method performed the greatest (Table 1).

TABLE I
TABLE OF PERFORMANCE METRICS WITH VARIOUS DETECTION MODELS [8]

Method	Precision (%)	Recall (%)
SSD	73.5	74.2
Faster-RCNN	72.3	70.2
Fast-RCNN	75.2	74.3
YOLOv4	86.3	74.3
YOLOv6	**98.6**	**97.4**

III. METHOD

The proposed method utilises traditional image processing techniques such as Canny edge detection and Hough circle transformation for data generation and detection, while leveraging the YOLOv8m deep learning model. The proposed method is to detect and classify New Zealand coins from images with varied conditions, such as differing backgrounds, overlapping coins, and different lighting.

A. Dataset

Large New Zealand coin datasets are not publicly available, and manual annotation of training data is time consuming and prone to human error. To overcome this, various image techniques were used to automatically annotate the data. 15 videos of New Zealand coins were recorded, each 30 seconds in length, where each video had a different variation in coin appearance. In each video, the coins were placed in different lighting conditions, backgrounds and camera angles. The positions and orientations of the coins were randomly shuffled and flipped to capture both sides of the coins. These videos were then processed, where every sixth frame was extracted to produce a training image. Automated annotation was then carried out with conversion to HSV colour space, allowing for hue-based segmentation of the copper, silver and gold coins. Traditional image techniques were used which include the Hough Circle Transform which Duda and Hart [9] found proposed a computationally efficient procedure to detect lines in images. Canny edge detection was proposed by Canny [10] to create edge detection that captures the desirable properties of an edge operator. To help detect coins, it was used to find the circular regions of the coins, allowing boundary boxes to be automatically drawn over them. To avoid any overlap from the automated detections, Euclidean distance between centers (x_i, y_i) and (x_j, y_j) was computed:

$$d = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

Two coins were considered to be overlapping if their distance was less than half the sum of their radius

$$d < 0.5 \cdot (r_i + r_j)$$

Class assignment relies on the mean hue, saturation, and relative size comparisons of each coin. This automation of the annotation diminished the issue of a large dataset needed to accurately classify coins. While the automatic annotations addressed the issue of manually drawing boundary boxes, the classifications were not highly accurate. As a result, each training image was manually reviewed and corrected where necessary to ensure the labels were accurate. The boundary boxes would then be manually reviewed to make sure they aligned with the coins. Data augmentation further addressed the issue of a large dataset, generating three augmented images per original training image to simulate real-world variations of images. The data was augmented using geometric transformations, colour, and lighting adjustments.

Geometric Transformations:

- **Rotation:** Random angles between -45° and 45°.
- **Flipping:** Horizontal or vertical flips of the image and bounding boxes.
- **Scaling:** Random scale factors between 0.8 and 1.2.

Color and Lighting Adjustments:

- **Brightness and Contrast:** Adjusted using random values for realistic lighting variance.
- **Hue and Saturation:** The HSV color space was modified to simulate different camera calibrations.

In the process of data augmentation, the boundary boxes were transformed according to each augmentation. This process lead to a large dataset, with 42,154 total annotations (Table 2)

TABLE II
SUMMARY OF IMAGE AND ANNOTATION COUNTS PER CLASS

Category	Count
Total Images	3467
Total Annotations	42,154
10 Cents	8267
20 Cents	7477
50 Cents	9432
1 Dollar	8862
2 Dollars	8116

This approach limited the impact of requiring a large dataset that was common across related works.

B. Model

Related works found pre-trained models to show high accuracy compared to other methods. Consequently, the proposed method utilises YOLOv8m, which is the medium-size variant of the YOLO (You Only Look Once) family. A YOLOv8 model was chosen for this proposed method due to its strong real-time performance and robustness on smaller datasets. YOLOv8m was selected as the best choice for the proposed method, due to the balance between accuracy and efficiency. YOLOv8m specifically has more parameters and size than the smaller models (YOLOv8n, YOLOv8s), which have high speed, but lack complex learning due to their lack of parameters (Table 3). The proposed method is to detect coins from a single image at a time in a real-world setting, which does not need large computational demands that the larger models offer (YOLOv8l, YOLOv8x).

