Currency Denomination Identification using Mask-RCNN

Major Project Report

Submitted in fulfillment of the requirements

for the degree of

Master of Technology

in

Computer Science Engineering

(Networking Technology)

By

Tejaskumar Bajania

(18MCEN17)

Computer Science Engineering Department

Institute of Technology

Nirma University

Ahmedabad-382 481

December 2019

Contents

List of Figures

Chapter 1

Introduction

1.1 Currency Identification

There are many approaches used for detecting currencies without using a computer vision. For visually impaired persons it is detected using marks that are present on the notes. There are special diagram marks present on the currency to identify every denomination individually. Diagrams marks are like a circle, triangle, square which are useful in identifying currency by physically sensing it. Also the persons that are not visually impaired have to individually identify and count each of the denominations in-order to count the amount.

There are various approaches that are used to identify currencies by visually impaired persons. That are:

- Using different tools that use the size of notes to identify currency denomination. These scales are referred to as money identifier cards.Figures [1.1,1.2,1.3] show the scales that are used for identify currency.
- Blind people can fold a denomination of money in a way, which helps in recognizing the money.
- Also there are several money identifier tools that are used for only detecting the currencies one by one.

Figure 1.1: Currency identification is done by visually impaired persons. It is the scale that uses the currency size to identify the currency.

Figure 1.2: Using the scale to identify the denomination of fifty rupee.

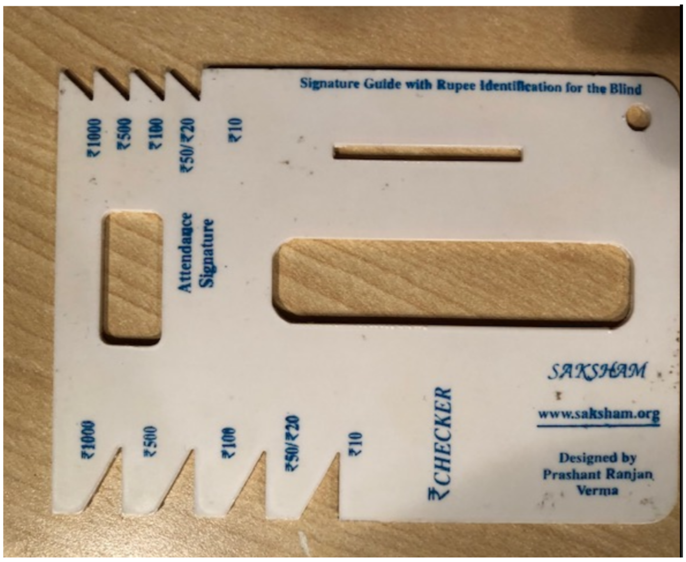

Figure 1.3: Money identifier card.

The problem arises when there are multiple currencies present in an image, using computer vision each of the currency denominations can be identified. So the objective is to identify each denomination present in an image. For identification there are computer vision approaches that can be used are discussed in section 1.4.

1.2 Why Computer Vision

There is a need of a prototype that can detect a currency without physically sensing the banknotes and it can be able to count the notes also. The idea is to identify the currency even if there are no marks on the currency paper. It can be done with the help of image recognition. It can be used to interact with the physical world with the digital world. So for identifying currencies computer vision can help the physical world (i.e., currency) interact with the digital world (i.e., currency image).

For detecting and identifying the currencies with the help of physical sensing also identifying each denomination with the help of automation, for this purpose computer vision is required.

1.3 About Computer Vision

It is the interpretation of an image or adding some meaning to an image through a computer system. Here we give input as an image and get an output as meaning related to that image. In our approach the input will be an image with currency present in that image and output will be the mask over the entered image.Computer vision is a versatile field, which is used to gain an understanding from images and videos. It aims to work like human vision system[14].

1.4 Computer Vision Techniques

Some of the major computer vision techniques are:

- Image Classification
- Object Detection
- Semantic Segmentation
- Instance Segmentation

1.4.1 Image Classification

It is a classification for images. Given a set of images, where we try to predict the class of a new set of images. Here the input is given as an image and the meaning related to that image is in the form of a class. For example: if an image contains images of two classes, let A and B be two classes. If an input image contains an unknown (from A or B) class and we predict that it belongs to class B or not. This categorization is the classification of image. The figure 1.4 shows image classification. By identifying unique balloons in an image.

1.4.2 Object Detection

It is the task of defining objects around images. It involves classification and localization to single object [16]. For instance if an image contains a unique object and we try

Figure 1.4: Image classification. Which detects that if an image contains balloons or not. [1]

to detect the presence of it in that image. Creating bounding boxes around that object can be a way of detection. The figure 1.5 shows object detection, here bounding boxes are used to detect balloons in an image.

1.4.3 Semantic Segmentation

It is a concept that attempts to comprehend job of every pixel in the image [16]. It works on classifying each object in that image. It takes an input image and gives dense pixel-wise predictions on that image. The group of balloons that are segmented in figure 1.6

1.4.4 Instance Segmentation

Instance Segmentation is the task of recognizing and portraying each unique object of interest appearing in an image. It is identifying of objects at a pixel level, that is we are using the pixel-wise masks for each object. For example, multiple objects are present in an image at certain locations. It will identify which pixel belongs to each object. It is the combination of object detection and semantic segmentation. In figure 1.7 we can see each balloons have their own class.

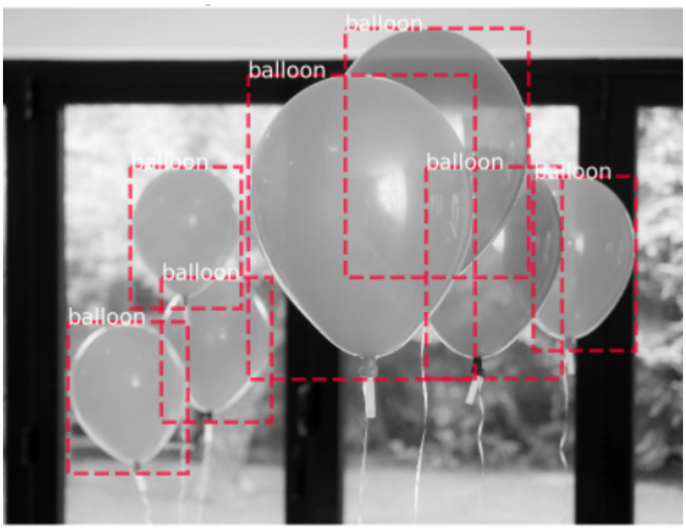

Figure 1.5: Object detection on an image. Which detects all the balloons that are present in an image and creates bounding boxes around it.
[1]

Figure 1.6: Semantic segmentation on an image. Here balloons are detected and differentiate it from the background.
[1]

