Rare Coin Grading using Convolutional Neural Networks

Orhan Hosten-Mittermaier
College Preparatory School, Oakland, California
orhanhm13@gmail.com

Olivia Lee
Department of Computer Science, Stanford University, California
oliviayl@stanford.edu

***Abstract* – Rare coins differ greatly in value based on their quality, leading to the development of a standardized 1-70 scale on which coins are graded. The grading process, though, is time and resource-intensive, making it far too expensive for a casual coin collector to get a coin graded. To democratize this process, we developed a computer vision model that grades coins based on an image alone, which makes grading a coin essentially free of cost and provides a dramatic reduction in time. Our dataset consisted of labeled image data from Professional Coin Grading Services, on which we applied affine geometric data augmentations to mitigate possible domain shift on new, "imperfect" images. We experimented with training several different convolutional neural network architectures on this data. Furthermore, we utilized several metrics to effectively evaluate and compare different coin grading models. Our final accuracy was over 50%, and 77% of the predictions were within one grade of the correct grade, affirming that a model sufficiently performant for consumer use is possible. Future work could involve crowdsourcing natural datasets of coin images and making a web or mobile application interface to make our model available to the public.**

Index Terms – Computer Vision (CV), Convolutional Neural Networks (CNNs), data augmentation, Professional Coin Grading Service (PCGS), rare coin grading

Introduction

The primary objective of this paper is to explore the viability of using modern computer vision techniques to automate the rare coin grading process. The reason behind this is the prohibitive cost of current coin grading methods. In the coin-collecting world, coins can be worth drastically more or less based on their level of circulation and wear. Because the quality of a coin is so integral to determining its value, a 1-70 scale has been developed to assess coins. The initial reason behind the 1-70 scale is that a grade 70 coin should be worth about 70 times more than a grade 1 coin. While this scaling does not hold true in practice, the grading system nevertheless stuck and is now what is generally used [1]. However, this grade can only be accurately determined by professional graders from companies such as Professional Coin Grading Services (PCGS). The coins must be securely shipped to PCGS before they are painstakingly inspected by professional human graders in a multi-step process. Therefore, the current grading process is expensive and slow, making it inaccessible or unfavorable to the majority of casual coin collectors. Most of the time, the cost to grade a coin is a substantial fraction of the value of the coin itself, except in the case of high-value coins owned by serious coin collectors. For example, the absolute lowest price to get a single coin graded with PCGS is about $40 and takes over a month at this price point [2]. To make this aspect of rare coin collecting more accessible to the casual collector, the grading process needs to be revamped.

Additionally, very little research has been conducted in the past applying machine learning, especially deep learning models like convolutional neural networks (CNNs), to rare coin grading. While automated coin grading systems analyzing "fuzzy" spots on coin imaging or color histograms of coin images have been applied to coin grading, very little work has been done in this area using CNNs [3, 4]. The field has essentially not been explored using modern machine learning models and taking advantage of increases in computing capabilities in recent years.

Therefore, our primary research aim is to explore the viability of developing and training a deep learning model that assigns grades to coins with high accuracy based on an image alone. We hypothesize that a computer vision model that meets this criterion would drastically decrease the cost and time needed to grade a coin. Additionally, we hope to develop and test scoring metrics that accurately evaluate a model's performance

for the coin grading task at hand. Given the limitation of a small dataset, our primary objective is to present a proof of concept to stimulate further research in the area and to demonstrate that a sophisticated coin-grading computer vision model is feasible. After further testing and improvements to our model, we intend to release it for public use, most likely in the form of a mobile application to provide casual collectors access to this aspect of coin collecting.

Data

Despite rigorously searching for an online dataset containing labeled coin image data that had previously been used for a similar application, we were not able to find any such dataset. Fortunately, PCGS has numerous image examples of many different coin varieties on their website, as well as the associated grades. Using this, we developed a program to scrape through the website and download as many image-grade pairs as possible. The result was 90,000 obverse and reverse[1] image pairs of US coins including pennies, nickels, dimes, quarters, and more, all from the past 150 years, along with their associated grade (see Figure 1).

