University of South Wales
Prifysgol De Cymru

Deep Learning Model for Age Prediction from Fascial Image

30091254

1. Introduction

Age is one of the important features for the identification of a person. Now-a-days human faces are being used in many sectors like fascial expression recognition, face detection, gender classification as well as for biometric and security purposes. Our aim is to make a model with deep learning that can predict human ages from the fascial images.

2. Model Description

In our study we use CNN model. Convolutional Neutral Network is an algorithm of deep learning that is mostly used for image classification. This model works with different layers like convolutional layer, pooling layer and fully connected layer.

3. Data Details

There are 19906 images of young, middle and old ages. These labels are defined in a CSV and images are in different folder. We need to merge them together for the further process.

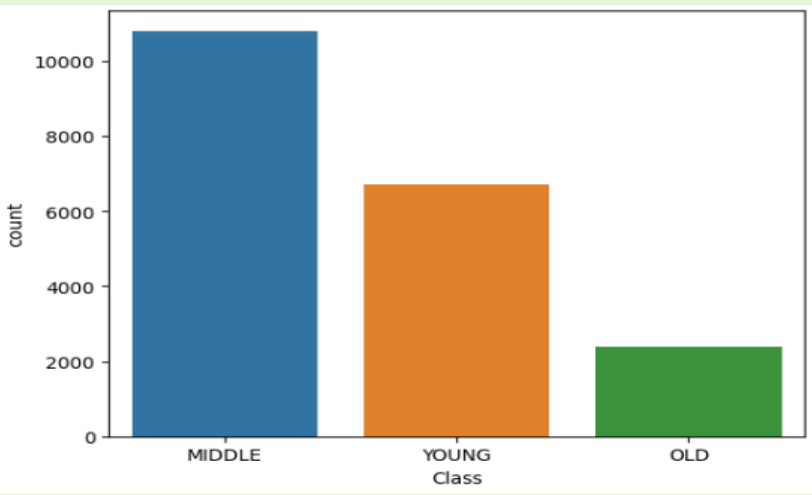

Fig 1:Age classes

4. Data Processing

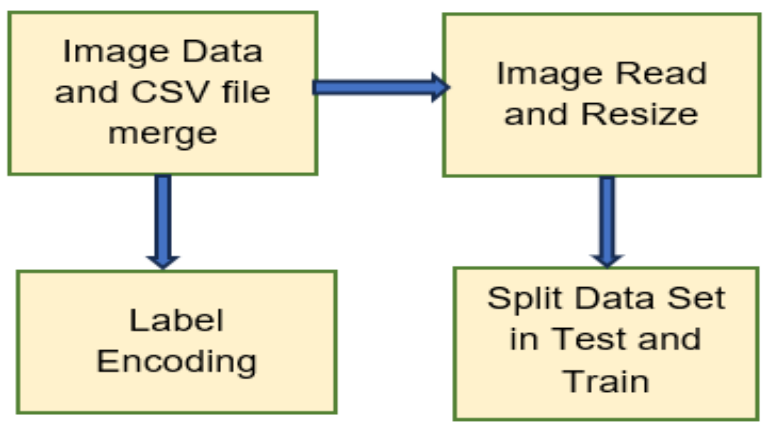

Fig 2: Flowchart for data processing

Train and Test Data set

For training and testing we need to split the image dataset. Among 19906 images 15879 images are assigned in train data set and 3969 images are assigned for test dataset.

Label Encoding

Label encoding is a technique to encode categorical data. The categorical values are assigned with some numeric values. Here we assigned young, middle, old as 0,1,2.

```
data['Class'].replace(['YOUNG','MIDDLE','OLD'],[0,1,2],inplace=True)
data.head(5)
```

	ID	Class
0	377.jpg	1
1	17814.jpg	0
2	21283.jpg	1
3	16496.jpg	0
4	4487.jpg	1

Fig 3: Code snippet for Label Encoding

5. Image Augmentation

In our study we use grayscale, random brightness, adjust saturation and rot90 augmentation techniques

