RECOGNITION OF FACIAL EMOTIONSIN CARTOONS USING DEEP LEARNING

By

B.Pranathi

Project Guide: Dr.Malmathanraj
Assistant Professor
NIT TRICHY - ECE

ABSTRACT (concised)

From many years Human emotion recognition is an active research topic. The human emotions are biological states associated with neuron system. In cartoons the emotions are a kind of imitation of human emotions.

Different emotions are caused because of different thoughts and Our thinking process is affected by our viewing habits.Children spend a lot of time watching cartoons.The quality of the shows has to be monitored and emotions will help us analysing them to certain extent.

This project aims at developing a model which can recognise emotions in cartoons.

OBJECTIVES :

1. This project is primarily an attempt to understand the deep learning efficiency in cartoons and develop a model to accurately recognise emotion in cartoons.
2. Create our own haar cascades file (object detection file) to detect the cartoon faces.
3. Building the dataset.
4. Design an appropriate model for training the cartoon emotion recognition.
5. Evaluate the different models and propose a model based on the accuracy.

Literature survey and Novelty:

1. Previously there is no work done in the area of emotion recognition in cartoons.

2. The close relevant research is in humans and the accuracy results of ~(95-96)% are achieved through the advancements of different techniques over the years.

3. This project also develops a file to recognise cartoon over a video. None exists previously as there is no research made in this area.

Societal relevance:

1. There is a lot of research in the study of emotions in the HCI (Human Computer Interaction). When study on the content on the screen is studied together with the reaction of different humans will create a better understanding of the cognitive behaviour. A similar kind of study can be made on the text we see on the screen.

2. Animators in the 2D cartoon domain, cartoonists would benefit from this.

3. Can find some space in automatic subtitle works and recommendation systems.

Tools Used:

1. VLC player
2. Image Cropping tool
3. Cascade-Trainer GUI application
4. Python ->3.8.8
5. Tensorflow ->2.4.0
6. Keras ->2.4.3
7. Numpy ->1.19.2
8. cv2 ->4.5.1
9. Jupyter Notebook.

Building the Dataset

1. Different videos of tom and jerry are downloaded from you tube.
2. Then collected frames from VLC player and segregated images for three different emotions.
3. All the collected images are cropped and made the dataset.
4. Developed haar cascades file using Cascade GUI Trainer application to auto detect faces automatically but kept aside at the beginning because of its poor efficiency.

Haar cascade classifier:

1.The haar features based cascade classifiers are effective for object detection developed using positive and negative images.

2. The positive images are taken from the dataset and the negative images can be anything else but here we have taken related cartoon backgrounds for good efficiency.

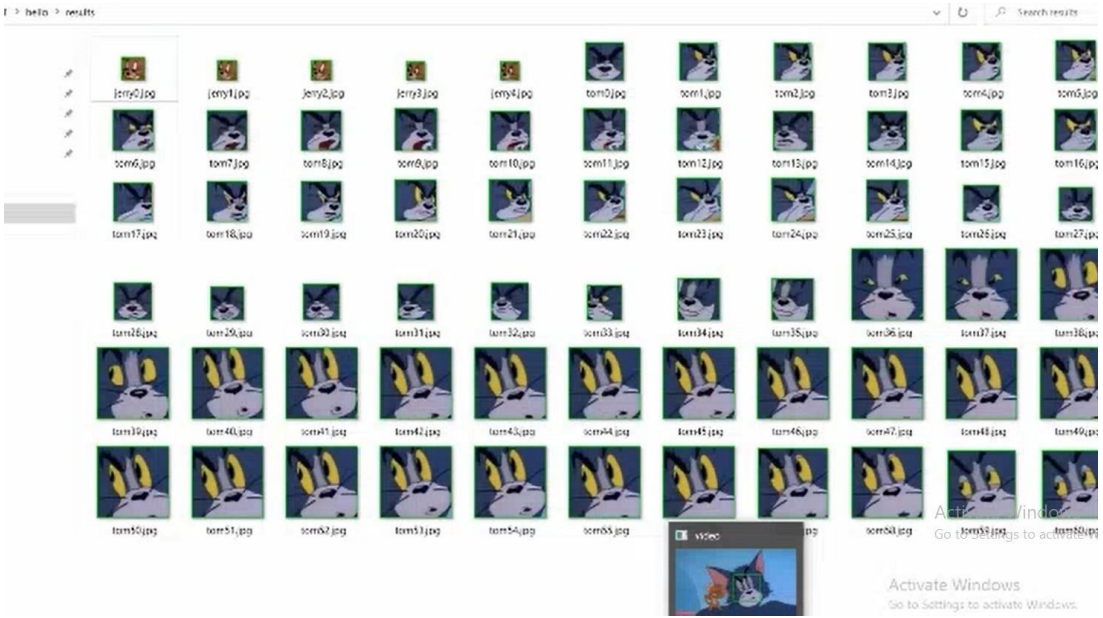
3. The cascade gui trainer application is used for the purpose of developing the classifier.