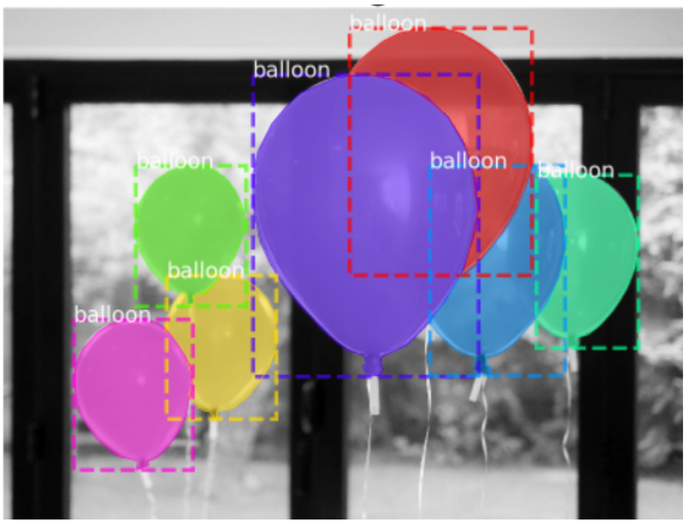

Figure 1.7: Instance segmentation on an image. Here all the balloons are detected and differentiated by each of the other balloons.

[1]

1.5 Motivation

As per the problem we need an architecture that helps in distinguishing the number of groups (i.e., denominations) preset in a picture and give the audio from that picture.

We need to locate and detect number of denominations in an image. For each image there should be a mask created over it, which can be done using MRCNN architecture for mask generation and object detection. So the main objective is to identify the number of denominations present in an image and output that image in audio format.

The approach used here can be explained by figure 1.8 Firstly, it takes input as an image with Indian currencies present in it. Then it Mask-RCNN model is used for detection, in figure it Instance Segmentation. The output masked image is then used by Google text to Speech(gTTS) for giving the output in audio format.

The motivation behind detecting the individual currencies is that presently [19, 18, 23] the approaches that are used to detect and identify currency which is on a single image. Our approach is to identify each of the individual currencies that are present in the image. Further it can give the count of the number of notes that are present in an

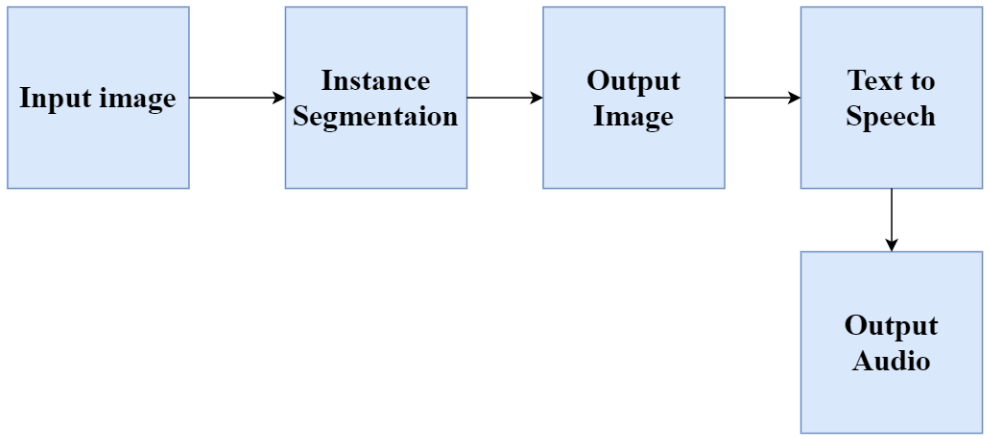

Figure 1.8: Proposed approach.

image.

The purpose of using instance segmentation is because it can help detect the unique currencies also even if they are present in dense form. Classifying connected objects is difficult, so instance segmentation is needed.

As a use case the collected output is helpful for visually impaired persons, when the output is converted into a speech format which is the part of proposed approach.

Chapter 2

Literature Survey

2.1 Work Related to Currency Identification

2.1.1 A Neural Network-Based Model for Paper Currency Recognition and Verification(1996)

Initially most of the work was related to verification and recognition. The work uses budget-friendly optoelectronic gadgets which produce a sign related with the light refracted by the paper currencies[5].

2.1.2 A Currency Recognition System Using Negatively Correlated Neural Networks Ensemble(2009)

Also there is some work related to negative correlation learning to ensemble patterns among the currencies. It creates an ensemble network for the currency recognition system [24]. Also for classifying different types of currencies.

2.1.3 Survey Of Currency Recognition System Using Image Processing(2013)

There is some work done before for currency recognition which used image processing for currency recognition. Which acquires images, pre-processes it, extracts the required features, and gives the result[25]. Those are the techniques used for currency recognition.

2.1.4 Real-Time Recognition of Series Seven New Zealand Banknotes(2018)

Considering the visually impaired person's real-time identification of banknotes is also done using color feature extraction [29], it uses neural networks for classification. Also recognition results are considered for an indoor environment.

2.1.5 MANI App

The same work is done that was for object detection by Reserve Bank of India. It is a MANI app. It detects the currency that is captured in camera and speech to text output is given. It is for visually impaired persons. Which is available at [20]. It is Mobile Aided Note Identifier.

2.2 Model Preparation

2.2.1 ImageNet classification with Deep Convolutional Neural Networks(2012)

The model trains a large, convolutional neural network to categorize images with high-resolution images. It also gives an improvement in image classification accuracy[15]..

The basic approach for using a model which can help generate the masks and for object detection blocks. We have gone through Restnet101 architecture.

2.2.2 Deep Residual Learning for Image Recognition(2015)

The paper presents a residual learning framework to simplify the training of networks that are more profound than previously used ones. It also evaluates the proposed Resnet architecture with VGG Nets on ImageNet datasets [11]. Figure 2.1 shows the Residual Building Block:

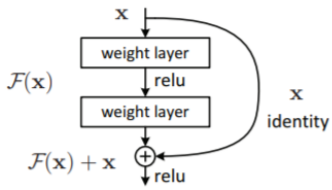

Figure 2.1: Residual building block.

2.2.3 Fast-RCNN(2015)

This paper proposes a Fast R-CNN builds based on previous work(i.e. RCNN).Here training a classifier and bounding box regressor is parallel working. It has more detection accuracy compared to previous. It is implemented in Python and C++ [6].

2.2.4 Region-based Convolutional Networks for Accurate Object Detection and Segmentation(2015)

It is the implementation using Convolutional Neural Networks. Which uses classifier and bounding box regressor in a sequential manner. The paper combines the region proposals with Convolutional Neural Networks, the resulting model is called Region-based CNN. It measures the PASCAL VOC challenge datasets. [7]. It is discussed later in chapter 3.

