Currency Notes Classification and Detection

by

Asikur Rahaman Dipu

1607012

&

Niloy Banik

1607038

Khulna University of Engineering & Technology

Khulna 9203, Bangladesh

March 2022

Currency Notes Classification and Detection

by

Asikur Rahaman Dipu

1607012

&

Niloy Banik

1607038

Supervisor:

Nazia Jahan Khan Chowdhury

Lecturer

Department of Computer Science and Engineering

Khulna University of Engineering & Technology

Signature

A thesis submitted in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering

Khulna University of Engineering & Technology

Khulna 9203, Bangladesh

March 2022

Acknowledgement

The first gratitute is to the almighty Allah who has permitted and blessed us to carry out this study and without His instruction nothing is possible. The physical and mental fitness also helped us to complete the work which is another supreme favor of Allah.

We would like to express our deepest sense of gratitude and heartfelt respect and thanks to our honorable teacher and supervisor Nazia Jahan Khan Chowdhury, Lecturer, Department of Computer Science and Engineering, Khulna University of Engineering & Technology who has been a tremendous mentor to us and his continuous guidance, kind advises, sincere motivations and compassionate care have helped us greatly throughout this thesis work.

We thank all the teachers of the Department of Computer Science and Engineering who helped us providing guidelines to perform the work. We would also like to thank our friends and family for their cordial support.

Authors

Abstract

Currency recognition is an important area of pattern recognition in the modern world. An automatic system which can recognise the currency notes is very important and useful technology of the today's automation systems. Depending on the features of any note, currency recognition can be determined as the features of any currency note differ from one country to another country. For automated detection of currency note,several approaches are already applied using machine learning and image processing techniques. Nowadays deep learning approaches are very popular for detecting,recognizing any kind of image. Because feature and important information extraction is very efficient and accurate in deep learning methods. In this process, machine can analyze a lots of data and learn and identify the important features. For this reason, an automatic system has been proposed to identify and classify Bangladeshi banknotes using a convolutional neural network(CNN) and popular transfer learning models such as MobileNet, VGG16, DenseNet, Xception . A dataset for bangladeshi banknotes has been gathered for this research. Apply different CNN model by changing filters and layers and observe the performance of each model. Using the pretrained transfer learning model, dataset has been trained. Then performance of the models are measured and tested on that dataset. Finally several models are combined and generated to obtain the collective performance. Mainly our model works well for full image of banknote and white background. To test the performance of the model for various banknote, Iriun Webcam is used to use smartphone's camera as webcam for laptop.

Contents

List of Figures

List of Tables

Chapter I

Introduction

1.1 Introduction

Currency is a medium of exchange for goods and services. Simply, it's money, usually in the form of paper or coins which is issued by a government and widely accepted at its face value as a method of payment. Currency is the primary medium of exchange in the modern world, having long ago replaced bartering as a means of trading goods and services. It is a system of money that is widely used,particularly by citizens of a country. Currency is a generally accepted form of payment which is usually issued by a government and circulated within its jurisdiction. The value of any currency fluctuates constantly in relation to other currencies. The currency exchange market exists as a means of profiting from those fluctuations. Before the concept of currency was introduced, goods and services were exchanged for other products and services through the barter system. Money's evolution as a medium of exchange resulted in a much more efficient economy. So, it is very important for blind people to classify and detecting the currency notes properly. An automatic detection and recognition of banknotes can be a very beneficial technology for people with visual impairments, as well as for banks, since it allows for more efficient processing for handling various paper currencies. So, new efficient methods for classification and detection the currency notes are being developed constantly. As different types of currencies need to be classified for different types of computer perspectives such as augmented reality, graphics systems, visual acuity etc. This region has received great attention for a number of applications. Image processing and machine learning techniques can be used to classify and identify currencies. Nowadays convolutional neural network (CNN) is a popular deep learning method which is widely used for any kind of image classification,image recognition,image segmentation , object detection and so on.

1.2 Motivation

Deep learning is a branch of machine learning that focuses on iterative learning approaches that expose machines to large data sets.When dealing with unstructured data, deep learning's capacity to process vast amounts of features makes it incredibly strong as it has the ability to process large volumes of data efficiently[1][2]. So, main motive is to apply deep learning techniques to learn how to classify different types of the currency note. For this, different types of currency note images are collected to make an image dataset. An image classification system is developed to detect various currency images and find the currency value of a note. Already there exists many currency recognition and detection techniques. But most of those technologies are expensive and costly and frequently wrong. In the era of machine learning and deep learning,this issue can be easily overcomed with various deep learning techniques and methods .

1.3 Problem Statement

One of the most essential aspects of human history is money. Unfortunately, Blind persons are unable to differentiate banknotes. They face a lot of problems to recognise banknotes for day-to-day transaction. According to the experts, more than 750,000 people suffer from blindness in Bangladesh[3]. So, automatic detection technology can be very effective for the visually impaired people [4]. If there exists any automated way to help them recognising banknotes for their day-to-day transaction, visually impaired peoples will get rid of recognising and detecing banknote problem. So the main purpose of the currency recognition system is to recognize the currency note by applying and utilizing different techniques and methods on the dataset of currency notes. In this research, an automated system for identifying and recognising the Bangladeshi banknotes using a convolutional neural network and transfer learning methods have been proposed for helping the people with visual difficulties.

1.4 Objective

The main purpose of this research is to propose a technique to classify and recognise bangladeshi banknotes into 9 classes which includes two taka, five taka, ten taka, twenty taka, fifty taka, one hundred taka, two hundred taka, five hundred taka and one thousand taka. The main objective of our thesis is summarized as follows:

- A dataset of bangladeshi banknotes is collected from internet. A new dataset for two hundred taka is created and added with the previously downloaded dataset. Finally, there are nine classes in the final dataset, with a total of 5400 images of bangladeshi banknotes.
- In our proposed method, different sequential convolutional neural network models are applied. Then popular CNN-based pretrained transfer learning architectures like MobileNet, Xception, VGG16 and DenseNet are applied on our dataset. Finally we deeply observe the performance of those models result.
- In this thesis, mobile phone's camera is being used as wabcam using Iriun Webcam. Then, the model is tested for those capturing the images of different banknotes.

1.5 Organization of this Thesis

The remaining part of this thesis consists of the parts as follows:

Chapter II will briefly discuss about related work.
Chapter III will explain the proposed method.
Chapter IVwill present the detailed results regarding the proposed method.
Chapter V contains conclusion, achievement of works.

Chapter II

Literature Review

2.1 Introduction

In computer vision, image classification is a key study field to solve any real-world complex problem. Various image classification techniques are previously developed to solve those complex and real world problems. In recent year, researchers have been focusing on the Deep Learning (DL) which is a branch of Machine Learning (ML). This deep learning has a great impact on various research fields and it is now widely used . In our Thesis, we try to develop a method using convolutional neural network (CNN) based models and different transfer learning approaches for bangladeshi banknotes classification.

2.2 Previous Works

Adiba Zarin et al.[5] developed a hybrid model using optical character recognition(OCR), face recognition and hough features to detect fake banknotes. Using macro lenses, micro printing images are obtained. To extract micro printed letters from that images, OCR is applied. Bangladeshi banknotes contain the face of Bangabandhu Sheikh Mujibur Rahman,so some faces of Bangabandhu Sheikh Mujibur Rahman are used as training set and face recognition algorithm is applied to recognise that feature. In real bangladeshi currency notes,there exists ultraviolet lines. To extract that line feature, hough transform is used. For getting better accuracy, all those model are combined.

Jesmin Akter et al.[6] developed a Currency Recognition System using Supervised Learning. Here, total 185 images of different bangladeshi banknotes are used as training set and total 74 images are used as testing set. To apply filtering, each image is split into the red, green, and blue channels. Then the RGB image is recombined by combining those three channels. The RGB image is used to extract several characteristics such as HSV, edge, and grey-level co-occurrence matrix. By saving the total number of features,the Euclidean distance of features is calculated between the input image and the template images. The output is provided via the shortest distance. Finally, the proposed system's performance has been evaluated using a challenging dataset containing a wide range of conditions, including old, unclean, dirty and ripped banknotes.

