Bangladeshi Paper Currency Recognition Using Lightweight CNN Architectures

1st Tazwar Mohammed Shoumik
tazwar.mohammed.shoumik@g.bracu.ac.bd

2nd Sartaj Jamal Chowdhury
sartaj.jamal.chowdhury@g.bracu.ac.bd

3rd Tanzim Mostafa
tanzim.mostafa@g.bracu.ac.bd

4th Adib Muhammad Amit
adib.muhammad.amit@g.bracu.ac.bd

5th Oyshik Ahmed Aadi
oyshik.ahmed.aadi@g.bracu.ac.bd

6th Shah Abul Hasnat Chowdhury
shah.abul.hasnat.chowdhury@g.bracu.ac.bd

7th Sifatul Amin
sifatul.amin@g.bracu.ac.bd

8th Md. Humaion Kabir Mehedi
humaion.kabir.mehedi@g.bracu.ac.bd

9th Shadab Iqbal
shadab.iqbal@g.bracu.ac.bd

10th Annajiat Alim Rasel
annajiat@gmail.com

***Abstract*—Paper note currency is the most ubiquitous way of completing a transaction globally, and this applies to most developing countries such as Bangladesh. Hence, it is essential to recognize any of the currency notes within minutes to save time and an uninterrupted transaction. This paper aims to recognize different banknotes of Bangladesh using the avant-garde CNN models with transfer learning by utilizing the dataset at hand and updating and augmenting the dataset. In order to achieve a robust, lightweight, and efficient model for this research problem, we used various augmentation techniques on a current publicly available dataset and applied custom hyperparameter tuning to pre-trained CNN models to attain a maximum accuracy of 100%.**

***Index Terms*—Bangladeshi Paper Currency recognition, Bangladeshi Banknote classification, Deep Learning, Transfer Learning, Convolutional Neural Network, Multi-class Classification**

I. Introduction

Paper currency is the monetary symbol for transactions in any country, and Bangladesh is no exception. The paper currencies have gone through several changes throughout the decades due to transposition on many sides, i.e. government. The banks and other establishments must recognize all the paper notes in regulation to complete transactions. This technology has been tinkered with before, but our aim is to see if we can push the boundaries in building a robust and accurate model and make an updated and augmented dataset that we can then make publicly available for researchers worldwide for their contribution to this research topic. The existing dataset is not updated because the newly introduced the 200 taka note that is now in regulation is missing from the existing dataset we are using. It also does not contain some key Bengali paper currency data in regulation, and there is not enough data for each banknote, and we hope to mitigate that.

In the following steps in this paper, we will discuss the past and current works on this specific topic, the methodology, which includes the datasets, and the methods and models we will be using to accomplish our research objective. Then we will discuss any future research opportunities in this field of interest and if any improvements can be implemented in this field of research.

II. Existing Works

The classification of currency has been previously attempted before. Over time, many researchers have played a role in currency recognition, incorporating various features such as color, texture, anti-spoofing, and many more.

In [1] and [5], a convolutional neural network was used, and for [1] specifically, they tested their methods on every denomination of India with a 95% level of accuracy. That said, other forms of learning, such as Deep Learning or Transfer Learning, were also used, such as that in [2] where the author also mentioned that most other methods often fail when it comes to damaged notes. In [3], K-NN based classification for the denomination was used for identifying banknotes after extracting color and texture features. Moreover, some papers were also concerned in the speed of recognition of these currencies such as in [4], which used YOLO V3 architecture for faster identification. As demonstrated by [6], currency notes can also be identified by using a unique identification mark. This mark can exist as specific shapes.

The paper [7] classified four currencies, namely, “pound,” “yen,” “euro,” and “rupee,” using Artificial Neural Network (ANN). For this, they first preprocessed the images using a Median filter to denoise the images. In the feature extraction stage, they used Sobel masks and Canny edge detection to find out specific edges. Finally, they identified the currencies using 100 samples of each currency kind to train a feedforward backpropagation neural network. Through this method, they achieved an average recognition rate of 93.84%. However, their system lacked high processing speed.