Figure 1: Sample obverse-reverse image pair before data augmentation

However, the images were all shot straight-on, with perfectly centered, unrotated coins taking up the majority of the frame. We predicted that this "perfect" data would create an issue with domain shift later on, since images taken on a consumer's smartphone, for example, would probably not have the coin perfectly centered or be shot straight on. These images taken by hand would most likely look very different from the training images and lead to impaired performance on the new images. Therefore, we utilized simple geometric data augmentations as outlined by Shorten and Khoshgoftaar to diversify the dataset and make the images more similar to what a consumer might send in [5]. The data augmentations we used were affine transformations including rotation, scaling, shear, and reflection, applied randomly throughout the dataset.

In addition to the transformations of the images, the dataset size was altered as well because multiple new, unique images were created from each "perfect image." The primary reason for this was to even out the distribution of data points across the grades. Data was scarce for low grades in our initial dataset, which could have had detrimental effects on the classification accuracy for those grades specifically. Additionally, image resolution was decreased to 256x256 to limit the size of the model we would need and therefore decrease training time. Finally, the obverse and reverse images were combined into a single 256x512 image for ease of processing. As a result, our final working dataset contains around 220,000 data points, each consisting of a 256x512 color image, along with the coin's grade as the label.

Models

We used a variety of machine learning models during our testing. Our baseline model was a scikit-learn SGD classifier, referred to as "SGD," with logistic loss. This was used to establish a starting accuracy from which we could improve with larger, more sophisticated deep learning models. The model is essentially a logistic regression classifier with minibatch stochastic gradient descent. Given the size of our dataset and the storage-heavy nature of image data, we were unable to load the whole dataset to RAM at any given time and therefore needed to use minibatches for all of our models.

In addition to the baseline SGD classifier, the main models we used were all convolutional neural networks. CNNs are very good at feature extraction with pixel data and have grown in popularity for computer vision applications as a result [6]. As part of testing to maximize grading accuracy, we experimented with three different CNN architectures with increasing depth and kernel sizes (see Figures 2, 3, 4). We progressively increased the complexity of our models to minimize overfitting, especially because the dataset is relatively small compared to those of many computer vision projects.

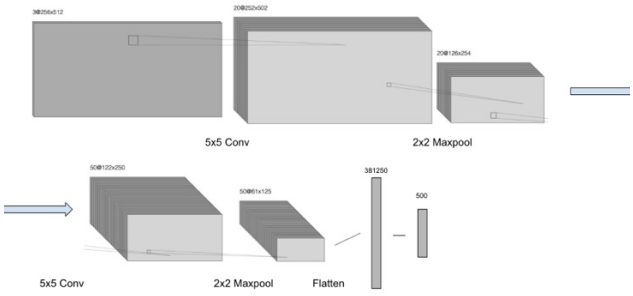

Figure 2: CNN architecture with two convolutional layers. Referred to as "CNN1"

[1] Obverse and reverse are technical terms for the front and back of a coin

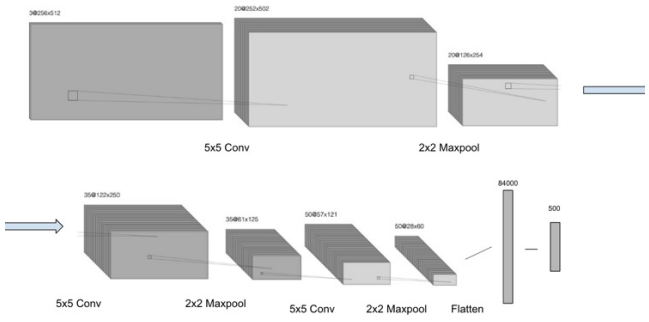

Figure 3: CNN architecture with three convolutional layers. Referred to as "CNN2"

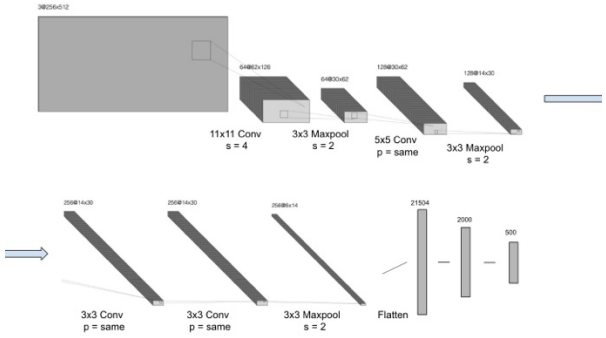

Figure 4: CNN architecture with four convolutional layers. Inspired by the AlexNet architecture developed by Krizhevsky, et al [7]. Referred to as "CNN3"