Rahnuma Tasnim et al.[7] developed a Currency Recognition System in Real-time using Convolutional Neural Network for Visually Impaired People. First, from the banknote's 50 samples, videos of both sides of a banknote are captured in various lighting conditions. All banknotes must be on a flat surface and fully visible in the frame while the videos are being recorded. The videos are organized into eight folders based on eight different types of banknotes. Finally, a new collection with over 70,000 images of currently available Bangladeshi banknotes has been generated from those videos to use as training set for the proposed system. To train those images, a CNN based model has been designed. Here, Adam optimizer is used for optimization and Categorical cross-entropy is used to determine the loss and categorical accuracy is used as the targeted performance parameter for evaluating the model's efficiency after each iteration. A real-time currency detector has been developed using CNN based model.

Hasan Murad et al.[8] also proposed a Bangla Currency Recognizer for visually Impaired People. Here, they have developed a camera-based automatic currency recognizer for the visually impaired people in Bangladesh. Almost 8000 images are used as training set. To train and classify the banknotes,the deep learning architecture MobileNet is used. In their proposed architecture, there are two distinct parts. The initial step is to automatically extract the feature and the next process is the classification of various banknotes. Finally, they designed a mobile android application to verify the effectiveness and feasibility of the proposed model and evaluated and validated the application with the users from a blind community.

Ali Hasan et al. [9] is proposed Deep Learning Approach Combining Lightweight CNN Architecture with Transfer Learning to recognise bangladehi banknotes. They have created their dataset by collecting bangladeshi banknote images from Bangla Money dataset which consists of 1970 images and Bangla Currency dataset which consists of 8000 images. As a base model, deep learning methods based on Lightweight Convolutional Neural Network architectures combining with transfer learning ResNet152v2, MobileNet and NASNetMobile have been used. Various augmentations, hyperparameter tuning, and optimizations techniques are used to increase the accuracy of the model. According to their proposed solution, lightweight models like MobileNet and NasNetMobile have achieved better accuracy than the heavyweight model ResNet152v2.

Md. Ferdousur Rahman Sarker et al. [10] proposed a a real-time Bangladeshi currency detection system using image processing algorithms and ORB based algorithm to help visually impaired people to recognize banknotes more easily in different viewpoints and scales. This recognition system is also able to recognise the banknotes those are rumpled, decrepit or even worn. Image preprocessing, image analysis and image recognition techniques are included in this recognition system. After creating dataset, input images are converted into gray-scaled images. Because gray-scale images reduce the inappropriate results due to variation of lighting. For detecting the keypoints of the images, ORB keypoint detection technique is used. Using ORB Descriptor Extractor the descriptors are extracted. Then they store all the detected descriptor of input image to use those features in image Recognition part.

Md. Ekram Hossain utilized et al. [7] and rebuilt the Siamese Neural Network architecture for One-shot learning to detect Bangladeshi bank notes using a little dataset. They only looked at 20 photos for five distinct Bangladeshi bank notes that individuals used on a daily basis for this research. This study will aid the general public in having a better experience with vending machines that can detect notes based on a single data example. They employed 5 notes (5,10,20,50,100 TAKA) and applied convolutional architecture to achieve a great result with 97.38 percent accuracy.

U. Bhattacharya et al. [11] offer a unique and successful approach for extracting Devanagari and Bangla texts from camera acquired scene photos based on analysis of related components. The existence of headline is a common property of these two scripts, and the suggested technique extracts them using mathematical morphological procedures. They also discuss a few criteria for robust text component filtering from such scene photos. They also looked into the issue of binarization of such scene images and discovered that in some cases, repeated binarization using a well-known global thresholding approach is effective.

Rahnuma Tasnim et al. [12] has developed and tested a real-time Bangladeshi currency detector. For this goal, a unique Bangladeshi banknote dataset was created. A unique Bangladeshi banknote dataset was created for this purpose. A CNN model based on deep learning was created, and the model achieved 92 percent accuracy in real-time. A unique

Bangladeshi banknote dataset was produced for this purpose. A CNN model based on deep learning was created, and the model achieved 92 percent accuracy in real-time. At 180 degrees, the system is rotationally invariant. This system also requires less complicated technology than existing banknote detecting systems and can identify currency in real time.

Nadim Jahangir and Ahsan Raja Chowdhury [5] have devised a recognition technique for Bangladeshi banknotes based on neural networks. The technique can be easily implemented on low-cost hardware, which might be highly valuable in a variety of situations. The recognition system uses scanned pictures of banknotes that are scanned using low-cost optoelectronic sensors and then put into a Multilayer Perceptron that has been trained using the Back-propagation technique. In the preprocessing step, Axis Symmetric Masks are employed to minimize network size and ensure correct detection even if the note is reversed. The experimental findings demonstrate that this technique can effectively recognize the currently available 8 notes (1, 2, 5, 10, 20, 50, 100, and 500 Taka) with an average accuracy of 98.57 percent.

2.3 Discussion

After analysing all the existing banknotes classification and detection techniques, we found that some of them bulid a system using traditional image processing technique or machine learning algorithm. In some cases, there exists some limitation in dataset. Finally, we have developed a more efficient method and technique for recognition and detection of bangladeshi banknotes using convolutional neural network and transfer learing methods.

Chapter III

Methodology

3.1 Introduction

Machine learning (ML) [12] is considered to be a part of artificial intelligence. It is the study of computer algorithms that can learn and develop on their own with experience and the use of data. Different algorithms in machine learning create a model based on training data to make predictions or decisions without having to be explicitly programmed to do so. Deep learning is a machine learning subfield that employs algorithms based on the structure and function of the brain's neural network. It consists with three or more layers neural network. These neural networks aim to simulate the activity of the human brain by allowing it to learn from enormous amounts of data. While a single-layer neural network may produce approximate predictions, additional hidden layers can help to optimize and improve for accuracy. For image classification task, a lot of algorithms and procedures are available. The fundamental goal of image recognition is to extract hidden pattern and characterize them in order to generate some important information. Nowadays deep learning is very popular for this task. The detailed methodology for bangladeshi banknotes classification and recognition is developed in this section. Firstly a CNN model is created to perform this recognition and classification task. Then a model is also created using SVM with CNN. Various transfer learning architectures are used to train the dataset and detect the notes. Finally an ensemble of different models is tested to analyze the prediction and performance of the models.

3.2 Basic Definitions

In this section, we explain some necessary terminology with their definition.

3.2.1 Optical Character Recognition(OCR)

The use of technology to detect printed or handwritten text characters inside digital pictures of physical documents, such as scanned paper documents, is known as OCR (optical character recognition). OCR is a technology that examines a document's text and converts the characters into code that may be used to process data. Text recognition is another name for OCR[11]. OCR systems are hardware and software systems that turn physical documents into machine-readable text. The text is copied or read using hardware such as an optical scanner or dedicated circuit board, while additional processing is usually handled by software. Artificial intelligence (AI) may also be used in software to develop more complex intelligent character recognition (ICR) approaches, such as recognizing languages or handwriting styles[13][14]. The most typical application of OCR is to convert physical copies of legal or historical documents into PDFs[15]. Users may edit, format, and search the document as if it were generated in a word processor once it has been saved as a soft copy.

3.2.2 Neural Network

Deep learning and neural networks [16] are important issues in computer science and the technology. They offer the best answers to a variety of challenges in image recognition, audio recognition, and natural language processing. Artificial neurons that receive and process data make up a neural network. Here data is passed through the input layer, the hidden layer, and the output layer. When data is given into a neural network, it begins to process it. After then, the data is processed through its levels to get the desired result.

A neural network is made up of vertically stacked components called Layers. There are three types of layers in neural networks:

i) The input layer is the first layer in neural network. The data will be accepted by this layer, which will then send it to the remainder of the network.

ii) The hidden layer is the second type of layer. A neural network's hidden layers can be one or more in number. The hidden layers are the ones that are actually responsible for neural networks' high performance and complexity. They may do a variety of tasks at once, such as data transformation, automatic feature development, and so on.

iii) The output layer is the final type of layer. The outcome or result of the problem is stored in the output layer. The input layer receives raw images, and the output layer receives the output.