2.2.5 Identifying Mappings in Deep Residual Networks(2016)

The paper examines the extensions formulation of Residual building blocks. From the below Figure 2.2 it shows the usage of skip connections and after addition activation. It improves the transfer of training: [12].

2.2.6 Faster-RCNN:Towards Real-Time Object Detection with Region Proposal Networks(2016)

This paper introduces Region Proposal Networks (RPN) that shares complete convolutional features from the image to the detection network. RPN simultaneously predicts object bounds and objectness scores that are present in an image for each locations[22].

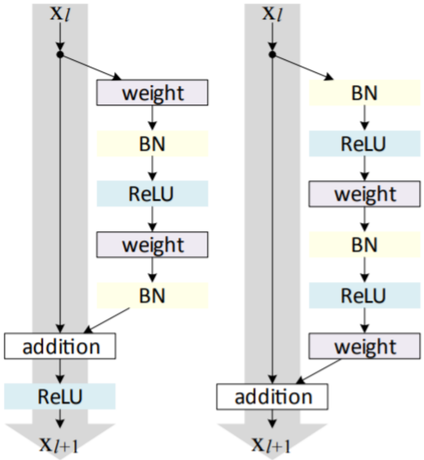

Figure 2.2: Original Residual Unit and Proposed Residual Unit.

2.2.7 Feature Pyramid Networks for Object Detection(2017)

The paper discusses about the architecture of Feature Pyramid Networks (FPN). It uses it in Faster-RCNN and exceeds the results of existing models [17].

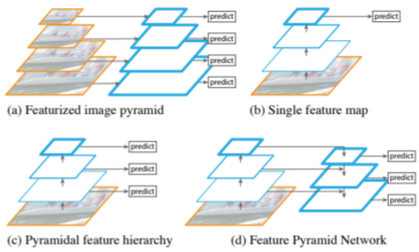

Figure 2.3: Feature pyramid networks.

2.2.8 Mask-RCNN(2017)

The paper presents the method of Mask R-CNN which detects objects and parallelly generates masks for each of the instance that is present in an image. Also it covers various experiments of object detection[13].

The approach represented in this paper satisfies objective for instance segmentation. It can be useful to detect and generate masks over the detected region.

2.3 Instance Segmentation Approaches

2.3.1 Deep Mask(2015)

Some work proposes another way for generating object proposals, based on the discriminative convolutional network [21]. It first outputs a segmentation mask, after it gives the likelihood patch. It is referred to as Deep Mask.

For preparing a model architecture, there were approaches for instance segmentation. Which can also be used for satisfying our objective.

2.3.2 Hybrid Task Cascade for Instance Segmentation(2019)

There is a lot of work-related to image instance segmentation. One of which is Hybrid Task Cascade Instance Segmentation [3], which uses progressively updating the feature once added in every stage. Which uses Cascade boosts the performance for different tasks.

2.3.3 Similarity Group Proposal Network for 3D Point Cloud Instance Segmentation(2019)

SGPN utilizes a solitary network to predict point grouping proposals with its corresponding class for each detected class[27]. From here we can directly extract the results from the input image. It is used for 3-D images. It covers one the approaches related to instance segmentation.

2.3.4 YOLACT: Real-time Instance Segmentation(2019)

Also a work presents a model for real-time instance segmentation [2]. It first creates sample masks and each class mask coefficients. The process is not dependent on re-pooling the layers. The work was able to high-quality masks using the proposed approach. It is referred to as YOLACT.

2.4 Solution Proposed

From the study related to the literature, Mask-RCNN helps satisfy the objective of instance segmentation. With the help of Resnet101 architecture. So, the approach will be that model will take input as an image, then will pass through the Resnet architecture which will extract features from that image.

Features that are related to the currency in an image. Then it will be passed through Region Proposal Network, which will predict if the object is present in that image or not. And then will be taken through ROIs which will detect the anchors in an image.

After those created anchors, there will be a segmentation mask that will be added to the image. That will be used for individually identifying currency in an image. The output count will be audio of the number of denominations present in that image.

Chapter 3

Methodology

For understanding the mechanism behind Mask-RCNN, we have gone through some basic approaches that are used in computer vision. One such approach is the use of the Convolutional Neural Network. And how other methods were developed after it. That are RCNN, Fast-RCNN, and Faster-RCNN. Mask-RCNN is the modified version of Faster-RCNN.

3.1 Convolutional Neural Networks

The general approach for detecting the.The model is basically series of covolution and pooling operations at the end fully connected layers are used. At the end before output layer an activation function is applied(like Softmax for classification) to the output. Figure 3.1 takes image as an input and is passed through convolutions and pooling operations.

Pooling layers summarize the features that are present in an image. It is used to downsample features into feature maps. Convolutions apply learned filters to input images which creates feature maps.

The main building block of CNN is the convolutional layer. Convolution is applied using convolutional filter.

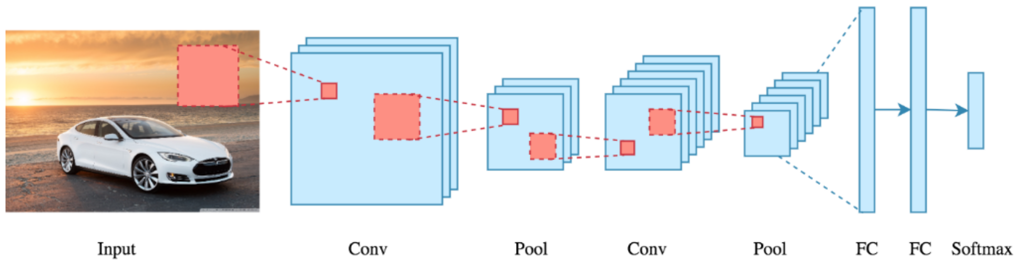

Figure 3.1: CNN architecture.It takes image as an input and is passed through convolutions and pooling operations.

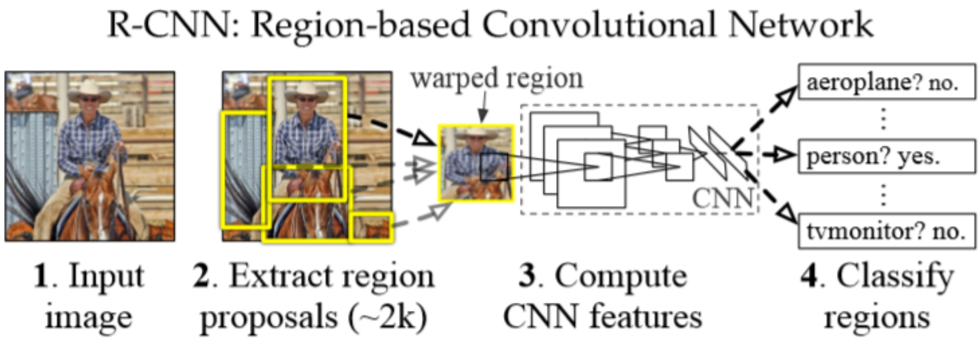

Figure 3.2: R-CNN architecture.
[8]

We cannot use a single CNN to identify a currency image that contains dense classes. It can be used for object detection related operations. We need a more dense model that can locate a currency present in an image. It can locate which pixels belong to that class. So there is a need for Region based CNN(RCNN).