Lastly, the combination of additional convolutional layers and larger kernels substantially decreases the size of our first fully connected layer in CNN3, which contains 21504 neurons, as opposed to 381250 and 84000 neurons for CNN1 and CNN2, respectively. In all models, this was by far the largest layer in terms of trainable parameters and therefore determined both the RAM usage of the model and the batch size we were able to use. Because of the smaller first fully connected layer of CNN3, we were able to increase the batch size we trained with from 64 to 128, allowing us to train the model for more epochs compared to the others.

Metrics

To evaluate our three models, we used a collection of direct accuracy, confusion matrices, and two metrics we developed ourselves. The first special scoring metric is a *bucket accuracy*. To make the confusion matrices more legible, we grouped our data into five buckets of either five or six grades so that the confusion matrices were 5x5 instead of 29x29[2]. From this, we defined the bucket accuracy score as the percent of the time the model predicts a given data point to be in the correct bucket. The bucket accuracy provides info on whether a model, although it might have low direct accuracy, is at least getting close to the correct grade, as one might expect a human to act. In terms of eventual practical usage, having a model that predicts grades close to the correct grade if it is going to be wrong is preferable over one that spits out seemingly random guesses for its incorrect answers. While the bucket accuracy metric helps determine whether this is true about a model, it is also a product of our slightly arbitrary choices of buckets and is not perfect for the aforementioned goal.

Therefore, we utilized an additional *within one* or W1 score that tracks the percentage of the time a given model predicts either the correct grade or is one away (i.e. one grade too high or low). This resolves the issue of arbitrary buckets, since we are always considering just the grades directly neighboring the correct grade. As a result, the percentages are lower than the broader bucket accuracy, but the score is more informative. A high W1 score corresponds to the type of performance we want the models to exhibit, since it is favorable for the model to be close to the correct grade if it does not predict correctly.

Results

Model	Raw Test Accuracy	Bucket Accuracy	**Within 1 (W1)**
SGD	.10	.40	**.18**
CNN1	.26	.72	**.57**
CNN2	.39	.79	**.68**
CNN3	.52	.85	**.77**

Figure 5: Summary of results from our four models

Through the lens of these metrics, we evaluated our models on novel testing data that had not been previously fed to the models as either training or validation data. (See Figure 5 for a summary of all results). Given the comparatively low predictive power of our baseline SGD classifier, we were not surprised by the low performance.

Although the raw accuracy of 10% is low, the 40% bucket accuracy demonstrated that even a small, less sophisticated model made guesses relatively close to the actual grade approximately half of the time. In the confusion matrix, a loosely defined band of high numbers on the upper-left to bottom-right diagonal is present

[2] An attentive reader might realize that the professional grading system has 70 grades, while we only had 29. For unclear reasons, PCGS only assigns coins to 30 numerical grades on the 1-70 scale, of which we had sufficient data for all grades except grade 60, which we had to omit. The grades we used were arranged into the buckets: [1, 2, 3, 4, 6, 8], [10, 12, 15, 20, 25, 30], [35, 40, 45, 50, 53, 55], [58, 61, 62, 63, 64, 65], [66, 67, 68, 69, 70].

(see Figure 6). This corresponds to all the data points that have been classified into the correct bucket. Furthermore, the general trend of fewer guesses in the upper-right and bottom-left corners indicates that the model is not predicting grades that are very wrong often.

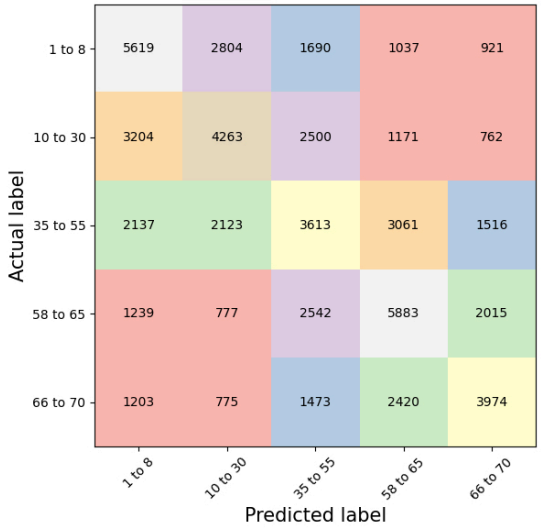

Figure 6: Bucket confusion matrix of testing results from the baseline SGD classifier model