TABLE III
YOLOv8 MODEL VARIANT COMPARISON [11]

Model	Parameters (M)	Size (MB)	Speed (ms/img)
YOLOv8n	3.2	4.4	1.5
YOLOv8s	11.2	11.2	2.5
YOLOv8m	**25.9**	**22.5**	**4.5**
YOLOv8l	43.7	46.1	6.9
YOLOv8x	68.2	99.1	9.2

YOLO was proposed by Redmon, Divvala, Girshick, and Farhadi [12] as a single neural network that can predict boundary boxes and class probabilities in one evaluation. YOLO models are CNNs, processing input images through a hierarchy of convolutional layers, where each one extracts features such as edges, textures, and general patterns. The YOLOv8 architecture is composed of a backbone that extracts features from an input image, a neck that refines the features using FPN (Feature Pyramid Network) and PAN (Path Aggregation Network) structures, and a head that performs object detection by predicting bounding boxes, scores, and class probabilities. This architecture allows for real-time detection (Fig. 4)

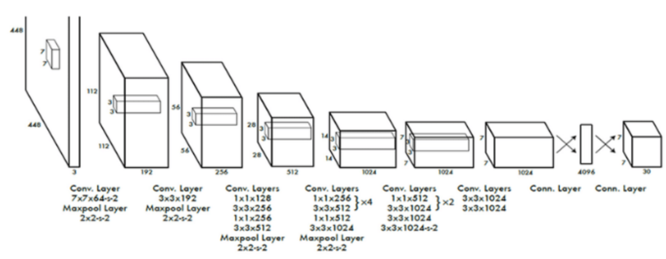

Fig. 4. A diagram of YOLO convolutional layers [13]

The annotated datasets were split into four subsets, where the YOLOv8m was then trained sequentially on each dataset, allowing the model to incrementally learn, monitoring the model as it learnt. This approach simulates a form of staged training, where the knowledge gained in earlier stages is saved and refined in future training sessions. Each training run consisted of 50 epochs, using an 80/20 split for training and validation sets, where the sets were randomly shuffled. Once trained, an unseen image can be uploaded to then sum the denominations of coins found (Fig. 5).

Fig. 5. Example output for coin detection method on an unseen image

IV. Results

Training of the model was done with the following specifications (Table 4)

TABLE IV
Training and Evaluation Environment

Device	PC
OS	Linux Mint 21.1
Processor	12th Gen Intel Core i7-12700
GPU	NVIDIA GeForce RTX 3070
Camera	iPhone 13 (12MP, 4K 60fps)
IDE	PyCharm
Language	Python
OpenCV version	4.11.0

The model was trained for 50 epochs, per 4 dataset stages, with performance metrics. There are multiple performance metrics that can be analysed to study the accuracy of the model. Mean Average Precision (mAP) can be measured, which is the average of Average Precisions over all classes from the area under the precision-recall curve [14]. It measures how accurately the model detects and classifies objects when predicted boxes overlap the ground truth by at least 50% (mAP@50). The metric of mAP@50-95 is also a performance metric that averages the mAP over the threshold from 50% to 95%.

The precision metric shows the percentage of detected coins, highlighting the amount of false positives. The recall metric shows the percentage of actual coins percentage of actual coins in the images, highlighting the amount of false negatives. In the final training stage, the best performance had an overall precision of 0.995, recall of 0.995, mAP@50 of 0.994, and mAP@50-95 of 0.91. These metrics (Table 5) reflect the high accuracy in detecting, drawing boundary boxes, and classifying the New Zealand coins. The inference time averaged 45.3 milliseconds per frame, which shows the applicability of the proposed method in a real-world environment.