In the paper [8] the authors carried out experiments using three lightweight CNN models with Transfer Learning, the models being pre-trained on the 'ImageNet' dataset.' They used ResNet152v2, MObileNet, and NASNetMobile, to recognize Bangladeshi notes from two different datasets. They achieved 98.88% accuracy on the 8000 image dataset using MobileNet and 100% on the 1970 image dataset using NASNetMobile. Finally, they combined the two datasets where they found that MobileNet performed the best, achieving 97.77% accuracy. According to their results, Transfer Learning helped immensely with detecting banknotes. Nonetheless, their model struggled to recognize notes in case of complex background and arbitrary lighting.

AlexNet, a pre-trained Convolutional Neural Network (CNN), was utilized in the paper [9], to recognize Indian coins and classify them into four categories, namely, one, two, five, and ten-rupee coins. According to their results, only 1000 coin images of all instances are enough for training the model. Moreover, they proposed an architecture that involves high-speed classification suitable for real-time requirements, and they plan to deploy their algorithms in real-time embedded systems.

The authors of the research [10] looked at how orientation and region affected banknote identification. With Malaysian Ringgit notes as their dataset, they used a variety of classification techniques, including k-Nearest Neighbor (kNN), Decision Tree Classifier (DTC), Support Vector Machine (SVM), Bayesian Classifier (BC), and CNN model AlexNet. DTC and kNN had 99.7% accuracy, whereas SVM and BC had 100% accuracy, according to their findings. They also found that while AlexNet worked admirably with 100 percent accuracy on a database of notes with identical orientations, it underperformed when tested with notes of diverse orientations.

III. Methodology

This section demonstrates the procedures taken to acquire Bangladeshi paper currency images and classify them into their respective categories. We used four state-of-the-art CNN models to compare their accuracy and losses. All these models have under 8 million parameters, making them light enough to be implemented on IoT and embedded devices. Furthermore, the source and how the dataset was used for research purposes are provided in the subsections below.

A. Dataset Acquisition

The dataset we used is Kaggle's open-access dataset called "Bangla Money" [11] under Taka recognition datasets. It was created in 2018 for a project named "Bangla Taka Recognition." It consists of nine target variables: Bangladeshi paper currency denominations of 1Tk, 2Tk, 5Tk, 10Tk, 20Tk, 50Tk, 100Tk, 500Tk, and 1000Tk excluding the class of 200Tk as shown in Fig. 1.

Mobile devices of unknown models were used to capture a total of 1970 banknote images under different lighting conditions. Some of the banknotes in the dataset are worn out, old, and had images of both sides, which enabled the model to generalize well on the features of any class of banknotes, enabling it to correctly classify different paper currency denominations regardless of the note being worn out and old. Furthermore, some of the banknotes vary between the respective classes. For example, there are two different visual representations of 2Tk and 5Tk which further helps the model to learn the features of different banknotes within the same denomination. The dataset is divided into 80% training data(1637 images) and 20% for testing purposes(333 images). Each image was captured at various resolutions; hence it was cropped and scaled down to a resolution of 120, 250, 3 which means the resulting image is an RGB image.

Fig. 1. Nine different Bangladeshi Paper Currency Denomination in Bangla Money Dataset

However, the existing dataset has some discrepancies. Each target class has around 100-250 images of training samples upon which the CNN models will be trained. The class of 1Tk has 102 training images, and the class of 50Tk has 214 training images. Due to this inconsistency in training data, the features of images in the successive convolution layers of the CNN models will also be "learned" inconsistently across different classes, which might result in incorrect classification, hence decreasing the model's overall accuracy. The more the training data of a target class, the better the model can learn to classify that variable. Hence using this dataset, theoretically the model will learn to classify 50Tk with better overall accuracy compared to 1Tk, which is undoubtedly not sensible in the real world.