CNN1, our first deep learning model, yielded the largest improvements in model performance overall. Direct test accuracy increased by a factor of 2.5, and the W1 score improved from 18% to 57% in comparison to the baseline model. CNN2 saw smaller but significant increases in all metrics, including identifying the correct bucket of a new image over 79% of the time and reaching a W1 percentage of 68%. In terms of the confusion matrices, we see the top-left to bottom-right diagonal become more pronounced, especially with CNN2 (see Figures 7, 8).

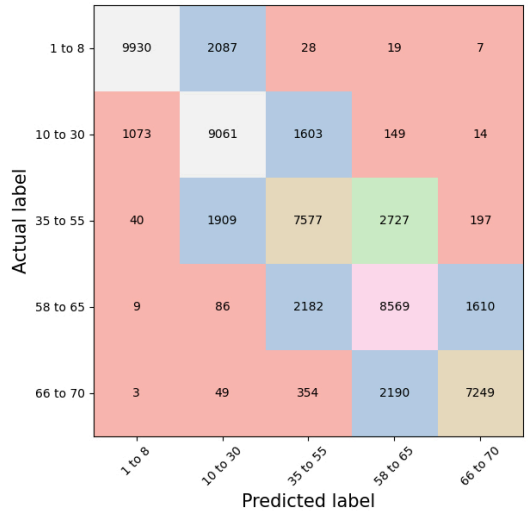

Figure 7: Bucket confusion matrix of testing results from CNN1.

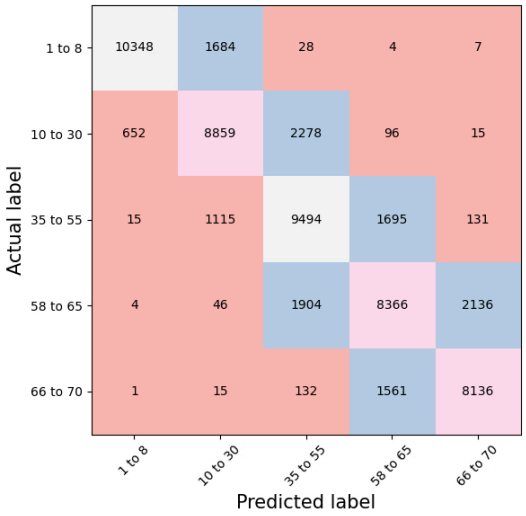

Figure 8: Bucket confusion matrix of testing results from CNN2.

Our final and best-performing model was CNN3, the model with the most convolutional layers. Part of this success comes from the lower RAM usage which allows us to train with larger batch sizes and therefore for more epochs (see Models section). This, in combination with careful validation accuracy monitoring to watch for overfitting, yielded the best results in all metrics and the most convincing confusion matrix. CNN3 had a direct accuracy of 52% and a W1 score of 77%, meaning it correctly graded the novel test images over half the time and was only one grade away an additional quarter of the time. As for the confusion matrix, the diagonal trend is the most refined out of all the models, and there are less than 150 predictions that are more than one bucket away from the correct bucket (see Figure 9).

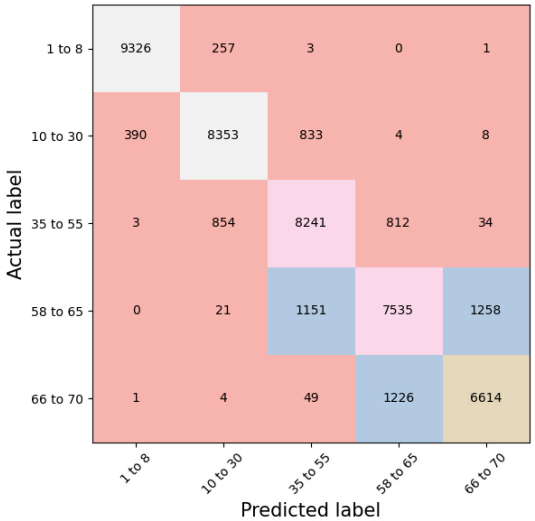

Figure 9: Bucket confusion matrix of testing results from CNN3.