TABLE V
Metrics on Test Set (Epoch 50)

Metric	Precision	Recall	mAP@50	mAP@50-95
Overall	0.995	0.995	0.994	0.91

Training losses provide another insight into the model's performance and learning process. Box loss measures how well the boundary boxes align with the actual locations of the objects in the image. Classification loss measures how well the model assigns correct classes to the detected objects. Distributed Focal Loss (DFL) measures how well the model has learnt the spatial distributions of object locations. The losses at the final stage of training at epoch 50, had a box loss of 0.5, classification loss of 0.25, and DFL loss of 1 (Table 6). These metrics suggest the model performs well at distinguishing different coins and predicting the boundary boxes of each coin.

TABLE VI
Approximate Loss Metrics at Epoch 50

Metric	Training Loss	Validation Loss
Box Loss	0.50	0.97
Classification Loss	0.25	0.32
DFL Loss	1.00	1.32

Observed was a steady decline in training losses throughout training, indicating near convergence, though the losses suggest there is potential for improved performance (Fig.5).

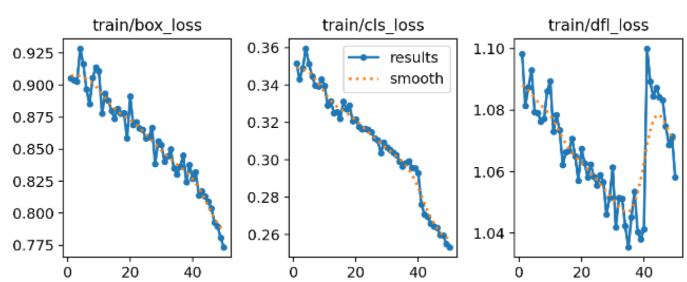

Fig. 6. Box, Classification, and DFL Loss for the proposed method

To evaluate the model's real-world performance, 50 unseen images on the front (heads) sides were tested using the trained

YOLOv8m model. The 50 images contained a total of 472 ground truth coins, with different lighting conditions. The model detected 463 out of the 472 coins, resulting in a detection accuracy of 98.3%. The undetected coins were due to darkness and shadows on the coins, or extreme glare. Per-class performance from the unseen images was strong, with 97% for 10 cent coins, 84% for 20 cent coins, 86% for 50 cent coins, 91% for 1 dollar coins, and 89% for 2 dollar coins (Table 7).

TABLE VII
PER-CLASS DETECTION PERFORMANCE

Coin Type	Total Coins	Correct Detections
10 cents	89	86
20 cents	89	75
50 cents	104	90
1 dollar	92	84
2 dollars	98	88

A confusion matrix can be made to see where the model is mis-predicting the classes [15]. In the confusion matrix, a general trend can be seen where 20 cent coins get confused the most with 50 cents, and 50 cents gets confused the most with 20 cents. There is also a trend between 1 dollar coins and 2 dollar coins being confused with each other. However, the 10 cent coin has the strongest classification and rarely gets confused with other coins. This can be attributed to its unique composition and design that is unlike any of the other coins (Table 8).

TABLE VIII
CONFUSION MATRIX FOR COIN CLASSIFICATION (HEADS SIDE)

Actual\Predicted	10c	20c	50c	$1	$2
10c	86	0	0	2	1
20c	0	75	12	2	0
50c	0	11	90	2	1
$1	0	0	1	84	7
$2	1	0	1	8	88

The difficulty of accurately detecting the 20 cent and 50 cent denominations can also be attributed to the coins being prone to glare, causing the model to struggle when presented with glare on either of the coins (Fig. 7).