3.2 Region based Convolutional Neural Network (RCNN)

The model combines the region proposals generated with Convolutional Neural Networks (CNN) for an input image[8]. Here it uses step by step approach for creating region proposals and applying CNN. Figure 3.2 describes the architecture which takes input image, region proposals, computing CNN features, and classifying based on that features. It trains a classifier and bounding box regressor in a sequential manner.

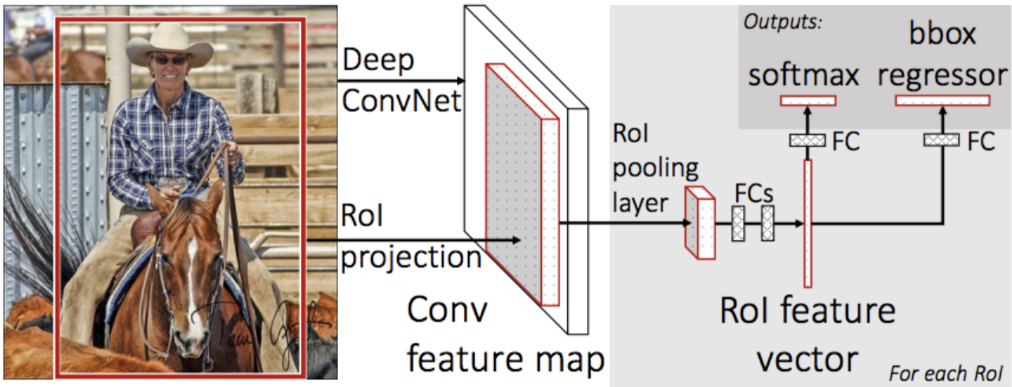

Figure 3.3: Fast R-CNN architecture. Working of Convnet and RoI projection for feature extraction.

[6]

Here the process is sequential and we cannot create a mask around the detected object. We need a process that is parallel and can help in locating and recognizing objects. For working in a parallel flow and also creating masks around that objects, we go through Fast-RCNN.

3.3 Fast RCNN

Drawbacks with RCNN

- Training is a multi-stage pipeline [6], it is a step by step process.
- Training is expensive
- Object detection is slow

Here the approach is same as RCNN.Here instead of passing the region proposals to CNN, input image is passed to CNN which generates convolutional feature map. It is the modified version of Region based Convolutional Neural Networks. It uses selective search for region proposals. Here classification and bounding box regressor are trained parallely. Figure 3.3 gives the architecture for Fast RCNN.

The drawback of this model is it takes more time.

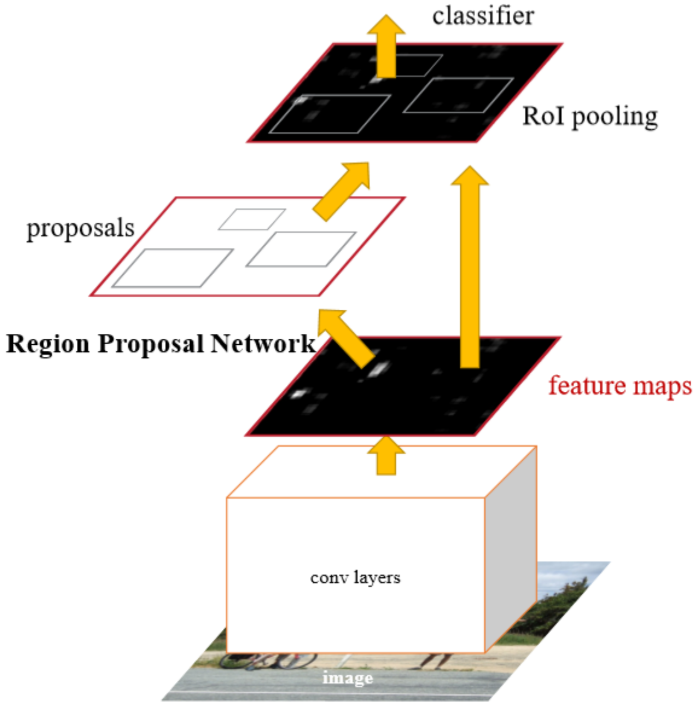

Figure 3.4: Faster R-CNN network for object detection. From bottom there is image passed through conv layers.

[22]

3.4 Faster RCNN

The approaches RCNN and Fast RCNN use selective search to find region proposals, which time consuming which affects the performance of the network [22]. Therefore [22] another algorithm that obliterates selective search and network learns the proposals is introduced.

Faster RCNN is similiar to Fast RCNN, the image is given as an input to the convolutional layer which provides the feature map. It introduces novel approach, which is Region Proposal Network that shares an image features with observation networks[22].

From figure 3.4 it can be visible from bottom up the image is passed through Conv layers with region proposal network and region of interest (RoI) pooling.

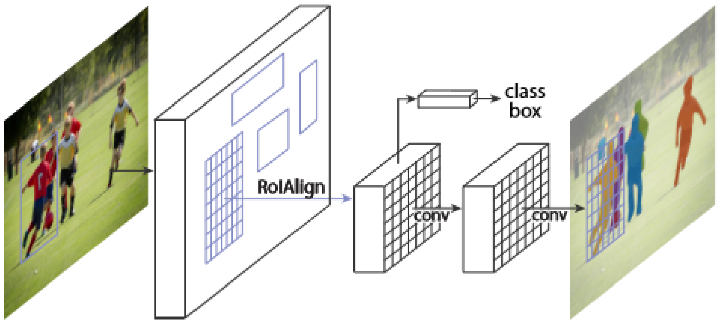

Figure 3.5: Mask R-CNN process for identification and mask generation.
[13]

3.5 Mask-RCNN

Mask-RCNN is modified form of Faster-RCNN, which applies another branch for mask generation. It parallelly adds a branch for predicting an object mask.(i.e. It is similiar to Faster-RCNN, it has a parallel head for predicting the mask). It adds another property which is RoI align which breaks pixel to pixel alignment. The model is used for current project, for detecting multiple currencies in an image. It is useful in solving instance segmentation problems which is useful for uniquely identifying individual currencies. Figure 3.5 represents the architecture of Mask-RCNN.