B. Data Pre-Processing

Various methods were used on the Bangla Money dataset to account for the inconsistencies present which could result in a misleading accuracy score or overfitting of the models. Firstly, we augment the dataset to increase the training samples of each currency denomination. Afterwards, we added a new target class which increased the number of total target variables from 9 to 10. Lastly, we split the augmented dataset into training and validation sets for the model to validate its training accuracy.

1) Data Augmentation: As the Bangla Money dataset contains a significantly smaller number of training images com-

Target Variables	Original	Augmented
1Tk	101	1000
2Tk	213	1000
5Tk	213	1000
10Tk	213	1000
20Tk	173	1000
50Tk	213	1000
100Tk	208	1000
200Tk	0	1000
500Tk	136	1000
1000Tk	167	1000

TABLE I
NUMBER OF SAMPLES IN BANGLA MONEY AND AUGMENTED DATASET

Parameters	Values
Rescale	1.0/255.0
Rotation Range	45
Width Shift Range	0.2
Height Shift Range	0.2
Shear Range	0.1
Horizontal Flip	True
Vertical Flip	True
Brightness Range	(0.8, 1.2)
Channel Shift Range	0.2
Fill Mode	'nearest'

TABLE II
AUGMENT OPERATIONS USING IMAGEDATAGENERATOR

pared to other popular datasets such as ImageNet(1.2 million images), MNIST(60k images), and CIFAR-10(50k images), we perform data augmentation on the training images such that each class label has exactly 1000 RGB images. Table I compares the number of training samples for each category of banknote in the Bangla Money dataset and the augmented dataset. Firstly, 1000 images were artificially produced from the original images in each class using the Augmentor package in Python, with each image having a size of 224 x 224. Afterwards, ImageDataGenerator from TensorFlow was used to perform several augmentation techniques(in Table II) randomly on each sample and split the dataset into 80% and 20% for training and validation, respectively. Fig. 2 shows a comparison between the original and augmented datasets.

Fig. 2. Samples before and after Augmentation

2) Addition of 200Tk: We added a new target variable to the existing dataset. This new target variable is the new paper currency of the 200Tk banknote of Bangladesh, released in early 2020. Fig. 3 shows a visual representation of both sides of a Bangladeshi 200Tk banknote. We manually take 30-40 photos of 200Tk notes with cameras from smartphones. Then we augment the images of this class up to 1000 images using the augmentation techniques as shown in Table II. This makes a total of 10 classes in our dataset, which equals the number of classes in the CIFAR-10 dataset. This addition brings novelty to the dataset and the research of classification of the paper currency of Bangladesh. Fig. 4 shows the updated paper currency denomination of Bangladesh, which was used for this research. The Kaggle link to the dataset used for this research is as follows: https://www.kaggle.com/datasets/tazwarmohammed/augmented-bangla-money-dataset-including-200-taka

Fig. 3. Both sides of 200Tk Banknote of Bangladesh

Fig. 4. New Currency Denomination

C. Convolutional Neural Network

We used four lightweight CNN-based architectures for our research objective and compared their accuracies and losses. The four architectures are as follows: EfficientNetV2B1, MobileNetV3Large, DenseNet121 and NASNetMobile. All these models are available to use via the Keras API from TensorFlow. We fine-tuned these models and used transfer learning to make use of previously trained weights on our research problem. Transfer Learning is a technique for using previously learnt information to tackle new issues quickly and effectively. A deep learning model was trained from scratch to recognize cats, for example. This model can also be used to detect dogs very quickly only if transfer learning is used. Transfer learning is applied when the model's weights, used to detect cats, are used as a starting point to train the model to detect dogs. It is not always a wise choice to use transfer learning, but in the case of cats and dogs, transfer learning helps a lot as the shape

Model Name	Trainable Params(M)	Layers
EfficientNetV2B1	6.9	337
MobileNetV3Large	5.5	273
DenseNet121	7.0	429
NASNetMobile	4.2	771

TABLE III
PARAMETER AND LAYER COUNT MODELS USED

and features of a cat and dog are very much similar, e.g., the position of their head, organs, and limbs.

