**IDENTIFYING CURRENCY DENOMINATION IN REALTIME**

***Synopsis***

***for***

**Major Project**

**Class XII**

**(Session 2020-21)**

***Submitted by***

**ARYAN KHATANA & DEEPTANSHU PRASAD**

***Under the Guidance***

***of***

**Ms. V.K. Remya**



**BLUEBELLS SCHOOL INTERNATIONAL**

***Kailash Colony, New Delhi***

|  |
| --- |
| **INTRODUCTION OF PROJECT/PROBLEM STATEMENT**  *In today’s world there are a lot of dishonest people who can do anything for money – steal, rob, fool, anyone. They don’t even show any compassion towards people with physical imparity. So, we have created rudimentary system that can help the visually impaired by identifying currency denomination while paying for something using a camera and machine learning.*  *In this project we have trained a simple Convolutional Neural Network to identify currency denomination in real-time using a camera feed.* |
|  |
| **OBJECTIVE OF THE PROJECT**  *Whenever a visually-impaired person goes to a store to shop for something and they have to pay using cash there is a possibility they can be robbed by the store owner or someone else. Through this project we hope to decrease the number of visually impaired people who are robbed in the name of helping them. Through this app they can easily use their phones camera and using the machine learning model the app would speak out the denomination of the currency.* |
|  |
|  |
| **FEASIBILITY STUDY**  *The app has no computational barriers as the model is pretrained and all it needs is a camera and speaker. In the background the image is cropped and passed through the model to get an output.* |
|  |
| **TOOLS & TECHNOLOGY USED**  *While creating this project the following tools and modules were used:*   * *TensorFlow* * *OpenCV* * *NumPy* * *Python* * *Keras* |
|  |
| **MODULE DESCRIPTION**   * ***TensorFlow:***   *It is a free open-source machine learning/deep learning framework created by Google that makes the implementation of models easy. We used this framework to implement and train our Convolutional Neural Network and save its weights for use in the app.*   * ***OpenCV:***   *It is a free library aimed at real-time Computer Vision task. It takes the camera feed and applies all the necessary transformations to make it ready to be able to pass through the model for the output.*   * ***NumPy:***   *It is a library that is used to perform all the mathematical computation including all high-level functions which are prebuilt and can work with higher dimensional vectors.*   * ***Python:***   *Python is a high-level programming language that is the base language used in the whole project.*   * ***Keras:***   *It is a high-level framework and a part of TensorFlow that is used for creating the machine learning model.*   * ***Convolutional Neural Network:***   *It is a type of neural network than takes images as input and spits out their label as output.*  *First, we pass in an image on which different filters are applied to get a sense of low-level features such as lines and curves. Then that is passed through a series of linear layers with activation functions so that the model can map non-linearities. Then we get a prediction which is tallied with the original label and a loss is calculated then the model started to learn using gradient descent and reduce the loss so that the prediction is as close to the actual label.* |
|  |
|  |
| **FUTURE SCOPE OF THE PROJECT**  *Right now, the project is working on a computer using the computer cameras and a rudimentary model without any hyperparameter tuning or data augmentation but later it can be made into a full-fledged android and IOS application with added functionalities such as counting the money and then speaking it out loud and counting coins etc. It can also use a database so improve the model as more and more users use it. And different techniques such as transfer learning, hyperparameter tuning, Cross validation, data augmentation can be used to improve the model’s accuracy even further.* |
|  |
| **REFERENCE OR BIBLIOGRAPHY**   * ***Introduction to Neural Networks***   [*https://www.youtube.com/playlist?list=PLZHQObOWTQDNU6R1\_67000Dx\_ZCJB-3pi*](https://www.youtube.com/playlist?list=PLZHQObOWTQDNU6R1_67000Dx_ZCJB-3pi)   * ***Understanding of a convolutional neural network***   [*https://ieeexplore.ieee.org/document/8308186*](https://ieeexplore.ieee.org/document/8308186)   * ***How to Visualize Filters and Feature Maps in Convolutional Neural Networks***   [*https://machinelearningmastery.com/how-to-visualize-filters-and-feature-maps-in-convolutional-neural-networks/*](https://machinelearningmastery.com/how-to-visualize-filters-and-feature-maps-in-convolutional-neural-networks/\)   * ***Visualizing Convolutional Neural Nets***   [*https://www.cs.ryerson.ca/~aharley/vis/conv/*](https://www.cs.ryerson.ca/~aharley/vis/conv/)   * ***TensorFlow Documentation***   [*https://www.tensorflow.org/api\_docs/python/tf/all\_symbols*](https://www.tensorflow.org/api_docs/python/tf/all_symbols)   * ***Keras Getting Started***   [*https://keras.io/getting\_started/*](https://keras.io/getting_started/)   * ***Multi-Label Classification***   [*https://towardsdatascience.com/multi-label-classification-and-class-activation-map-on-fashion-mnist-1454f09f5925*](https://towardsdatascience.com/multi-label-classification-and-class-activation-map-on-fashion-mnist-1454f09f5925) |
|  |