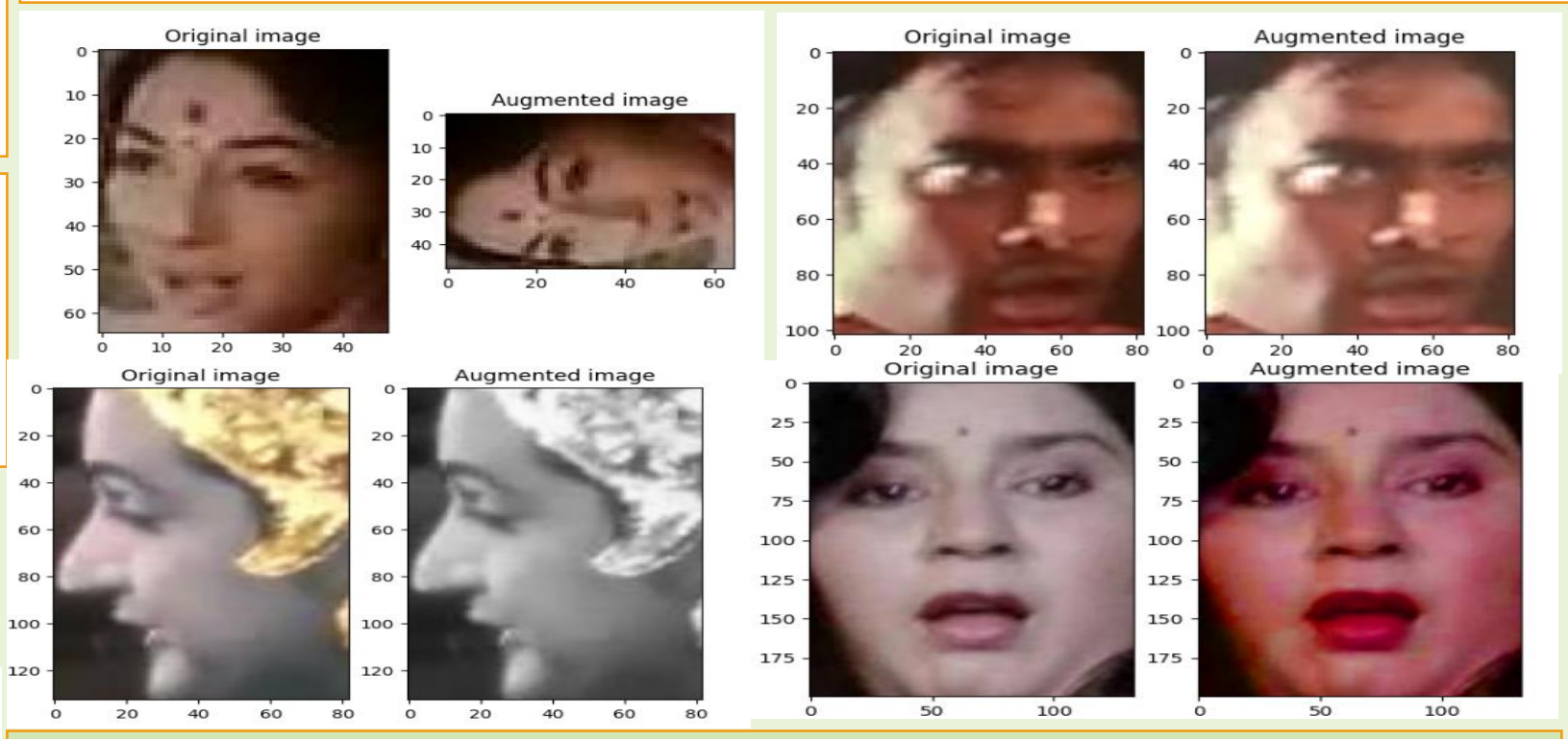

Fig 4: Augmented Images

6. Training

We use CNN for training our model. We split data set in train and test set where 80% are in train set and 20% in test set. We generate CNN model. We compile the model.

7. Result

We run 30 epochs to train the model. The train accuracy is 98% and test accuracy is 89%. The train loss is 3% and train loss is 87%

```
Epoch 26/30
498/498 [==============================] - 40s 81ms/step - loss: 0.0306 - accuracy: 0.9903 - val_loss: 0.8782 - val_accuracy: 0.8914
Epoch 27/30
498/498 [==============================] - 42s 84ms/step - loss: 0.0321 - accuracy: 0.9907 - val_loss: 0.9208 - val_accuracy: 0.8830
Epoch 28/30
498/498 [==============================] - 693s 1s/step - loss: 0.0380 - accuracy: 0.9881 - val_loss: 1.0305 - val_accuracy: 0.8816
Epoch 29/30
498/498 [==============================] - 43s 86ms/step - loss: 0.0304 - accuracy: 0.9916 - val_loss: 0.9088 - val_accuracy: 0.8916
Epoch 30/30
498/498 [==============================] - 41s 83ms/step - loss: 0.0380 - accuracy: 0.9895 - val_loss: 0.8727 - val_accuracy: 0.8998
```