All these models were previously trained on the ImageNet dataset and achieved remarkable top-1 and top-5 accuracies, considering they have less than 8 million parameters. Table III shows the parameter and layer count of each model. Before training, we removed the output layer from all the models, which had 1000 nodes because ImageNet has 1000 target variables, and replaced them with layers of 10 nodes and softmax activation function since our dataset has ten target values. Then we made all the layers in the model trainable. Even though ImageNet does not have training samples of banknotes, we used the weights associated with the training of ImageNet as a starting point for all the models to train on our dataset instead of randomly initializing the weights. This resulted in a faster convergence as the model has already been trained to detect edges and features of numerous objects in ImageNet. Hence, the model does not have to learn from scratch to detect and classify banknotes, which would otherwise require a significantly higher number of epochs for the model to converge. The subsections below provide a brief history and description of the models' distinctive characteristics that we employed for our research.

1) EfficientNetV2B1: Introduced in 2021 at the 38th International Conference on Machine Learning, this model is the latest of all the other models used in this research. The EfficientNetV2B1 model is one of the members of the family of EfficientNetV2, which is an improved version over the EfficientNet [17].

2) MobileNetV3Large: The MobileNetV3 network was introduced at the ICCV 2019 as a successor to the MobileNetV2. The MobileNetV3 has two networks: MobileNetV3Large and MobileNetV3Small for high and low resource use cases, respectively. MobileNetV3Large is 20% faster, in terms of training, and 3.2% more accurate on ImageNet classification compared to MobileNetV2. MobileNetV3 uses a combination of blocks from MobileNetV2 [18] and MnasNet [19], which are the "Inverted Residual and Linear Bottleneck" and the "Squeeze-and-Excite" respectively.

3) DenseNet121: The DenseNet architecture was introduced at the Conference on Computer Vision and Pattern 2017. DenseNet consists of 4 individual networks of different number of layers and parameters: DenseNet-121, DenseNet-169, DenseNet-201 and DenseNet-264. The main goal of DenseNet is to make sure that as much information as possible flows between layers of the network by having direct connections from one layer to every other layer inside a "dense" block.

4) NASNetMobile: In 2018, NASNet's research was presented at the Conference on Computer Vision and Pattern Recognition. The key distinguishing property of NASNet is the introduction of "NASNet Search Space," which allows transportability from dataset to dataset. With this model, a new regularization technique, ScheduledDropPath, is also introduced which greatly enhances the generalization when learning from a dataset. NASNet has two models: NASNet-Large and NASNetMobile for high and low resource use case respectively.

As all the models had previously been trained on the ImageNet dataset, and we utilized the 'ImageNet' weights as a starting point for the models to train, all of the images had to be 224 × 224 pixels in height and breadth to be fed into the models as input. EfficientNetV2B1, the latest of the four models, was trained on ImageNet pictures with a resolution of 240 by 240 pixels. As a result, we had to feed EfficientNetV2B1 photos from our dataset with the same dimensions.

Except for DenseNet121, we utilized a batch size of 128 for all models; since we did not have access to enough computing resources to train such a huge batch size in the heaviest model among the four, hence we opted to train DenseNet121 with a batch size of 64. We used the Adam optimizer with an initial learning rate of 0.001 and categorical cross-entropy as the loss function since our research problem is a multi-class classification problem. The image labels were one-hot encoded before inputting the batches into the model. Later on, all the models were trained for 100 epochs and converged at random epochs, which will be discussed in detail in the next section.