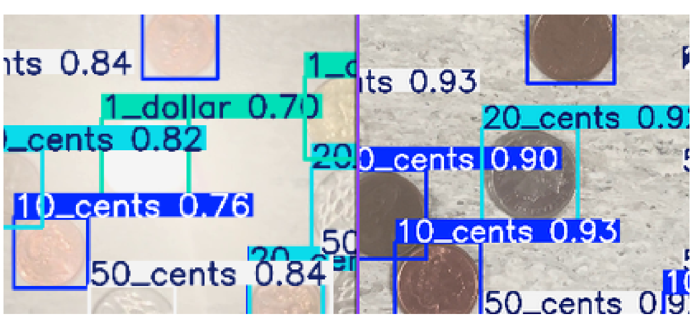

Fig. 7. Model classification for a 20 cent coin in glare vs no glare

The lower accuracy in the 20 cent and 50 cent coins compared to other denominations suggests that further refinement is needed in distinguishing visually similar coins.

50 unseen images were also tested on the back (tails) side of the coins. These images were subject to the same varying lighting conditions. The model detected 465 out of the 472 coins, resulting in a detection accuracy of 98.5%. The per-class performance was significantly reduced, which can be attributed to the lack of distinguishing features on the back (tails) sides of the coins, which the model needs to accurately make predictions. The model detected 95% of the 10 cent coins, 76% of the 20 cent coins, 75% of the 50 cent coins, 86% of the 1 dollar coins, and 71% of the 2 dollar coins (Table 9).

TABLE IX
PER-CLASS DETECTION PERFORMANCE

Coin Type	Total Coins	Correct Detections
10 cents	89	85
20 cents	89	68
50 cents	104	78
1 dollar	92	79
2 dollars	98	70

A confusion matrix can also be made for the tail-side predictions. This matrix again shows the struggle that the model shows when classifying the 20 cent and 50 cent coins. It can also be seen that the 2 dollar and 50 cent coins struggle significantly more. This can be attributed to the similar size and design on the tails side with the 2 dollar coin and the 50 cent coin. The 2 dollar coin and 50 cent coin have different compositions, however have very similar designs on the tails side, which under certain lighting conditions can cause the model to misrepresent these coins (Table 10).

TABLE X
CONFUSION MATRIX FOR COIN CLASSIFICATION (TAILS SIDE)

Actual\Predicted	10c	20c	50c	$1	$2
10c	85	0	1	2	1
20c	2	68	17	1	1
50c	3	17	78	1	8
$1	0	1	1	79	11
$2	0	0	9	19	83

These tests suggest that the model performs well in controlled environments where the coins are front (heads) side up in order to make accurate predictions. For real-world use cases such as banking or vending machines, this is not realistic as not all coins will be oriented in this way.

V. CONCLUSIONS

This study proposed a method for automated detection and classification of New Zealand coins using YOLOv8m, achieving a precision of 0.995, recall of 0.995, and mAP@50 of 0.994

Despite the limitations of the model's ability to mis-predict coins due to different orientations and lighting conditions, the current model demonstrates strong feasibility for deployment in controlled environments. With an overall precision and recall of 99.5%, the model is capable of performing accurate, real-time detection and classification of New Zealand coins. This is feasible in smaller tasks such as an aid for cashiers to help count the amount of change given.

In comparison to the related works, an improvement in the proposed method is the creation of large datasets due to automated annotations and data augmentations. Many related works struggled with limited datasets that did not correlate well with real-world conditions. This lack of data leads to many models underperforming. However, the proposed method enables the annotation of large datasets of New Zealand coins that reflect real-world conditions through varied lighting and orientations. The use of the YOLOv8m model achieved high detection accuracy, where other related works relied on traditional image processing techniques that dealt with many challenges such as edge detection, that the YOLOv8m model could overcome.