Neurons in a neural network dominate one another. Neurons make up each layer. The data received by the input layer is redirected to the hidden layer. Here, weights are allocated to each input. In a neural network, the weight is a value that converts input data within the network's hidden layers. In this case, Weight values are multiplied with input data. Then the value for the first hidden layer is generated. The inputs and weights are multiplied, and the result is delivered to the hidden layer neurons as a sum. Each neuron is given a bias. To calculate the sum, each neuron adds the inputs it receives. After that, the value passes via the activation function. The result of the activation function determines whether or not a neuron is activated. When a neuron is active, it sends information to the other layers. The data is created in the network till the neuron reaches the output layer using this method. The input data is transformed by the hidden layers and passed to the other layer. The output layer is responsible for generating the intended result.

3.2.3 Convolutional Neural Network

A Convolutional Neural Network (CNN) is a deep learning network architecture that learns directly from data, removing the requirement for manual feature extraction. It can take an input image, assign importance (learnable weights and biases) to various aspects or objects in the image, and distinguish between them. When compared to other classification methods, the amount of pre-processing required by a CNN model is significantly less. While basic approaches require hand-engineering of filters, CNN model can learn these filters or characteristics with enough training. CNN is a multilayer network which has the ability to perform state-of-the-art in image classification, image recognition, image segmentation, object detection and many other tasks. CNNs are specially useful for recognizing objects, faces and scenes by finding patterns in imagess.

Convolutional Layer

Convolution is an orderly technique that combines two sources of information. It is an operation that transforms one function into something else. Convolutions have been used in image processing for a long time to blur and sharpen images, but they can also be employed to do other tasks. CNNs enforce a local connection pattern between neurons of neighbouring layers. A Convolutional Neural Network's first layer is usually a Convolutional Layer. Convolutional layers perform a convolution on the input before forwarding the output to the next layer. Here the pixels in a convolution's receptive area are all converted to a single value.

For any CNN model, convolution operation is very important. Here, kernel operations are applied and this technique is commonly referred to as the sliding window approach. Kernels can be any size such as 3*3, 5*5 etc. In this process, kernel is moved across the entire image, multiplied by the intensity values, and then summed to replace the coordinate position in this procedure.

Pooling

Pooling is a method used in convolutional neural networks to enable the network detect features regardless of their position in the picture by generalizing information retrieved by convolutional filters. The basic building elements of a convolutional neural network used for computer vision applications like image identification are convolutional layers. A convolutional layer applies a filter to the picture and extracts features, resulting in a feature map that may be sent to a higher-level convolutional layer to extract higher-level features. As a result, CNNs can detect more complex structures and objects in an image by stacking many convolutional layers. Pooling is commonly done on the feature map created by a previous convolutional layer and a non-linear activation function in a convolutional neural network. Pooling is extremely similar to the convolution process in terms of its underlying approach. You choose a filter and drag it over the output feature map of the convolutional layer before it. The most typical filter size is 22, which is slid across the input with a stride of two. The pooling filter calculates an output on the receptive field based on the type of pooling operation you've chosen (the part of the feature map under the filter). Pooling may be done in a variety of ways. Max-pooling and average pooling are the two most prevalent ways.

- **Max-Pooling** : Pooling is a characteristic that many Convolutional Neural Network (CNN) designs use. A pooling layer's main goal is to "accumulate" characteristics from maps created by applying a filter to a picture. Its formal purpose is to gradually shrink the spatial dimension of the representation in order to minimize the number of parameters and computations in the network. Max pooling is the most prevalent type of pooling. Max pooling is used to aid over-fitting by giving an abstracted representation of the data. It also lowers the computational cost by minimizing the number of parameters that must be learned and gives basic translation invariance to the internal representation. By applying a max filter to (typically) non-overlapping subregions of the original representation, max pooling can be achieved.

- **Average-Pooling** :The layer selects the average values of the elements accessible in the patch of the feature map when using average pooling. Essentially, the entire feature map is downsampled to the average value obtained by the feature map's area. As a result, the max-pooling feature of every patch is revealed, whereas the average pooling feature reveals the average of the covered area. The picture below shows an example of average pooling of a 4*4 image using a 2*2 feature map in the pooling layer. Pooling gives us some amount of translation invariance. Also, pooling is faster to compute than convolutions. When we use average pooling it helps to extract the smooth features. If talking about image data if we apply the average pooling layers we will get them out as the combination of all colors presented in the region covered by the feature map. So if the distribution of the data points in data and colors in any image is smooth or more basically the distribution is proper then we can use the Average pooling to get proper results.

Fully Connected Layer

In a neural network, fully connected layers are those in which all of the inputs from one layer are connected to each activation unit of the following layer. The last few layers of most common machine learning models are fully connected layers that assemble the data retrieved by earlier layers to generate the final output. After the Convolution Layer, it is the layer that takes the most time.The last convolution or pooling layer's output feature maps are often flattened, or converted into a one-dimensional array of integers, and connected to one or more fully connected layers, also known as dense layers, in which each input is coupled to each output by a learnable weight. The features collected by the convolution layers and downsampled by the pooling layers are transferred to the network's final outputs, such as the probabilities for each class in classification tasks, by a subset of fully connected layers. In most cases, the number of output nodes in the final fully linked layer equals the number of classes. A nonlinear function, such as ReLU, is applied to each completely linked layer.

3.2.4 Hyper Parameter

A mathematical model containing a number of parameters that must be learned from data is referred to as a Machine Learning model. We can fit the model parameters by training a model using existing data. On the other hand, hyperparameters are a type of parameter that cannot be learned directly from the standard training procedure. Hyperparameters are the variables which determines the network structure and the variables which determine how the network is trained and their's value are set before training(before optimizing the weights and bias). Generally, a hyperparameter's value is used to control the learning process.

i) Number of Hidden Layers and units : The layers between the input and output layers are known as hidden layers. It's quite straightforward. Continue to add layers until the test error no longer improves. With regularization techniques, many hidden units inside a layer can increase accuracy. Underfitting may occur if the number of hidden units is reduced.

ii) Dropout : At the time of training a model, the network can be over trained and thus over fit occurs. Dropout is a regularization approach that helps to reduce overfitting (increasing validation accuracy) probability during the training time. It refers to the removal of units from a neural network, both hidden and visible. In this approach, some nodes are avoided for some epoch operation, while other nodes are trained. Avoided nodes are trained after the next epoch . As a result, each of the nodes has much more efficiently trained. In this way,the performance of any model can be improved.

iii) Learning Rate : The learning rate is the rate at which a network's parameters are updated. It determines how frequently the weight in the optimization method is updated. The learning process is slowed by a low learning rate, but it eventually converges. A faster learning rate accelerates the learning process, but it may not converge. A declining Learning rate is usually desired. Depending on the optimizer we employ (SGD, Adam, RMSProp etc), we can utilize a constant learning rate, a progressively decreasing learning rate, momentum-based approaches, or adaptive learning rates.

iv) Number of epochs : The number of epochs refers to how many times the full training data is run through the neural network. During training, the number of epochs is the number of times the entire training data is displayed to the network. The number of epochs should be increased until the difference between the test and training error is negligible.

v) Batch size : Batch size refers to the number of samples passed through the neural network throughout the training. If all of the data is passed at once, performance measures may be affected. So, in the learning phase of convolutional neural network, mini-batch is frequently preferred. The number of sub samples sent to the neural network is known as the mini batch size.

vi) Weight initialization : To avoid dead neurons, the weights of the each layer should be initialized with small random integers. At the time of initializing the random integers, we must be careful about zero gradient. In weight initialization, uniform distribution is mostly used.