Auto Segmentation of cartoon faces:

1. The difficulty here is there is no pre-existing file to detect faces in cartoons.
2. We have improved the accuracy of the built haar cascades file by tweaking epochs during training, min neighbours while testing.
3. After going through the difficulty in preparing the dataset , we felt it is important to develop a auto segmentation method for any further improvements.

Auto Segmentation of cartoon faces(continued):

1. The haar cascades file is used to detect the faces in the cartoons.
2. Using Opencv tools, the detected faces are stored in the required directory.



Proposed methodology:

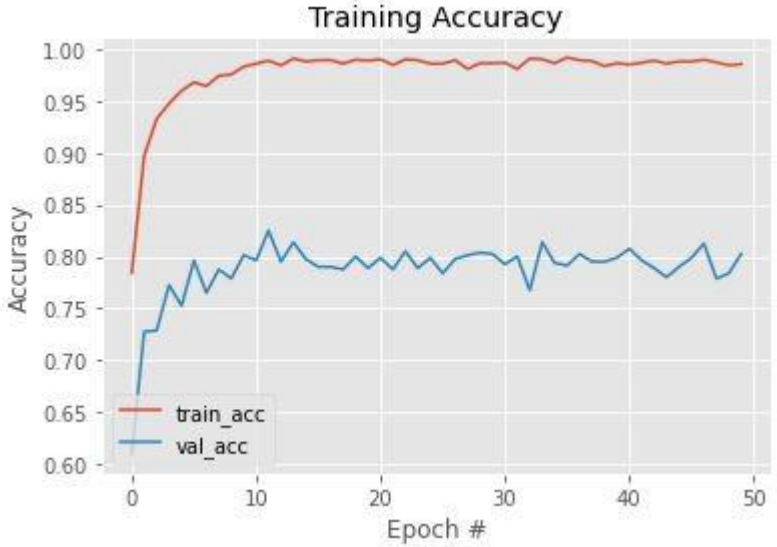
1. The neural network models in consideration are CNN and RNN.
2. The CNN model is found to be good over spatial data, And thus gives good results in recognising emotions from images.
3. The RNN model is found to be good for temporal data, and thus gives good results for applications over videos.
4. Theoritically RNN can be used for image classification but there are very few works done using this. So, we have tried RNN for image classification.

Convolution Neural Network

1. With the build dataset , to avoid overfitting much deeper neural network has not been taken.
2. Neural Network with 3-4 convolutional layers , and followed by two fully connected layers has been taken.
3. Tweaking of parameters with different optimization algorithms and change in the number of filters at each layer , change in many other parameters are made and studied.

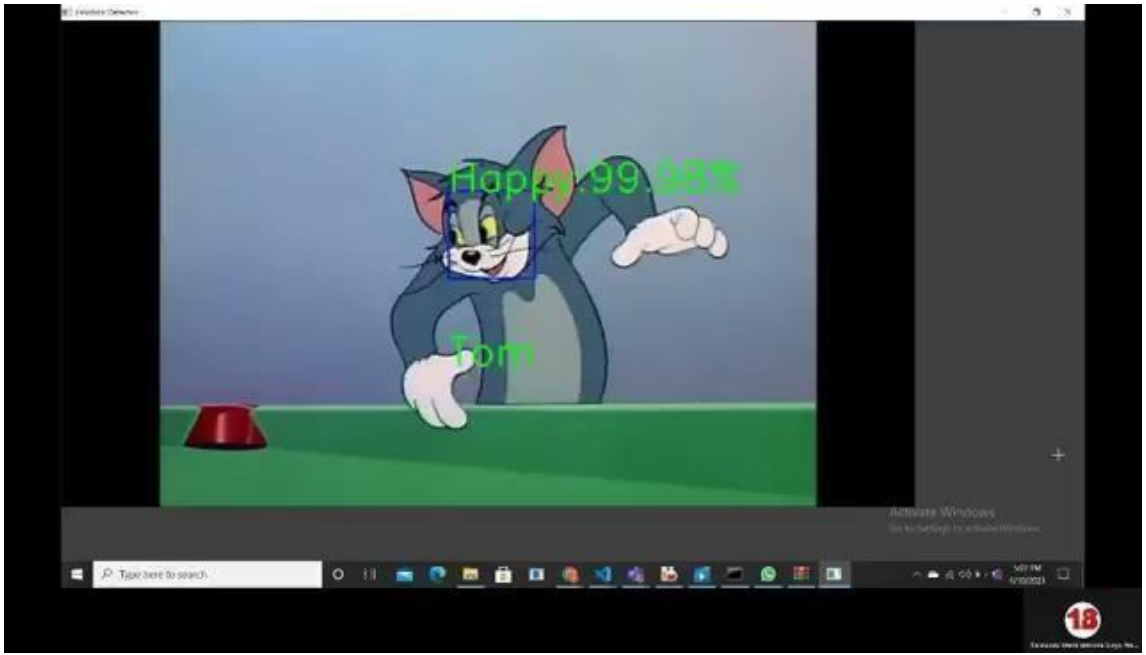
Results:

The below graphs are the plots of loss and accuracy for the best model developed so far. The optimiser used is “SGD” (Stochastic Gradient Descent) with a learning rate of 0.01.



Visualisation of results:

The model is saved during the training process. The saved model can be tested over images. Here a video is taken as frames (i.e as still images) and the saved model is tested.



Analysis of the results of CNN:

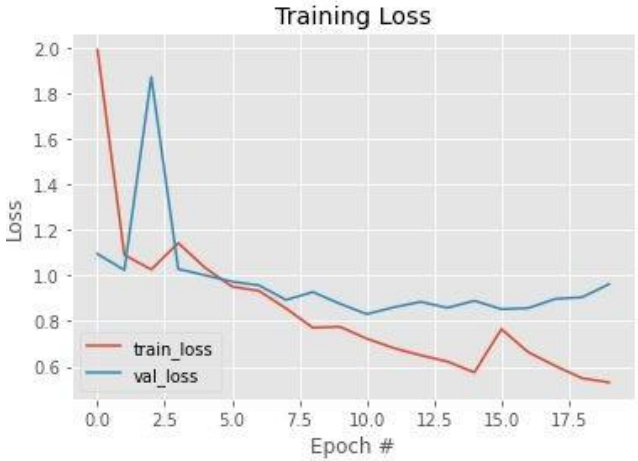
1. We have achieved accuracy of ~80% and loss of 0.7 . Results of the training the dataset can be improved further if we add more emotions.
2. The network was shallower than we had expected, there are problems of overfitting when tested with more convolutional and pooling layers.
3. The dropout layer was found helpful to some extent to reduce the overfitting.
4. The time taken for training the model can be reduced by using the GPU version of the tensorflow.

Recurrent Neural Network:

1. Usually RNN model learns to recognize image features over time. The model would generally take a video clip and each frame would have been flattened to a single array of pixels.
2. But since we are dealing with images alone, the row pixels are given as time steps and column pixels are given as features. The conversion of the images to required shape is done through a Lambda layer which helps in stateless computation.
3. The model has two LSTM blocks(to avoid problem with standard RNN) with relu actiavtion and followed by two dense layers, The last dense layer with softmax activation gives us the output.

Results:

The below graphs are the plots of loss and accuracy for the best model developed so far. The optimiser used is “Adam” (Adaptive moments) with a learning rate of 0.001.



Analysis of the results of RNN:

1. We have achieved accuracy of ~70% and loss of nearly 1. The results are comparitively low in comparison with the CNN model.
2. The reasons for less accuracy could be because of using images for RNN model which best suites for Videos. Further study is required to analyse the RNN model.
3. The outputs run at different times will give different plots because the smaples are randomized every time and also the memory element in LSTM treats the image differently.
4. The training time can be reduced using GPU version as it has CuDNN Lstm layer which speeds up the processing.

Future Scope:

1. The model can be developed by using a mixed CNN-RNN model , The mixed model requires a short video clips containing 30-120 frames.
2. The size of the dataset can be increased by adding some more emotions would help in imporving the results.
3. Some more cartoons can be taken and developed in together which can be useful for generalising the theme.
4. A similar kind of research can be done in the area of analysing the text, as different text information conveys different emotions.

Thank You