3.6 How Mask-RCNN helps

There are number of benefits using Mask-RCNN. Which are, Transfer Learning: We can use previously trained Mask-RCNN model which can help in training faster. Model is available for re-training, which makes it time efficient. Mask-RCNN is easy to train. As training takes one to two days on a eight-GPU machine[13]. Also here ROI Align is used which preserves the spatial locations of pixels.

Chapter 4

Implementation

4.1 Dataset

For training data, the dataset of currency images is created. Which contains 1500 images. From which 1000 images were used for a train set, 250 images for validation, and 250 images for testing. Time taken for preparing data is 5-6 days. For images that should be taken individually and dense.

1000 of the dataset were of single class images. Figures 4.1 and 4.2

4.1.1 Classes

The data contains 11 classes. Including one class for background in an image. Remaining 10 classes are 10 Rupees denomination with old and new, 20 Rupees old, 50 Rupees old and new,100 Rupees old and new,200 Rupees,500 Rupees, and 2000 Rupees.

4.1.2 Resolution

Input is taken for any resolution i.e., there is no specific limitation in terms of resolution. Once an image is passed for training image is resolution is changes to 1024×768 for simplification of model detection and mask generation. After detection the output is of

Figure 4.1: Dense currency image.

Figure 4.2: Single class image.

the original resolution.

4.2 Image Annotation

It is a part of identifying or marking each class manually. Also identifying each pixel that belongs to that particular object. It is identifying the image manually from the developer's end. So that it can be used by machines to identify each image. Many formats are there. One can be storing images of heights, widths, and other information in a CSV file. It is also done for creating bounding boxes around objects.

VGG is a simple annotation software for images, videos, and audios. It can be used in an HTML browser which takes less than 400 kilobytes of space [4]. It is an open-source project and is available at [26].

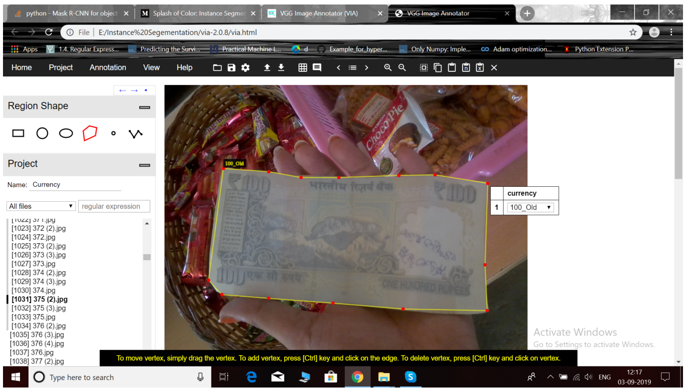

Figure 4.3: VGG annotation for 100 Rupee denomination.

4.2.1 How it is used for created data

It was used for annotation of 1100 images that were there in my dataset. Each one is annotated using a polygon format. It can be as per Figure 4.3 here the denomination of 100 Rupees is annotated with polygon format.

After annotation it creates a COCO formatted file which can be useful for giving input of each image to the machine. The tools can be used for other formats also, which are JSON, comma-separated values(CSV). But for model COCO format was required as it can be useful to simplify the input of each image.

4.2.2 COCO format for annotations

4.2.3 Other Annotation Tools

Here is the list of other annotation tools that can also be used for annotations [1]:

- Label me
- RectLabel
- Labelbox
- COCO UI

Figure 4.4: Annotation format which can be used to identify each currency.

```
{   "images":{"id":"0","filename":"100_new.jpg "},
    "annotations":{"id":"0","image_id":"0",
    "category_id" = "2",
    "area":"1125705","bbox"},
    "categories":{"100_new":"0"}
}
```

Listing 1: COCO format example which is same used format for training. Here category id is used to identify different classes.

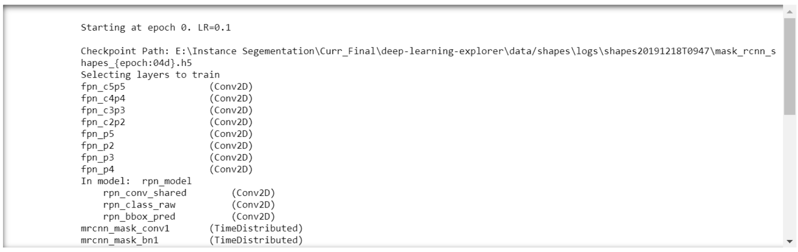

Figure 4.5: Training(a).

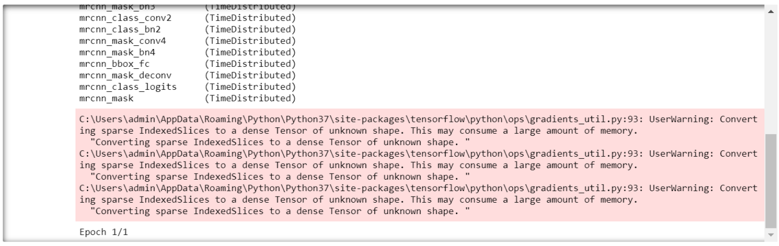

Figure 4.6: Training(b).

The above tools can be used but for VGG was more efficient when it comes to the annotation of images. It took 2-3 minutes per image for the annotation of a single class in it. With multiple classes it took 5 minutes per image.

4.3 Transfer Learning

There is a previously trained Mask-RCNN model available at [28]. It is used to training different shapes for instance segmentation. Using this, we don't need to train a model from scratch. This approach is described as transfer learning. The model generates a model file that is in (.h5) format.

4.4 Image Identification

4.4.1 Training

The training is completed using 1100 images with 1000 single class images which consisted of only one denomination, the remaining 100 were also included which contains multiple classes.

From figure 4.5 and 4.6 represents trainig of data

4.4.2 Configuration used

The system can be used for any machine i.e. for Windows,Ubuntu and MacOS. Provided it needs python as a basic requirement.

Following is the configuration used for model creation:

- Backbone: Resnet101
- Backbone strides: [4,8,16,32,64]
- Batch Size:1
- BBox Standard deviation:[0.1,0.1,0.2,0.2]
- GPU used: Tesla K80, Nvidia
- Learning rate:0.001,initial training is done using 0.1.
- Total Epochs: 201
- Steps per epoch: 400
- Image shape:[64,64,3]

4.4.3 Dependencies

These are the libraries that were used for instance segmentation. For backend python is used for preparation:

- Python: 3.7.4
- Tensorflow: 1.15.0
- Keras: 2.2.5
- Scikit-Image: 0.15.0

- imgaug:0.2.9
- gTTS:2.1.1
- Flask:1.1.2
- Flask Bootstrap: 3.3.7.1.dev1

4.4.4 Training Time

Ideally, it should take 19 hrs here the time taken was 3 days, if we consider on paper results and number of epochs [13]. For training 201 epochs, each epoch with 400 steps. The learning rate for the initial 50 epochs with 0.1, after those 50 epochs, the learning rate is changed to 0.001. So, a total of 251 epochs were used for training. The total time taken for training is 72 hrs.