3.2.5 Activation Function

The output of an input or a group of inputs is defined by the activation function. In other words, the activation function defines the node of the output of a node that is supplied in inputs. To produce the desired result, they simply determine whether to stimulate or deactivate neurons. It also applies a nonlinear adjustment to the input to improve the performance of a complicated neural network. The activation function may also be used to normalize the output of any input with a value in the range of 1 to -1. Because the neural network is trained on millions of data points, the activation function must be efficient and should decrease computation time. In every neural network, the activation function determines whether a particular input or receiving information is important or not.Here are some activation functions:

i) Sigmoid Function: Its non-linearity is its major benefit over other steps and linear functions. The function has a S shape and ranges from 0 to 1. In certain publications, it's also known as the logistic or squashing function. The sigmoid function is utilized for probability-based output in the DNN's output layers.Sharp damp gradients during backpropagation, gradient saturation, sluggish convergence, and non-zero centered output are some of its key shortcomings, allowing gradient updates to propagate in multiple directions.

ii) TanH Function: The hyperbolic tangent function has a range of -1 to 1 and is zero-centered. Because this function is zero centered, it's simple to simulate inputs with significantly negative, neutral, or positive values. If your output is not between 0 and 1, you should use the tanh function instead of the sigmoid function. Tanh functions are typically utilized in RNN for natural language processing and speech recognition.

iii) Rectified Linear Unit(ReLU): With state-of-the-art results, ReLU has been the most extensively employed activation function for DL applications. When compared to the Sigmoid and Tanh activation functions, it outperforms them in terms of performance and generalization. ReLU enables quicker calculation since it does not compute exponentials or divisions, in addition to the total speed of computation being improved. When compared to the sigmoid function, it readily overfits, which is one of the key drawbacks. Overfitting is reduced using techniques such as dropout.

iv) Softmax Function : For any kind of multi-class classification problem, final layer of any convolutional neural network must be softmax activation function . The output of the softmax activation function's is in the range between 0 to 1. As a result ,total summation of all the output class will be always 1.

v) Softplus Function: Dugas proposed Softplus in 2001, which is defined by the equation f(x)=log (1+ex). Softplus provides smoothing and nonzero gradient features, which helps to improve the stability and performance of DNNs that use soft plus units. When the Softplus function was compared to the ReLU and Sigmoid functions, the Softplus function performed better with fewer epochs to convergence during training.

3.2.6 Optimizer

Optimizers are techniques or approaches that adjust the characteristics of your neural network, such as weights and learning rate, to decrease losses.Optimization algorithms or methods are in charge of lowering losses and delivering the most accurate outcomes. Some types of optimizers are given below:

i) Gradient Descent: The simplest basic yet widely used optimization approach is Gradient Descent. In linear regression and classification techniques, it is widely utilized. The Gradient Descent technique is also used in backpropagation in neural networks.Gradient Descent is a first-order optimization approach that is based on the loss function's first order derivative. It determines which way the weights should be changed in order for the function to approach a minimum. The loss is passed from one layer to the next through backpropagation, and the model's parameters, also called as weights, are updated based on the losses in order to minimize the loss.

ii) Stochastic Gradient Descent(SGD): It tries to update the parameters of the model more often. After each training example's loss has been computed, the model parameters are changed. As a result, if the dataset has 1000 rows, SGD will update the model parameters 1000 times in one dataset cycle, rather than once as with Gradient Descent.

iii) Mini-Batch Gradient Descent(MBGD): It is the most effective of all gradient descent techniques. It's a step up from SGD and conventional gradient descent. After each batch, the model parameters are updated. As a result, the dataset is separated into batches, and the parameters are updated after each batch.

iv) Momentum: Momentum was created to ease the convergence and reduce the excessive volatility in the SGD. It decreases the volatility in the irrelevant direction while speeding up the convergence in the right direction.

v) Adam: Adam (Adaptive Moment Estimation) is a momentum expert who works with both first and second order momentums. The Adam's intuition is that we don't want to roll too rapidly only to go over the minimum; instead, we want to slow down a little to allow for a more thorough search.

3.2.7 Loss Function

The difference between the predicted output and the outcome delivered by the machine learning model is quantified by the loss function in a neural network. We can obtain the gradients that are utilized to update the weights from the loss function. The cost is calculated as the average of all losses.It's a way to see how effectively our algorithm models our dataset. Our loss function will provide a greater value if our forecasts are completely wrong. It'll give you a lower number if they're fairly decent. Our loss function will inform us if we're making progress when we tweak parts of our algorithm to try to enhance our model.A few of the most popular loss functions currently being used are given below:

i) Mean squared error (MSE):The workhorse of fundamental loss functions is mean squared error; it's simple to comprehend and apply, and it typically works well. Take the difference between your predictions and the ground truth, square it, then average it over the whole dataset to get MSE.

ii) Likelihood loss: The likelihood function is also a straightforward function that is frequently employed in classification difficulties. The function multiplies the expected probabilities for each input sample. The output is valuable for comparing models, even if it isn't quite human-interpretable.

iii) Log loss (cross entropy loss): One of the most prominent measurements for Kaggle tournaments is log loss, which is also used often in classification issues. It's simply a logarithmic version of the likelihood function.

3.2.8 Data Augmentation

Data augmentation is a collection of strategies for producing additional data points from current data in order to artificially enhance the quantity of data available. Making modest adjustments to data or utilizing deep learning models to produce additional data points are examples of this.By creating fresh and varied instances to train datasets, data augmentation can help enhance the performance and results of machine learning models. A machine learning model works better and more correctly when the dataset is rich and sufficient. Data collection and labeling may be time-consuming and costly for machine learning models. Companies can lower these operating expenses by transforming datasets using data augmentation techniques. Cleaning data is one of the phases in creating a data model, and it is required for high accuracy models. However, if data cleaning lowers representability, the model will be unable to make accurate predictions for real-world inputs.

Advanced data augmentation models are now available:

Adversarial training/Adversarial machine learning:It produces adversarial instances that disturb a machine learning model and injects them into a dataset to train.

GANs (generative adversarial networks): GAN algorithms can learn patterns from input datasets and generate new instances that look like training data automatically.

Neural style transfer: Neural style transfer models may mix content and style images while also separating style from content.

Reinforcement learning: In a virtual world, reinforcement learning models teach software agents how to achieve their objectives and make judgments.

3.3 Main procedures of our research

All the steps of our working procedure are briefly discussed below:

3.3.1 Feature extraction using OCR

All the availble currency notes in our country contain some bangla and english text and characters. At first, our main idea is to extract those text . So we study on the extraction of texts from images. Optical character recognition(OCR) is used for this task. Normally feature extraction works well for gray scale image. So, we convert the image into gray scale image. Then we apply different thresholding techniques like binary thresholding , adaptive thresholding , ostu's thresholding on the banknote image. On the resulting image, we apply different morphological operation such as erosion, dilation, opening, closing etc[11][15]. Finally we apply OCR using pytesseract on that preprocessed image. We spend a lot of time in this procedure. Unfortunately, we are unable to extract text and characters from our dataset. This procedure only works for the clear images.