Fig 5 : Epochs Run Accuracy and Loss

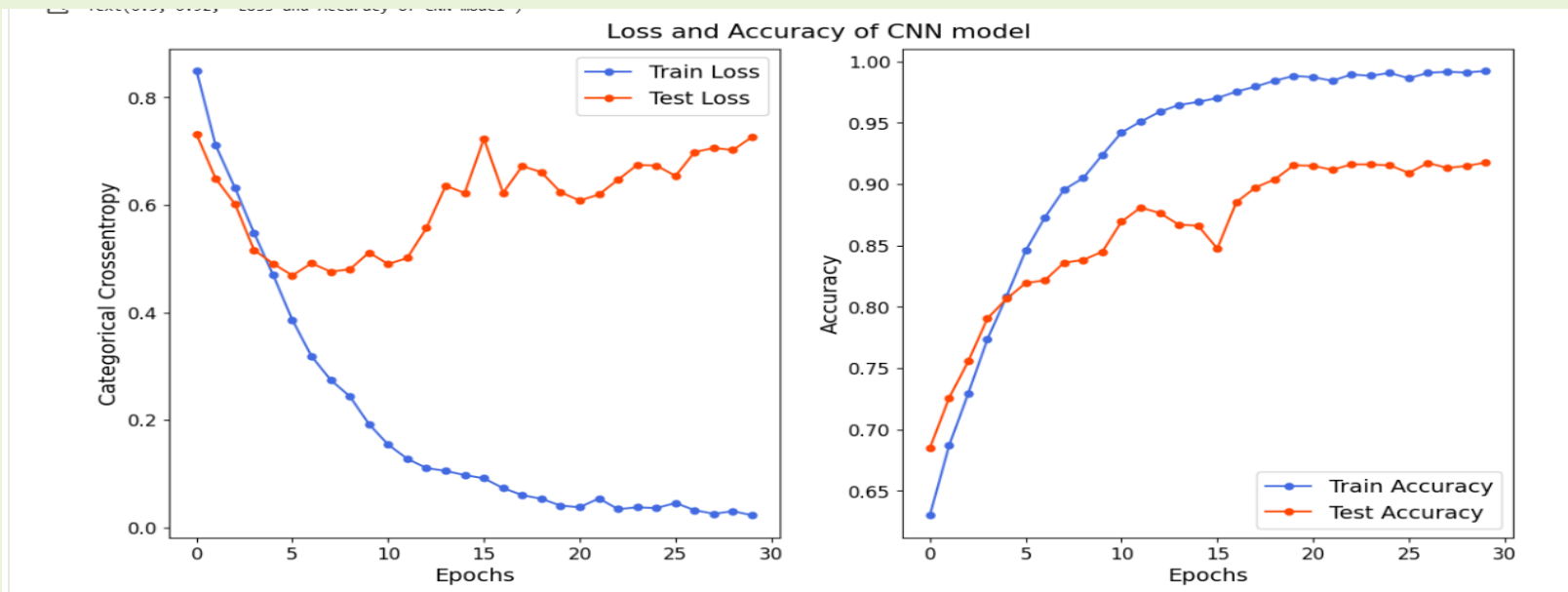

Fig 6: Line chart for Loss and Accuracy

8. Deployment

We use Anvil app for the deployment of the model.

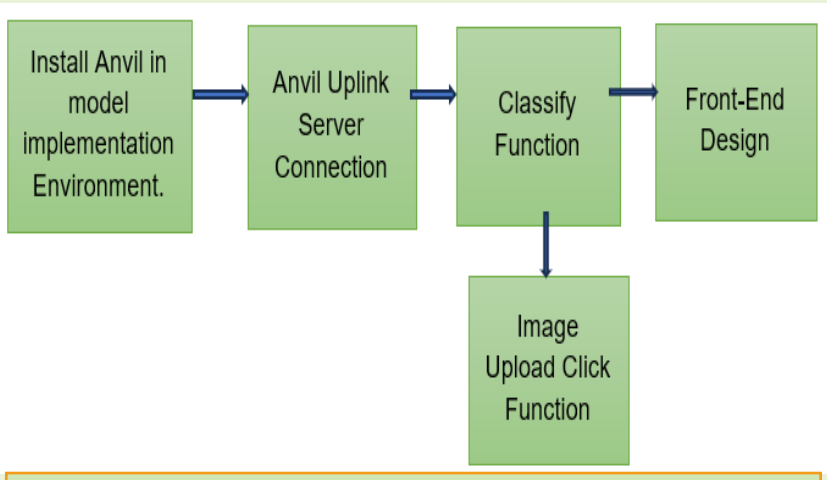

Fig 7: Anvil app workflow

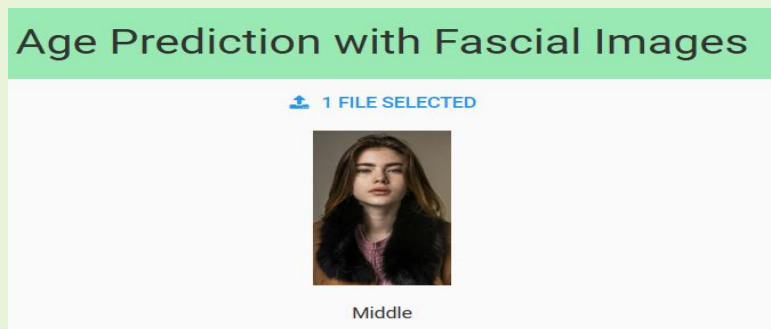

Fig 9: Demo Anvil App Age Prediction

9. Conclusion

Our study works on deep learning model that predicts ages from the human images. Our CNN model is deployed in anvil. We can upload images and our model predict the ages. As our images and labels of the images are in different folder, we face challenges to merge them.