4.5 Text to Speech

4.5.1 Google Text to Speech (gTTS)

The library is used for converting text to speech output. It is with Google translate text-to-speech API [9]. For the current project it generates an mp3 file of generated images which is then called with created python API. So that is can directly play the audio after detection. Figure 4.8 shows basic implementation of gTTS.

This is the last module from figure 1.8 which converts the generated text output into speech.

For speech, the input is taken from the detected image. That is in string format.

4.5.2 Other Text to Speech Approach

Python text to speech (pyttsx) It is a python package that can be used for windows, Linux, and Mac systems. It can also be considered for text to speech conversion of detected images [10]. The output generated is in the string format.

Write ‘hello’ in English to `hello.mp3`:

```
>>> from gtts import gTTS
>>> tts = gTTS('hello', lang='en')
>>> tts.save('hello.mp3')
```

Figure 4.7: Google text to speech example.

Usage :

```
import pyttsx3
engine = pyttsx3.init()
engine.say("I will speak this text")
engine.runAndWait()
```

Figure 4.8: Python text to speech usage example.

Chapter 5

Results

5.1 Image Identification Results

For input the image in figure 4.1 and 4.2 are used. The expected output for the given input are Figure .Figure 5.2 and 5.4 are the detection results that were generated after training.

For detection result figure 5.2 there will be a speech related to that output. Which is "The image contains two notes of ten Rupees, one note of twenty Rupee, two notes of fifty Rupee, two notes of hundred Rupee, one note of two hundred Rupee, one note of five hundred Rupee, one note of two thousand Rupees".

5.2 End to End system

For an end to end system initial idea was to go for a web application. But it cannot be converted to lightweight Tensorflow-lite. Also if it is converted then it will be time taking and accuracy may be compromised.In order to overcome this challenge we have created a web API. The web API is created for the project which can be used for identification.

The created API can be used for creating Web Page. Which can be referred from figure 5.5, 5.6 and, 5.7. Figure 5.5 is the home page. From home page after clicking the "Currency" button.It will redirect to the upload page. After redirecting it will take

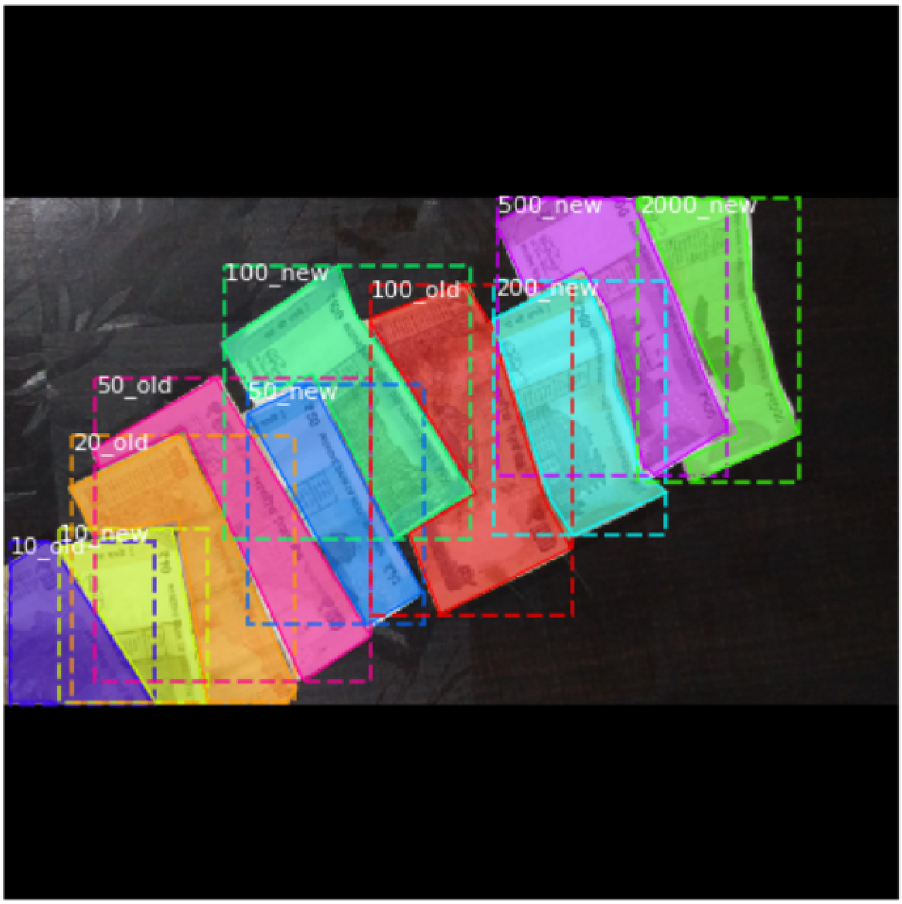

Figure 5.1: Expected output for figure 4.1 as input.