In the first procedure, we convert our image in gray scale image. Then adaptive thresholding and adaptive thresholding with gauss techniques are applied . From that resulted image, noise are removed . In the noise free image,some erosion and dilation are applied . Finally , orc is used using pytesseract on that final preprocessed image . Input and Ouput for our first procedure is given in figure 3.1:

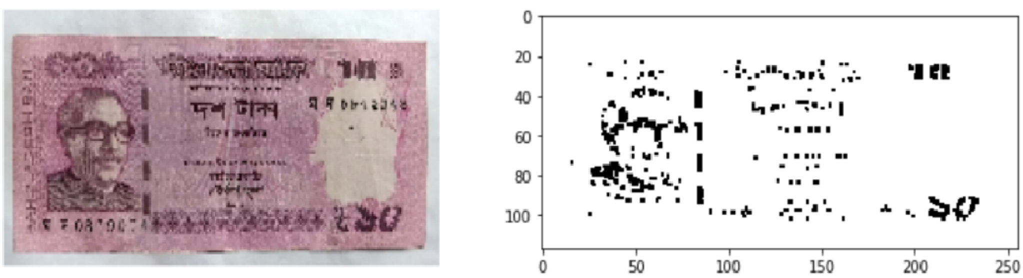

Figure 3.1: (a) Input (b) Output for procedure 1

In the second procedure, we convert our image in gray scale image. Then bilateral filter is applied on the gray scale image. To get the egdes of the text and characters of the images,we applied canny edge detection algorithm . Then we try to find the contours of the texts and characters . After finding the contours, we try to draw boxes around the contours . Again orc is used using pytesseract on that final preprocessed image . Input and Ouput for our second procedure is given in figure 3.2:

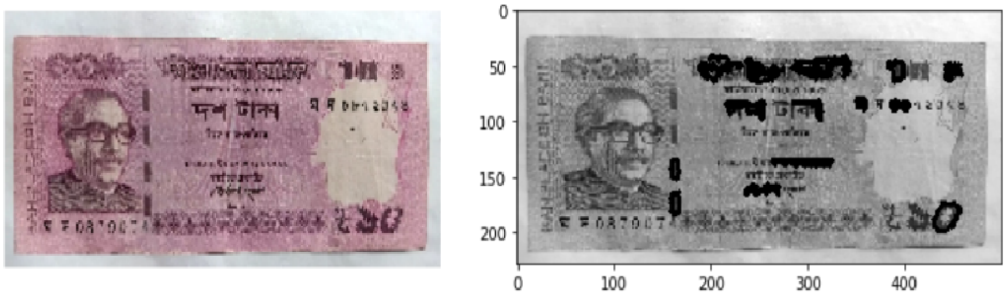

Figure 3.2: (a)Input (b)Output for procedure 2

3.3.2 Preprocessing Steps

Our next experiment is to apply deep learning approaches on our dataset. In deep learning, machine can easily analyze the images and then enable to recognise the important features which can identify the images. Nowadays for image recognition, convolution neural network is widely used. Before applying any CNN model,our main task is to process our dataset.

- **Image preprocessing** : In image preprocessing step,various type of preprocessing tasks such as removing noise from images,crop the images from the unwanted background,resize the images into a particular shape etc are performed . We gather most of our dataset from internet source and only the dataset for 200taka is collected by capturing differnt notes of 200 taka via smartphone. So there exists variation in the quality and resolution of images in our dataset . In this step, we resize all the images into 224*224 .
- **Data Augmentation** : After the preprocessing, data augmentation is very much useful for image classification. It is a process of creating new data from the existing training data. Mainly this is only applied on the training set. This process is not applicable for validation and test set. Different types of techniques such as horizontal and vertical shift, horizontal and vertical flip,random rotation,random brightness, random zoom etc are the parts of data augmentation. It plays an vital role on the performance of

the model. In our dataset, there are 4320 images for training. In the training set, we apply rescaling=(1/255),share range =0.2,zoom range = 0.2, fill model = 'nearest' and horizontal flip on our dataset using ImageDataGenerator from keras library.

A sample of training set after applying image preprocessing and data augmentation is shown in figure 3.3.

Figure 3.3: Sample training set after preprocessing and augmentation

3.3.3 Model Training Steps

- **Sequential CNN Model1** : In sequential CNN Model1, we use three convolutional layer along with three max-pooling layer. In the first convolutional layer our given input size is 224*224. In this layer, we use kernel size = 3*3, filters =64, strides = 2 and ReLU as activation function. In the first max-pooling layer, pool size is 2*2 and strides = 2. Rest of the convolutional layer,we use kernel size as 3*3, filters =32, strides = 2 and ReLU as activation function. And rest of the max pooling layer, we use pool size as 2*2. After the convolutional and max pooling layer, we apply dropout technique. In our case, dropout is 0.5. Finally we apply two dense layer. In the first dense layer, dense unit is 128 and ReLU activation function is used. In the final dense layer, dense unit is 9 as our total number of classes for banknotes classification are nine. With that dense unit, Softmax activation function is used. The main architecture of our proposed model1 is given in figure 3.4:

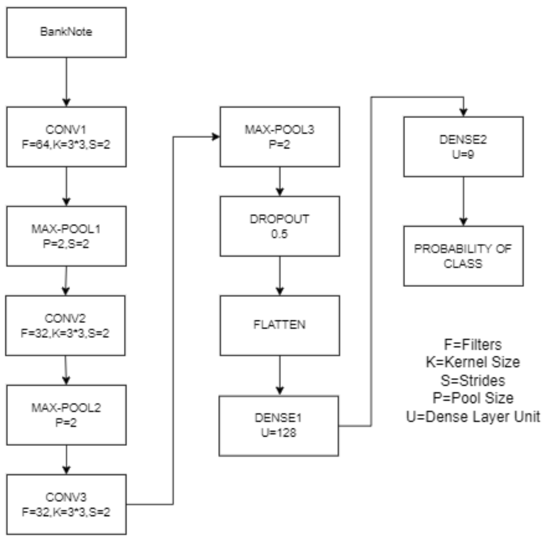

Figure 3.4: Flow Chart of our proposed model1

The training accuracy graph and loss training graph of our proposed model1 is given figure 3.5:

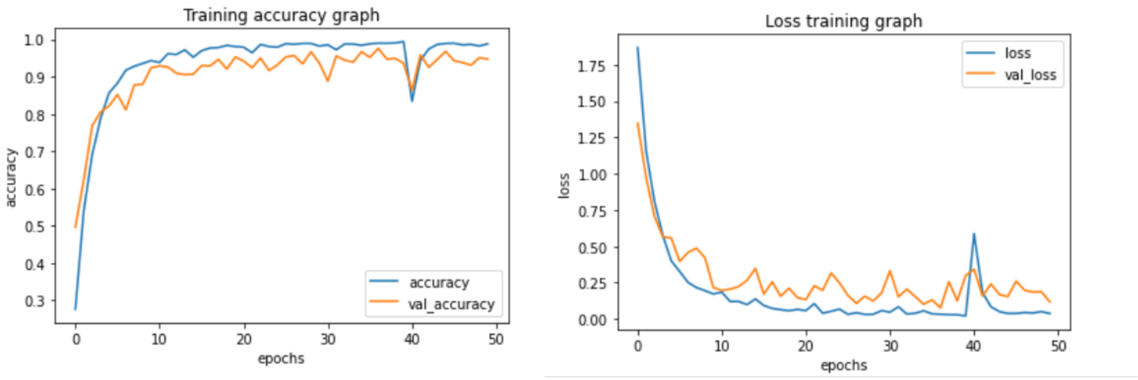

Figure 3.5: (a)Training accuracy (b)Loss Training graph for Model1

- **SVM with Sequential CNN** : We also try to combine SVM with our convolutional neural network model. For this, we import regularizers from keras library. In the final dense layer of our proposed CNN model, we add a regularizer as 0.01. Finally we train the model on our dataset. The training accuracy graph and loss training graph when we apply SVM regularization with our proposed model is given figure 3.6:

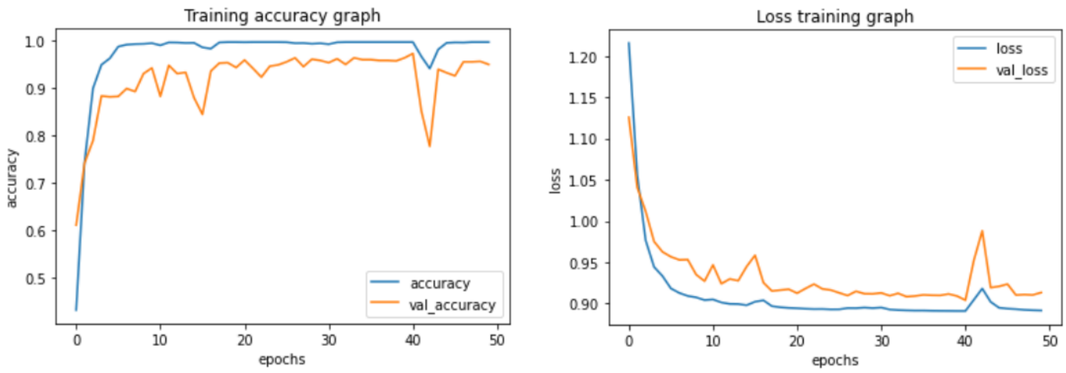

Figure 3.6: (a)Training accuracy (b)Loss Training graph for SVM with Model1

- **Sequential CNN Model2** : In convolutional neural network, any kind of combination of convolutional layer, pooling layer,dense layer is acceptable. But it doesn't mean that we can use any random convolutional neural network model. We have to accept any model which provides better accuracy and acts well on the dataset. So,we perfer a model which performs well in our banknote dataset for our research. So, we choose another sequential model to compare the new model with the previous one. In sequential CNN Model2, we use six convolutional layer along with six max-pooling layer. In the first convolutional layer our given input size is 224*224. In this layer, we use kernel size = 3*3, filters =256 and ReLU as activation function. In the first max-pooling layer, pool size is 2*2. Rest of the convolutional layer Kernel size and activation function is same. Here only difference is in filters. In those convolutional layer, we use filters as 256, 128, 64, 64, 32, 32. And rest of the max pooling layer, we use pool size as 2*2. After the convolutional and max pooling layer, we apply dropout technique. In our case, dropout is 0.3. Finally we apply two dense layer. In the first dense layer, dense unit is 128 and ReLU activation function is used. In the final dense layer, dense unit is 9 as our total number of classes for banknotes classification are nine. With that dense unit, Softmax activation function is used. Actually we try different combination. In each experiment, we change the number of convolutional layer and max-pooling layer. We also apply different size of kernels,filters and strides. After deep observation on some cnn model, we fix our second model called model2. This same process is also applicable for the model1. The main architecture of our proposed model2 is given figure 3.7.

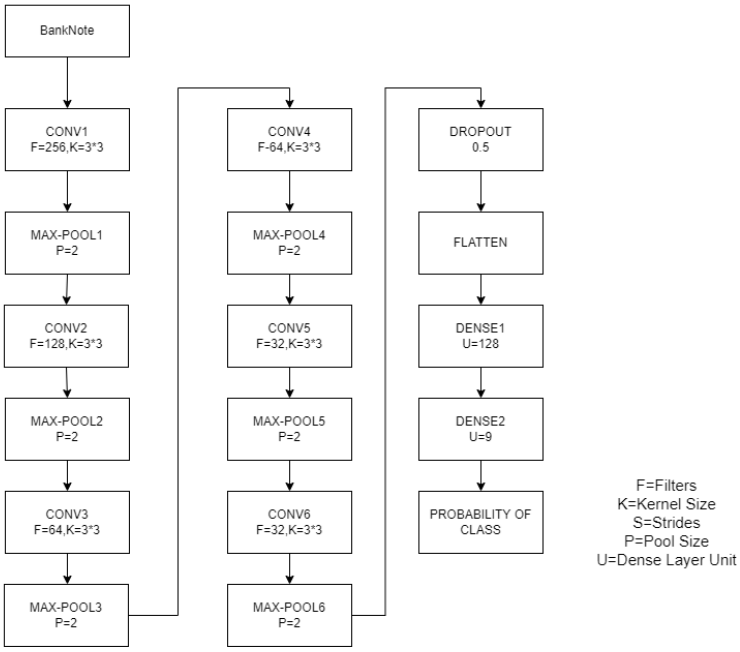

Figure 3.7: Flow Chart of our proposed model2

The training accuracy graph and loss training graph of our proposed model2 is given figure 3.8 .

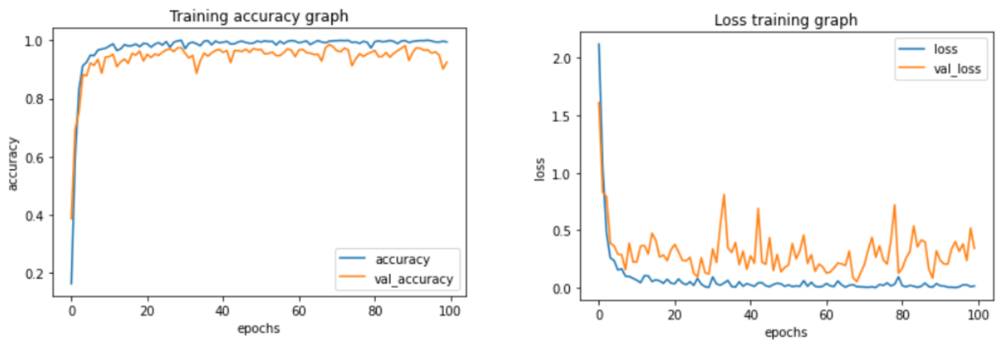

Figure 3.8: (a)Training accuracy (b)Loss Training graph for Model2

- **Transfer learning** : In the case of transfer learning , we choose MobileNet,DenseNet, VGG16 and Xception pretrained model . We apply those model on our dataset and then observe the performance of those model on our currency dataset .
- **VGG16** :VGG16 is a widely used Convolutional Neural Network (CNN) Architecture that was developed for ImageNet, a massive visual database project used in the development of visual object identification software. VGG16 is a common deep learning

image classification approach that is employed in many deep learning image classification algorithms owing to its simplicity of usage. Because of the benefits it provides, VGG16 is often utilized in learning applications.The ImageNet Large Scale Visual Recognition Challenge was won using VGG16, a CNN architecture (ILSVRC). It is still considered to be one of the greatest vision architectures ever created.

- **DenseNet** : DenseNet is a novel neural network for visual object recognition that was discovered recently. DenseNet and ResNet are quite similar, however there are a few key distinctions. DenseNet concatenates (.) the output of the previous layer with the output of the future layer, whereas ResNet utilizes an additive approach (+) that combines the previous layer (identity) with the future layer. DenseNet was created to combat the disappearing gradient in high-level neural networks, which causes accuracy to deteriorate. To put it another way, the information evaporates before it reaches its destination owing to the longer journey between the input and output layers. DenseNets use a straightforward connection topology in which each layer of the network is directly linked to every other layer. Yes, you read that correctly. Each layer takes inputs from all preceding levels and passes on its own feature-maps to all following layers to maintain the feed-forward nature.

- **Xception** : Xception is a depthwise separable convolutional neural network design. Inception modules in convolutional neural networks are described by Google as a step between normal convolution and the depthwise separable convolution process (a depthwise convolution followed by a pointwise convolution). In this view, a depthwise separable convolution may be thought of as an Inception module with the most towers possible. This discovery leads them to propose a new deep convolutional neural network architecture based on Inception, but using depthwise separable convolutions instead of Inception modules.The data initially passes via the entering flow, then eight times through the middle flow, and lastly through the exit flow. Batch normalization is applied to all Convolution and SeparableConvolution layers.In most traditional classification tests, the Xception architecture outperformed VGG-16, ResNet, and Inception V3. XCeption is a cost-effective design based on two key principles: (i)Convolution that can be separated in depth. (ii)Convolution block shortcuts, similar to ResNet.

- **MobileNet** : MobileNet is a CNN-based image classification and mobile vision architectural paradigm. There are alternative models, but what makes MobileNet unique is that it requires very little computational resources to execute or apply transfer learning. This makes it ideal for mobile devices, embedded systems, and PCs with limited computing efficiency or no GPU, without affecting the accuracy of the findings. It's also optimal for web browsers, which have limitations in terms of compute, graphics processing, and storage.

  MobileNet Architecture:

  • MobileNets, a simplified architecture that leverages depthwise separable convolutions to generate light weight deep neural networks, is introduced for mobile and embedded vision applications.

  • We present two simple global hyper-parameters for optimally balancing latency and accuracy.

  Depthwise Separable Convolution is the basic layer of MobileNet. Another thing that might help with performance is network structure. Finally, the width and resolution may be adjusted to optimize the latency/accuracy tradeoff.