A callback method called ReduceLROnPlateau was used to keep track of the learning rate during the training phase. The ReduceLROnPlateau callback function allows the learning rate to be decreased by a given factor if the model's validation loss or accuracy score fails to increase. We set the ReduceLROnPlateau function parameters such that the learning rate will reduce by a factor of 0.5 if the validation accuracy does not increase for five consecutive epochs. This will help the model to converge, given that with the previous larger learning rate, the model was overshooting the loss function, thereby not being able to reach the global or local minima. When the model is about to reach a global or local minimum point in the loss function, and the accuracy decreases in the following epoch, the model has a higher learning rate that needs to be reduced to stop the model from overshooting the loss function. A smaller learning rate will counteract this problem. However, it can also make the model stuck in a local minima, which is also an issue. This problem can be addressed by having a higher initial learning rate so that the model can overshoot from local minima, land in the valley' of a global minima, and then decreasing the learning rate to meet the loss curve's global minimum point, so obtaining best possible convergence. There must be a proper balance between a higher initial learning rate, and the decreasing factor of the learning rate for the model to converge optimally, and this balance depends from problem to

Model Name	Validation Accuracy(%)
EfficientNetV2B1	99.90
MobileNetV3Large	95.45
DenseNet121	100.0
NASNetMobile	99.60

TABLE IV
ACCURACY SCORES ON DIFFERENT MODELS

problem.

IV. RESULTS AND DISCUSSION

This section will present the details of the experimental results of our research. The entire experiment was conducted in the Google Colab Pro version, using GPU as runtime, which provides an extended amount of GPU usage compared to the regular version of Google Colab.

Among all the models, EfficientNetV2B1 is the latest model, being released in 2021, and DenseNet121 is the oldest model being released in 2017. Table IV shows the validation accuracy scores of the different models after 100 epochs. Since we initialized the models with 'imagenet' weights as a starting point, the models quickly recognized different banknotes from the training set as the training accuracy and loss reached their optimal points right after the second or third epoch, as evident in Fig. 5. DenseNet121 converged the fastest at around 20 epochs, followed by EfficientNetV2B1 and NASNetMobile at around 35 and 65 epochs, respectively. This might be because DenseNet121 is the largest model among all the four. However, MobileNetV3Large still did not seem to converge even after 100 epochs, as the steepness of the validation accuracy curve indicates that it is yet to reach its maximum accuracy, given that it increases at the same rate in subsequent epochs. This might be due to the fact that MobileNetV3Large has the lowest number of layers among the four. Training for more epochs would have increased the validation accuracy of MobileNetV3Large.

On average, the accuracy and loss curves show huge fluctuations from the 1st to the 30th epoch. This indicates that the model was overshooting with the initial learning rate. When the learning rate was reduced afterwards in later epochs, the learning became much 'smoother' as it gradually, but slowly, started to converge, as evident with fewer fluctuations in the graphs later on.

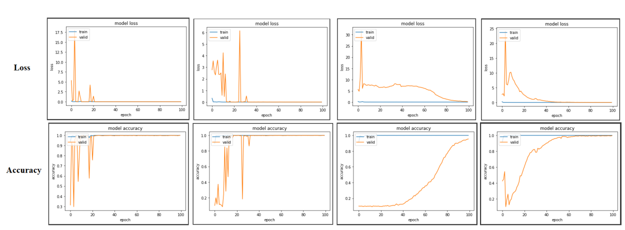

Fig. 5. Visual Comparison of Loss and Accuracy throughout Training

V. CONCLUSION

In this paper, we used four state-of-the-art CNN models to train an existing dataset. All the models used transfer learning to detect and classify Bangladeshi banknotes, which proved to be a vital and efficient approach to solving the research problem. We experiment with Kaggle's "Bangla Money" public dataset, which we improve by augmenting the data and adding an extra class to make it up to date with the current Bangladesh currency denomination.

In the long run, several other potential models that are not yet integrated into the TensorFlow Library can be used to evaluate the dataset. Some of them may even outperform others in terms of accuracy and computational or spatial complexity. Besides this, a real-time prototype for identifying Bangladeshi currency notes could be developed, which would be helpful to vision-impaired people in the future. In order to implement real-time detection of banknotes, models would need to be more heavyweight, which would require intricate and efficient engineering of CNN or other models to make them robust, accurate, and light enough to be able to implement them on IoT and embedded devices for real-world usage. There is also much potential for using the results of this research at ATMs to help visually impaired people and even to detect counterfeit Bangla paper money. Furthermore, extending the dataset can aid in making the model more robust to any potential changes in Bangladesh banknotes in the foreseeable future.