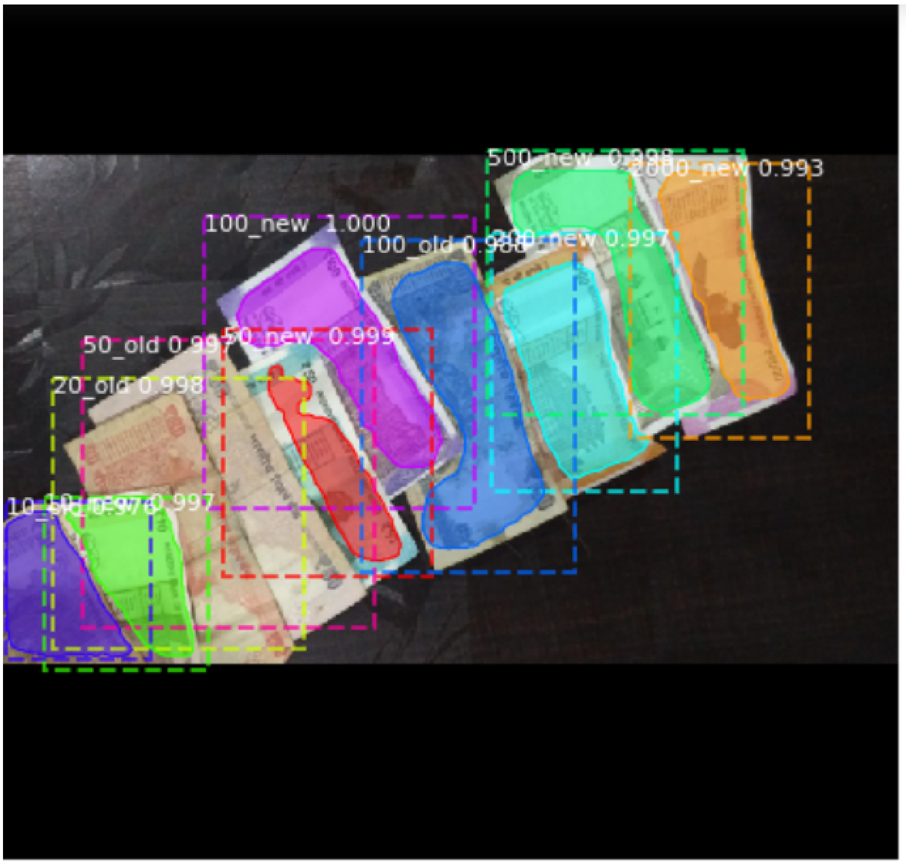

Figure 5.2: Output of image identification using image currency as input with dense multiple currencies.

Figure 5.3: Expected output for figure 4.2 as input.

Figure 5.4: Output of image identification using image currency as input with single image.

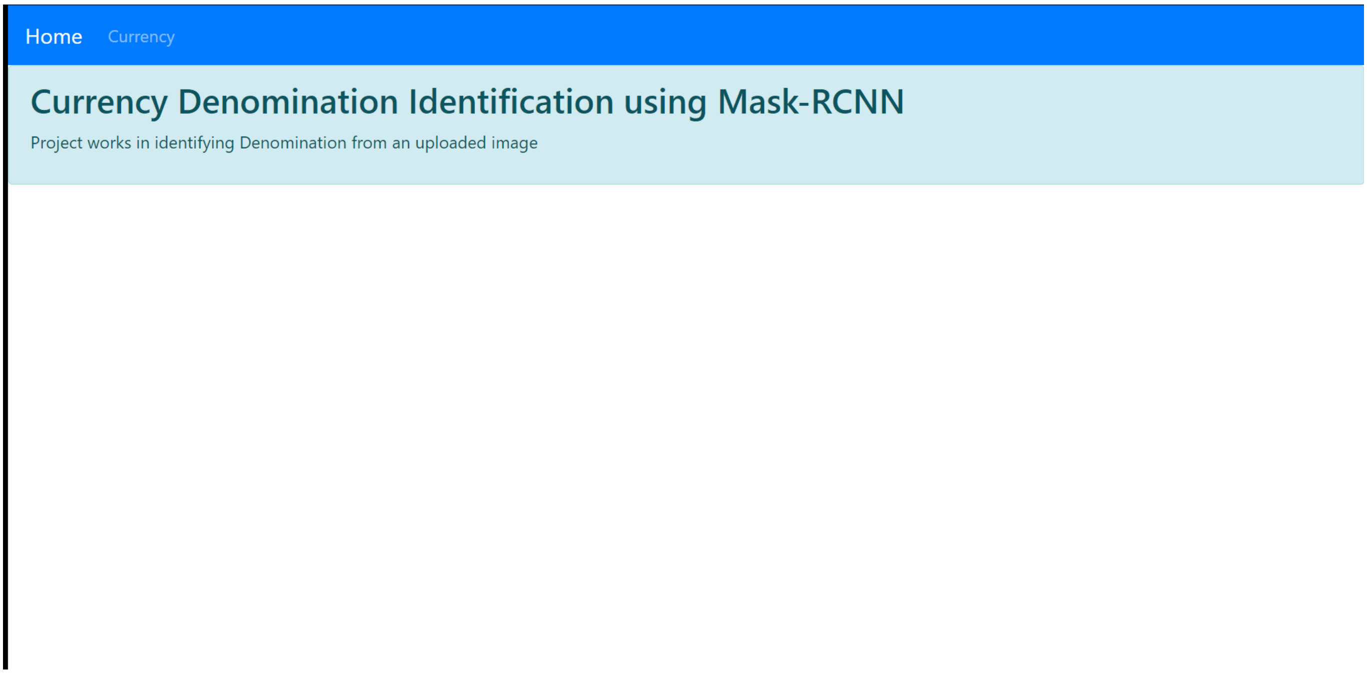

Figure 5.5: Home page for denomination identification.

image as input and will display the results. Figure 5.7 gives output of detected image and detected number of denominations.

5.3 Limitation

There are some challenges related to Mask-RCNN. Till now it is not available for converting the given model into Tensorflow lite for application development. Also it may degrade the results that are achieved currently.

For such model quantization can be used but it will not be able to achieve the results.

5.4 Conclusion

The generated model satisfies the objective of identifying multiple currencies present in an image. With Mean Average Precision (MAP) of 80 percent, for reference we can see in figure 5.8. Further it takes two minutes for overall implementation. Which is created using flask-bootstrap. Also the accuracy that it works is about 90%.