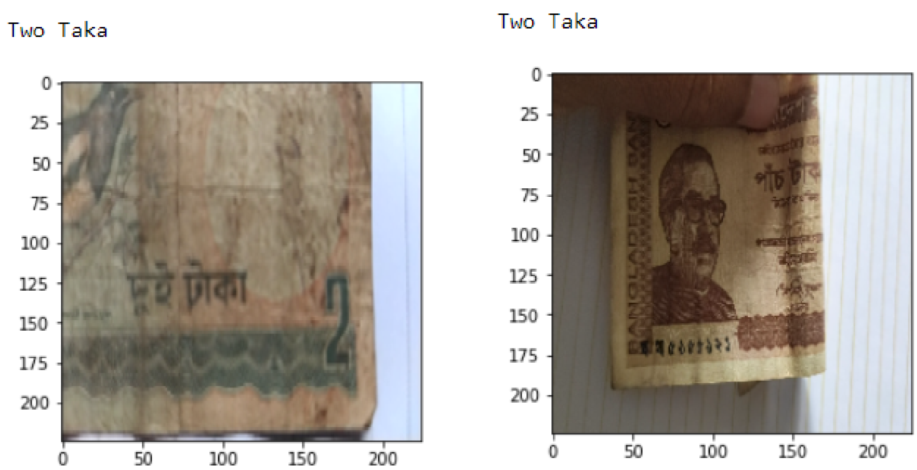

Figure 3.9: (a)Correct Prediction (b)Wrong prediction

- **Ensembling** : Ensemble is a technique of combining multiple models in order to obtain the collective performance . This ensembling procedure is more reliable and accurate compared with the general machine learning and deep learning approaches . So, we combine our proposed model with the model of transfer learning and observe the performance of combined model .

3.3.4 Model Testing Step

To test our model, we randomly provide image from the test dataset. Then we observe the performance of the model and check how well our model works. Moreover, we can capture image of banknotes using Iriun Webcam in our mobile phone. This webcam must be installed in both smartphone and laptop or pc. By this Iriun webcam, we can use our phone's camera as webcam in laptop or pc. Then we load the address of the captured image in the model and test whether our model is able to predict and recognise the banmkmote correctly or not. By this way,the performance of the model is measured. A demo of model's prediction is given in figure 3.9.

Chapter IV

Experimental Result

4.1 Introduction

For bangladeshi banknotes, there does not exist any standard dataset. Only few resources are available. We collect most of the images in the dataset from internet source. Only few images are created by ourselves. In this chapter, the details of creating the project's required datasets and the setup for hardware and software will be briefly discussed. Moreover, we also discuss about the model performance with different standard evaluation metrics like precision, recall, accuracy etc. Evaluation techniques will also be discussed in this chapter. Finally, experimental results have been presented.

4.2 Dataset collection

We collect our dataset from different internet source. But, dataset for 200 taka is not presented as 200 taka note publishes after 2020. So, we generate dataset for 200 taka by collecting some note of 200 taka and capturing images using smartphone camera.

4.3 Experimental Setup

Experimental setup includes different hardware and software which are related to the work. Sometimes, it is called as installation process or environment setup. Proper installation process must be required for any kind of programming related work.

4.3.1 Hardware Requirements

For any deep learning task, the configuration of the hardware plays an important role on the performance of the model. Any model can be trained on both CPU and GPU. For small amount of dataset, training on CPU or GPU does not make any difference. For a large amount of dataset, we can realize the importance of high configuration CPU or GPU. If we train our model in CPU, it requires a lot of time. Comparatively, training on GPU requires less time than training on CPU. We train our CNN and pretrained MobileNet model in CPU.

The configuration of our laptop is Intel Core(TM) i5-8250U CPU, RAM 8GB and windows 11 operating system. We train pretrained VGG16, DenseNet model for our dataset in google colab. To perform our proposed task efficiently and fastly, high configuration CPU and GPU must be required.

4.3.2 Software Requirements

For any programming related work, different kinds of software might be required. In the research, models are developed using Python 3.6.13. Here, we install anaconda navigator and then use jupyter notebook as code editor. We also use google colab as code editor for fast training. We choose python as our base platform. Tensorflow is an open source platform and keras is high level user friendly library as it is build in python. Both of those library are popular and efficient and easy for any machine learning and deep learning task. So, we implement our model using tensorflow and keras library. Moreover, various python libraries such as Numpy, Pandas, Matplotlib etc are used in the implementation of our work.

4.3.3 Dataset

For the classification and detection of bangladeshi banknotes, dataset of notes is very important. The majority of the images in the dataset are obtained from an internet source. We only make a few images by using smartphones. Our final dataset contains 5400 images divided into nine classes. Each of the class contains 600 images. Our targeted classes are currently available currency notes: two taka, five taka, ten taka, twenty taka, fifty taka, one hundred taka, two hundred taka, five hundred taka, one thousand taka. We collect our images from both internet sources and capturing via smartphones, so the resolution of the images may vary. To make the resolution same for all images, all the images are resized during the image preprocessing step. An image of each banknote class for our dataset is given in figure 4.1.

4.3.4 Training and Testing data

Training set is the collection of data used to train the model and teach it how to find hidden features and patterns in the data. The training set should include a diverse collection of inputs so that the model can predict any future data sample. The validation set is a set of data that is used to test the performance of our model during training.The main idea of dividing the

Figure 4.1: A sample of Banknote Dataset

dataset into a validation set is to avoid overfitting. After every epoch, the model is trained on the training set and evaluated on the validation set. On the otherhand , test set is a distinct set of data that is used to test the model once it has been trained. Actually when any model is trained, testing data confirms the accuracy of the correct prediction of the model . In our work , among 5400 images, 4320 images are used as training set and 1080 images are used as validation set . Moreover, we use 575 images as testing set .

4.3.5 Confusion Matrix

Confusion matrix is a standard technique to summarize the performance of a classifier. Just only depending on the accuracy, we can't measure the performance of the model. By summarizing and calculating the confusion matrix, we are able to understand model's right prediction and the error of model. For this reason, we have to learn and understand some inportant term first.

i) True Positive : When model's prediction and actual class both are true. ii) False Positive : When model's prediction is true but actual class is false. iii) False Negative : When model's prediction is flase but actual class is true. iv)False Positive : When model's prediction and actual class both are false.

Accuracy: This parameter shows the percentage of correctly predicted banknotes out of all the predicted banknotes in the dataset. It measures the performance of model across all classes. The following equation is used to calculate accuracy:

$$Accuracy = \frac{True\ Positive\ +\ True\ Negative}{True\ Positive\ +\ True\ Negative\ +\ False\ Positive\ +\ False\ Negative} * 100\% \tag{4.1}$$

Precision: This parameter shows the percentage of correctly predicted banknotes out of all the predicted banknotes in the dataset.The following equation is used to calculate precision:

$$Precision = \frac{True\ Positive}{True\ Positive\ +\ False\ Positive} * 100\% \tag{4.2}$$

Recall: This parameter measures the percentage of correctly identified Positive banknote classes to the total number of actual Positive banknote classes. The ability of the model to recognise actual positive classes is measured by the recall. The following equation is used to calculate recall:

$$Recall = \frac{True\ Positive}{True\ Positive\ +\ False\ Negative} * 100\% \tag{4.3}$$

F1-Score: This parameter is the combination of precision and recall . It measures model's accuracy on a banknote dataset . The following equation is used to calculate accuracy:

$$F1Score = \frac{2\ *\ Precision\ *\ Recall}{Precision\ +\ Recall} * 100\% \tag{4.4}$$

4.4 Evaluation of banknote classification

4.4.1 CNN Model1

For any model's performance is measured and evaluated on the result of accuracy, precision, recall, f1-score. For our proposed CNN model, values of accuracy, precision, recall, f1-score for our targeted classes are given in table 4.1.