Discussion and Conclusions

Overall, the test results support our hypothesis that a high-accuracy coin grading model is possible with current technologies. Given the dataset size and computing power limitations of our experiments, the 52% direct accuracy and 77% W1 score results that we achieved demonstrate that a high-performance model is feasible, and we will go into further detail on how this could be achieved. Additionally, our experimentation with scoring metrics proved insightful for our research goal of developing a set of metrics to evaluate a given model's performance. Both of the well-known evaluation techniques, direct accuracy and confusion matrices, proved useful. More interesting were the results of the bucket accuracy and W1 score. Especially for the deep learning models, the testing data confirms our prior belief that bucket accuracy is less helpful for the better-performing models because it is not granular enough to pick up improvements in performance the same way raw test accuracy or W1 can. On the other hand, we established W1 as a meaningful metric that can be used to evaluate our models and capture performance improvements.

As we realized in our training, one major bottleneck was the training speed of the models. Faster computing and more time would allow for more training and better results. Additionally, a larger computing array could be used for higher-resolution images, which would help models pick up the minute details that can affect a coin's grade. Further experimentation with CNN architecture and more rigorous hyperparameter optimization would both help increase model performance. Another future direction is to explore the use of visual transformers, which are gaining prominence for CV tasks [8].

With more sophisticated model architectures, gathering more labeled image data will be vital. We could potentially crowdsource image data from individual coin collectors to curate a larger, natural dataset. This would introduce more realistically imperfect data that would circumvent any domain shift issues.

As previously mentioned, we hope to eventually release this technology to the public through a web or mobile application interface. While we will continue to work on this project, we hope that our work can act as a precursor for further experimentation by other researchers as this will further develop the field and eventually lead to a consumer-accessible, high-performance coin grading model that democratizes the grading process.

Acknowledgements

We want to acknowledge the Professional Coin Grading Service for providing a large amount of high-resolution image data of rare coins on its website for public use. For obvious reasons, this project would not have been possible because we wouldn't have had any data.

1 Fagaly, Robert L. “Pricing Relationships of United States Type Coinage.” American Journal of Numismatics (1989-), vol. 23, 2011, pp. 257–63. JSTOR, http://www.jstor.org/stable/43619982. Accessed 12 Dec. 2023.

2 “PCGS Services.” *Professional Coin Grading Services*. https://www.pcgs.com/services. Accessed 12 Dec. 2023.

3 Pan, Xingyu and Laure Tougne. "Image analysis and deep learning for aiding professional coin grading." 2018 International Conference on Image and Video Processing, and Artificial Intelligence, proc. vol. 10836, 29 Oct. 2018. https://doi.org/10.1117/12.2500142. Accessed 12 Dec. 2023.

4 Basset, Rick, et al. “Development of an automated coin grader: a progress report.” Mid-Atlantic Student Workshop on Programming Languages and Systems, proc. Vol. 15, 19 Apr. 2002. http://www.rickbassett.com/publishing/published/Deployment%20of%20an%20Automated%20Coin%20Grader.pdf. Accessed 13 Dec. 2023.

5 Shorten, Conner, and Taghi M. Khoshgoftaar. “A survey on Image Data Augmentation for Deep Learning.” Journal of Big Data, vol. 6, 6 Jul. 2019. https://doi.org/10.1186/s40537-019-0197-0. Accessed 13 Dec. 2023.

6 Ramprasath, Muthukrishnan, M. Vijay Anand, and Shanmugasundaram Hariharan. “Image classification using convolutional neural networks.” International Journal of Pure and Applied Mathematics, vol. 119, 2018. https://www.researchgate.net/profile/Ram-Prasath-13/publication/357974255_Image_Classification_using_Convolutional_Neural_Networks/links/61e9a1ecdafcdb25fd3c6e62/Image-Classification-using-Convolutional-Neural-Networks.pdf. Accessed 13 Dec. 2023.

7 Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. “ImageNet classification with deep convolutional neural networks.” Advances in Neural Information Processing Systems, 2012. https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf. Accessed 13 Dec. 2023.

8 Liu, Yang, et al. "A Survey of Visual Transformers." *ArXiv*, 2021, https://doi.org/10.48550/arXiv.2111.06091. Accessed 26 Dec. 2023.