REFERENCES

[1] Rushikesh Jadhav, Swaraj Kalbande, Rohan Katkar, Rohan Katta, Prof. Rakhi Bharadwaj in "Currency Recognition using Machine Learning", Jan 1, 2022 vol 9, issue 1.
[2] "Suyash Mahesh Bahrani" in "Deep Learning approach for Indian currency classification", Oct 2020, Vol. 5, Issue 6, ISSN No. 2455-2143, Pages 335-340
[3] "Uttoran Roy Chowdhury, Susovan Jana, Ranjan Parekh" in "Automated System for Indian Banknote Recognition using Image Processing and Deep Learning"
[4] "Wafia Rarani, Vrushabh Rode, Chaitali Mahatme, Dhiraj Chavhan, Prof. Kalyani Gholap" in "Indian Currency Note Recognition System using YOLO v3 Methodology", Jan 2021, Volume 8, Issue 1.
[5] "Tedele Elsai" in "Developing Ethiopian Paper Currency Recognition Model Using Convolutional Neural Network (CNN)"
[6] "Syed Ejaz Ali, Mriganka Gogoi and Subra Mukherjee" in "Artifical Neural Network To Recognize An Indian Currency Note Using Unique Identification Mark"
[7] Jyothi, C. R., SundaraKrishna, Y. K., SrinivasaRao, V. (2016, March). Paper currency recognition for color images based on Artificial Neural Network. In 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT) (pp. 3402-3406). IEEE.
[8] Linkon, A. H. M., Labib, M. M., Bappy, F. H., Sarker, S., Jannat, M. E., Islam, M. S. (2020, December). Deep Learning Approach Combining Lightweight CNN Architecture with Transfer Learning: An Automatic Approach for the Detection and Recognition of Bangladeshi Banknotes. In 2020 11th International Conference on Electrical and Computer Engineering (ICECE) (pp. 214-217). IEEE.
[9] Tajane, A. U., Patil, J. M., Shahane, A. S., Dhulekar, P. A., Gandhe, S. T., Phade, G. M. (2018, February). Deep learning based indian currency coin recognition. In 2018 International Conference On Advances in Communication and Computing Technology (ICACCT) (pp. 130-134). IEEE.

[10] Sufri, N. A. J., Rahmad, N. A., Ghazali, N. F., Shahar, N., As'Ari, M. A. (2019, August). Vision based system for banknote recognition using different machine learning and deep learning approach. In 2019 IEEE 10th Control and System Graduate Research Colloquium (ICSGRC) (pp. 5-8). IEEE.
[11] Bangla Money. https://www.kaggle.com/datasets/nsojib/bangla-money
[12] Tan, M., Le, Q. (2021, July). Efficientnetv2: Smaller models and faster training. In International Conference on Machine Learning (pp. 10096-10106). PMLR.
[13] Howard, A., Sandler, M., Chu, G., Chen, L. C., Chen, B., Tan, M., ... Adam, H. (2019). Searching for mobilenetv3. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 1314-1324).
[14] Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K. Q. (2017). Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4700-4708).
[15] Zoph, B., Vasudevan, V., Shlens, J., Le, Q. V. (2018). Learning transferable architectures for scalable image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 8697-8710).
[16] Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., Fei-Fei, L. (2009, June). Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248-255). Ieee.
[17] Tan, M., Le, Q. (2019, May). Efficientnet: Rethinking model scaling for convolutional neural networks. In International conference on machine learning (pp. 6105-6114). PMLR.
[18] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L. C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4510-4520).
[19] Tan, M., Chen, B., Pang, R., Vasudevan, V., Sandler, M., Howard, A., Le, Q. V. (2019). Mnasnet: Platform-aware neural architecture search for mobile. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 2820-2828).