Confusion matrix of our proposed CNN model is given in table 4.2:

Class	*Precision*%	*Recall*%	F1-Score
2TK	100	100	1.00
5TK	95	98	0.97
10Tk	100	100	1.00
20TK	100	100	1.00
50TK	98	95	0.97
100Tk	100	100	1.00
200TK	100	100	1.00
500TK	100	100	1.00
1000Tk	100	100	1.00

Table 4.1: Precision,Recall,F1-Score Table for Model1

Class	2TK	5TK	10Tk	20TK	50TK	100Tk	200TK	500TK	1000Tk
2TK	72	0	0	0	0	0	0	0	0
5TK	0	59	0	0	0	0	0	0	0
10Tk	0	0	60	0	0	0	0	0	0
20TK	0	0	0	60	0	0	0	0	0
50TK	0	0	0	0	57	0	0	0	0
100Tk	0	0	0	0	0	72	0	0	0
200TK	0	0	0	0	0	0	47	0	0
500TK	0	0	0	0	0	0	0	72	0
1000Tk	0	0	0	0	0	0	0	0	72

Table 4.2: Confusion Matrix for Model1

4.4.2 CNN model2

After the training of model1, we also try another cnn model called model2 by changing the value of filters and dropout, increasing convolutional and max-pooling layer. We train our new model2 and then try to compare with the model1 by checking confusion matrix, precision, recall, f1-score. For this CNN model, values of precision, recall, f1-score for our targeted classes are given in table 4.3:

Confusion matrix of our proposed CNN model is given in table 4.4:

4.4.3 Transfer Learning Model

We apply different transfer learning models such as MobileNet,VGG16,DenseNet,Xception on our dataset. The accuracy of those given pretrained model are given in table 4.5:

From the table , we see that VGG16 model performs well on our dataset as test accuracy is 98.955% which is higher than other applied transfer learning method .

Class	*Precision*$_{\%}$	*Recall*$_{\%}$	F1-Score
2TK	100	100	1.00
5TK	92	100	0.96
10Tk	100	100	1.00
20TK	100	100	1.00
50TK	100	92	0.96
100Tk	100	100	1.00
200TK	100	100	1.00
500TK	100	100	1.00
1000Tk	100	100	1.00

Table 4.3: Precision,Recall,F1-Score Table for Model2

Class	2TK	5TK	10Tk	20TK	50TK	100Tk	200TK	500TK	1000Tk
2TK	72	0	0	0	0	0	0	0	0
5TK	0	60	0	0	0	0	0	0	0
10Tk	0	0	60	0	0	0	0	0	0
20TK	0	0	0	60	0	0	0	0	0
50TK	0	5	0	0	55	0	0	0	0
100Tk	0	0	0	0	0	72	0	0	0
200TK	0	0	0	0	0	0	47	0	0
500TK	0	0	0	0	0	0	0	72	0
1000Tk	0	0	0	0	0	0	0	0	72

Table 4.4: Confusion Matrix for Model2

4.4.4 Comparative Analysis

After the training of our both model, we campare between them . We also compare our model with predefined transfer learning model . Finally we come to the conclusion that our model performs well on given dataset. Moreover this process is only applicable for white background and full image of the datanote .

Model	$Accuracy_{\%}$	Total Parameters	Trainable Parameters
VGG16	98.955	14,981,961	267,273
Xception	94.78	20,879,921	18,441
DenseNet	98.80	7,566,921	529,417
MobileNet	94.087	3,758,281	529,417
MobileNetV2	88.52	2,918,473	660,489

Table 4.5: Result of different Transfer learning model

Chapter V

Conclusion

5.1 Summary

In the era of machine and deep learning,any kind of automated technology will be beneficial for human being. So we try to build a technology which will be helpful for people. We focus on visually impaired people. Any kind of currency note detection and recognition is a major problem for blind people. At first, we gather dataset of bangladeshi bankotes to establish a system which will recognise different kinds of bangladeshi banknotes. Then we apply convolutional neural network model on our dataset. We also check the performance of various transfer learning models such as VGG16, MobileNet, DenseNet, Xception on our dataset. Finally, we test our model by capturing the images of banknotes via smartphones using Iriun Webcam.

5.2 Future Scopes of Work

- We only train our model on 5400 images. In future we will try to enrich our dataset including folding, half images of banknote.
- Our model doesn't work well in complex background. It works well for white background. So, we will try to build a model which can detect and recognise banknotes any kind of complex background. For this, image segmentation can be a better approach.
- Our model is implemented on jupyter notebook and google colab and tested the performance via Iriun webcam. To make this project user friendly, android or Ios based software can be built. In future, we'll modify our model with web-based or mobile based currency recognizer application.
- We are able to recognize and classify single banknote at a time. In future, we will try to build a model which can detect multiple notes at a time and count total amount.

References

[1] K. Yoshida, M. Kamruzzaman, F. A. Jewel, and R. F. Sajal, “Design and implementation of a machine vision based but low cost stand alone system for real time counterfeit bangladeshi bank notes detection,” in *2007 10th international conference on computer and information technology*. IEEE, 2007, pp. 1–5.

[2] N. Jahangir and A. R. Chowdhury, “Bangladeshi banknote recognition by neural network with axis symmetrical masks,” in *2007 10th international conference on computer and information technology*. IEEE, 2007, pp. 1–5.

[3] A. H. M. Linkon, M. M. Labib, F. H. Bappy, S. Sarker, M.-E. Jannat, and M. S. Islam, “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)*. IEEE, 2020, pp. 214–217.

[4] M. E. Hossain, A. Islam, and M. S. Islam, “A proficient model to classify bangladeshi bank notes for automatic vending machine using a tıny dataset with one-shot learning & siamese networks,” in *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. IEEE, 2020, pp. 1–4.

[5] A. Zarin and J. Uddin, “A hybrid fake banknote detection model using ocr, face recognition and hough features,” in *2019 Cybersecurity and Cyberforensics Conference (CCC)*. IEEE, 2019, pp. 91–95.

[6] J. Akter, M. K. Hossen, and M. S. A. Chowdhury, “Bangladeshi paper currency recognition system using supervised learning,” in *2018 International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2)*. IEEE, 2018, pp. 1–4.

[7] R. Tasnim, S. T. Pritha, A. Das, and A. Dey, “Bangladeshi banknote recognition in real-time using convolutional neural network for visually impaired people,” in *2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST)*. IEEE, 2021, pp. 388–393.

[8] H. Murad, N. I. Tripto, and M. E. Ali, "Developing a bangla currency recognizer for visually impaired people," in *Proceedings of the Tenth International Conference on Information and Communication Technologies and Development*, 2019, pp. 1–5.

[9] A. H. M. Linkon, M. M. Labib, F. H. Bappy, S. Sarker, M.-E. Jannat, and M. S. Islam, "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)*. IEEE, 2020, pp. 214–217.

[10] M. F. R. Sarker, M. I. M. Raju, A. Al Marouf, R. Hafiz, S. A. Hossain, and M. H. K. Protik, "Real-time bangladeshi currency detection system for visually impaired person," in *2019 International Conference on Bangla Speech and Language Processing (ICBSLP)*. IEEE, 2019, pp. 1–4.

[11] M. Z. Islam and A. K. Mondal, "Towards a standard bangla photoocr: Text detection and localization," in *2014 17th International Conference on Computer and Information Technology (ICCIT)*. IEEE, 2014, pp. 198–203.

[12] Q. Zhang and W. Q. Yan, "Currency detection and recognition based on deep learning," in *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 2018, pp. 1–6.

[13] R. Sarkar, N. Das, S. Basu, M. Kundu, M. Nasipuri, and D. K. Basu, "Cmaterdb1: a database of unconstrained handwritten bangla and bangla–english mixed script document image," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 15, no. 1, pp. 71–83, 2012.

[14] J. Liang, D. Doermann, and H. Li, "Camera-based analysis of text and documents: a survey," *International Journal of Document Analysis and Recognition (IJDAR)*, vol. 7, no. 2, pp. 84–104, 2005.

[15] U. Bhattacharya, S. K. Parui, and S. Mondal, "Devanagari and bangla text extraction from natural scene images," in *2009 10th International Conference on Document Analysis and Recognition*. IEEE, 2009, pp. 171–175.

[16] S. Mittal and S. Mittal, “Indian banknote recognition using convolutional neural network,” in *2018 3rd International Conference On Internet of Things: Smart Innovation and Usages (IoT-SIU)*. IEEE, 2018, pp. 1–6.