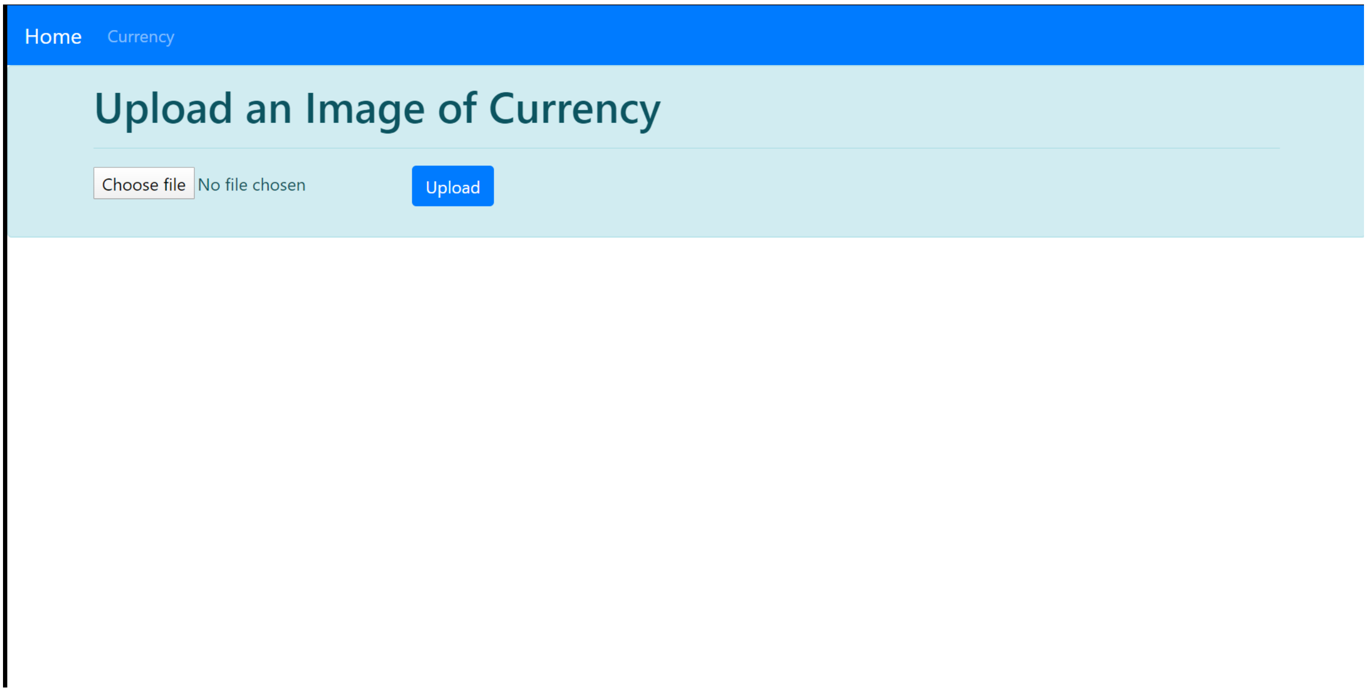

Figure 5.6: Web page for image upload.

Figure 5.7: Output page after the image is uploaded for identification .

Average Precision

```
10_old (1): 1.0
10_new  (1): 1.0
20_old (1): 0.0
50_old (1): 0.0
50_new  (1): 1.0
100_old (1): 1.0
100_new  (1): 1.0
200_new (1): 1.0
500_new  (1): 1.0
2000_new (1): 1.0
--------
Mean Average Precision MAP: 80.0
```

Figure 5.8: Mean average precision for each class.

The model is working with Indian currencies that were present until August 2019. It will not be able to identify concerning recent changes(i.e., notes that were changed post-August 2019). Overall in future performance improvement, generated API can also be used.

5.5 Future Work

For preparing the model that can be the same as currently used it will be challenging for creating the same annotations available, as it was more time taking. Also performance can be improved for creating it as a web application. Currently API is called for getting the output. The same created API can be used for generating improved results. Time taken by a single call is two minutes only when it is used as a web page. But for application both detection and result generation can be decreased.

Overall working on creating the application can be one thing for future development. Also the created API can be used for identification and audio output.

References

[1] Waleed Abdulla. *Splash of Color: Instance Segmentation with Mask R-CNN and TensorFlow*. https://engineering.matterport.com/splash-of-color-instance-segmentation-with-mask-r-cnn-and-tensorflow-7c761e238b46. 2018.

[2] Daniel Bolya et al. *YOLACT: Real-time Instance Segmentation*. 2019. arXiv: 1904.02689 [cs.CV].

[3] Kai Chen et al. *Hybrid Task Cascade for Instance Segmentation*. 2019. arXiv: 1901.07518 [cs.CV].

[4] Abhishek Dutta and Andrew Zisserman. "The VGG image annotator (VIA)". In: *arXiv preprint arXiv:1904.10699* (2019).

[5] A. Frosini, M. Gori, and P. Priami. "A neural network-based model for paper currency recognition and verification". In: *IEEE Transactions on Neural Networks* 7.6 (1996), pp. 1482–1490.

[6] Ross Girshick. "Fast r-cnn". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1440–1448.

[7] Ross Girshick et al. "Region-based convolutional networks for accurate object detection and segmentation". In: *IEEE transactions on pattern analysis and machine intelligence* 38.1 (2015), pp. 142–158.

[8] Ross Girshick et al. "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2014.

[9] *Google Text to Speech*. https://gtts.readthedocs.io/en/latest/.

[10] *Google Text to Speech*. https://gtts.readthedocs.io/en/latest/.

[11] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[12] Kaiming He et al. "Identity mappings in deep residual networks". In: *European conference on computer vision*. Springer. 2016, pp. 630–645.

[13] Kaiming He et al. "Mask r-cnn". In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2961–2969.

[14] T Huang. "Computer Vision: Evolution And Promise". In: (1996). DOI: 10.5170/CERN-1996-008.21. URL: http://cds.cern.ch/record/400313.

[15] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems*. 2012, pp. 1097–1105.

[16] James Lee. *Computer Vision Techniques*. https://heartbeat.fritz.ai/the-5-computer-vision-techniques-that-will-change-how-you-see-the-world-1ee19334354b. 2018.

[17] Tsung-Yi Lin et al. "Feature pyramid networks for object detection". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 2117–2125.

[18] Ingo Lütkebohle. *Fake Currency Identification*. https://github.com/hritik25/DRDO-Fake-Currency-Identification. 2018.

[19] Ingo Lütkebohle. *Indian-Currency-Recognition*. https://github.com/10zinten/Indian-Currency-Recognition. 2018.

[20] *Mani App*. https://play.google.com/store/apps/details?id=com.rbi.manihl=en$_IN$.

[21] Pedro O. Pinheiro, Ronan Collobert, and Piotr Dollar. *Learning to Segment Object Candidates*. 2015. arXiv: 1506.06204 [cs.CV].

[22] Shaoqing Ren et al. "Faster r-cnn: Towards real-time object detection with region proposal networks". In: *Advances in neural information processing systems*. 2015, pp. 91–99.

[23] Noura Semary et al. "Currency Recognition System for Visually Impaired: Egyptian Banknote as a Study Case". In: Dec. 2015. DOI: 10.1109/ICTA.2015.7426896.

[24] Md Shahjahan et al. "A currency recognition system using negatively correlated neural network ensemble". In: *2009 12th International Conference on Computers and Information Technology*. IEEE. 2009, pp. 367–372.

[25] Amol A. Shirsath and Sangita D. Bharkad. "Survey Of Currency Recognition System Using Image Processing". In: 2013.

[26] *Visual Geometry Group*. http://www.robots.ox.ac.uk/ vgg/software/via/.

[27] Weiyue Wang et al. *SGPN: Similarity Group Proposal Network for 3D Point Cloud Instance Segmentation*. 2017. arXiv: 1711.08588 [cs.CV].

[28] *Waspinator Mask-RCNN*.

[29] Wei Qi Yan, Yueqiu Ren, and Minh Nguyen. "Real-Time Recognition of Series Seven New Zealand Banknotes". In: *Int. J. Digit. Crime For.* 10.3 (July 2018), pp. 50–65. ISSN: 1941-6210. DOI: 10.4018/IJDCF.2018070105. URL: https://doi.org/10.4018/IJDCF.2018070105.