A. Future Research

Future work done on the model could enhance the robustness of the model in real-world environments. The challenge of various lighting and glare that impacts the model's ability to detect and classify the coins would need to be addressed further. Increased data augmentation techniques would need to be expanded, techniques such as glare removal would greatly improve the model's performance. Expanding the dataset to more diverse environments in lighting and backgrounds would further improve the model's performance. Another area for improvement is the model's ability to correctly classify coins regardless of their orientation, as it is seen that the per-class accuracy varied greatly between the front (heads) and back (side) facing up. Future research should include a larger dataset, and image processing techniques to help the model perceive distinguishing features.

References

[1] Reserve Bank of New Zealand. "Coin Specifications and Images by Denomination." Internet: https://www.rbnz.govt.nz/money-and-cash/banknotes-and-coins/coins-in-circulation/coin-specifications-and-images-by-denomination, [May 8, 2025].

[2] Te Ara - The Encyclopedia of New Zealand. "Story: Coins and Banknotes" Internet: https://teara.govt.nz/en/photograph/36410/current-coins, [May 8, 2025].

[3] New Zealand Post. "A Brief History of New Zealand Currency." Internet: https://collectables.nzpost.co.nz/a-brief-history-of-new-zealand-currency/?srsltid=AfmBOorgaEmGcUjPSCvjaSb-CDd2qwL5sFutBlMHy5Ek7DENCUqY76jD, [May 10, 2025].

[4] L. van der Maaten and E. Postma, "Towards automatic coin classification," in *Proc. EVA-Vienna*, Vienna, 2006, pp. 2-8.

[5] M. Nolle, H. Penz, M. Rubik, K. Mayer, I. Hollander, and R. Granec, "Dagobert – A New Coin Recognition and Sorting System," in *Proc. 7th Int. Conf. Digital Image Comput. Techniques and Applications (DICTA)*, Macquarie Univ., Sydney, Australia, Dec. 2003, pp. 2-5

[6] J. Kim and V. Pavlovic, "Discovering characteristic landmarks on ancient coins using convolutional networks," *arXiv preprint arXiv:1506.09174*, 2015.

[7] N. Capece, U. Erra, and A. V. Ciliberto, "Implementation of a coin recognition system for mobile devices with deep learning," in *Proc. 12th Int. Conf. Signal-Image Technology Internet-Based Syst. (SITIS)*, Naples, Italy, Nov. 2016, pp. 3–7.

[8] S. Prabu, K. J. A. Sundar, N. Jawali, K. Sharvani, G. Shanmukhanjali, and N. Veeramani, "Indian coin detection and recognition using deep learning algorithm," in *Proc. 6th Asian Conf. Artif. Intell. Technol. (ACAIT)*, Dec. 2022, pp. 1–8.

[9] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Commun. ACM*, vol. 15, no. 1, pp. 11–15, Jan. 1972.

[10] J. F. Canny, "Finding edges and lines in images," AI Tech. Rep. AITR-720, MIT, Cambridge, MA, Jun. 1983. [Online]. Available: http://hdl.handle.net/1721.1/6939

[11] Ultralytics, "YOLOv8 Documentation," Internet: https://docs.ultralytics.com/models/yolov8/, [May 11, 2025].

[12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *arXiv*, vol. abs/1506.02640, 2016. [Online]. Available: https://doi.org/10.48550/arXiv.1506.02640.

[13] C. Kwan, B. Chou, J. Yang, A. Rangamani, T. Tran, J. Zhang, and R. Etienne-Cummings, "Target tracking and classification using compressive sensing camera for SWIR videos," *Signal Image Video Process.*, vol. 13, pp. 1629–1637, Jun. 2019.

[14] S. Srivastava, A. V. Divekar, C. Anilkumar, I. Naik, V. Kulkarni, and V. Pattabiraman, "Comparative analysis of deep learning image detection algorithms," *J. Big Data*, vol. 8, Art. no. 66, 2021.

[15] J. Liang, "Confusion Matrix: Machine Learning," *POGIL Activity Clearinghouse*, vol. 3, no. 4, Dec. 2022. [Online]. Available: https://pac.pogil.org/index.php/pac/article/view/304