ELL888- Assignment 2

Manoj rathor (2016EEY7520), Priyank soni (2016EET2639), Krishna prasanth (2016EET2630)

1. Introduction: In this assignment we need to identify the dominant speaker in a video, where:

- We have six speakers, 7-classes (class-7 for none).
- We object detection methods like CascadeClassifier in **opencv**, **yolo** object detection algorithm to detect person in video frames.
- We are training some CNN architectures like **VGG16**, **ResNet50**, **InceptionNetv3**, etc with various settings on extracted frames and also on detected faces from frames.

2. Dataset preparation:

- Downloaded 20-25 videos (720p) of each speaker from youtube.
- We used different videos for frame extraction rather than extracting more frames from same video.
- Used VideoCapture() and read() opencv functions to extract frames from videos with desired rate.
- We First tried Haar CascadeClassifier in opencv for face detection from frames:
  Advantages: Speed, good performance.
  Limitation: Sadhguru's face not detected.

So we moved to **Yolo** algorithm
Split frames into 2 categories using yolo:
1.Frames without person object.
2.Frames with person object.

After that Frames with person again split into two categories:
1. Frames with one person.
2. Frames with multiple persons.

For frames with multiple persons we cropped the person with highest probability of person (more than 98%) given by yolo.
Now frames with one person Further split into:
1.Required speaker
2. Other person

With the help of pixel wise difference between frame of required speaker and other person.
We tried with:

- Took 2-norm (also tried with 1-norm ,infinity-norm) of each frame (norm$_i$)
- Calculated the average of norms (norm$_{avg}$)
- Took the difference between norm$_i$ and norm$_{avg}$, if absolute difference is less than threshold (eg. 10000) than frame was labeled as desired speaker otherwise discarded.
- Removed some noisy frames or labeled some of them as they do not contain speaker.

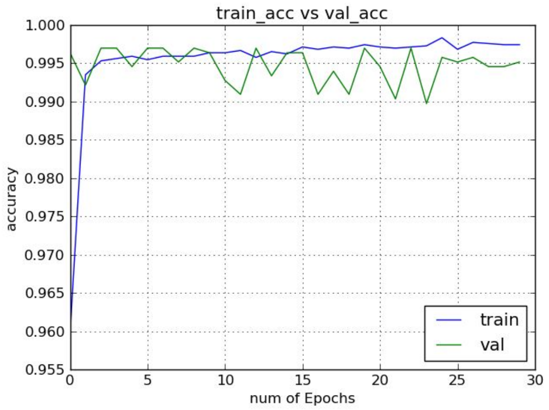
3. Training and testing:

3.1 VGG16 with only classifier trained

- Using video frames as it is (face detection not applied):

Here we freezed all the layers and training only classifier (softmax) having 7 classes with our data.

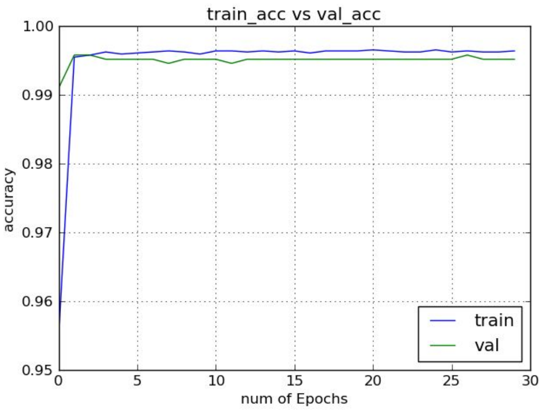
No. of examples	No. of epochs	Validation acc.	Test acc.
8297	30	99.54%	62.05%
3142	30	98.73%	59.08%

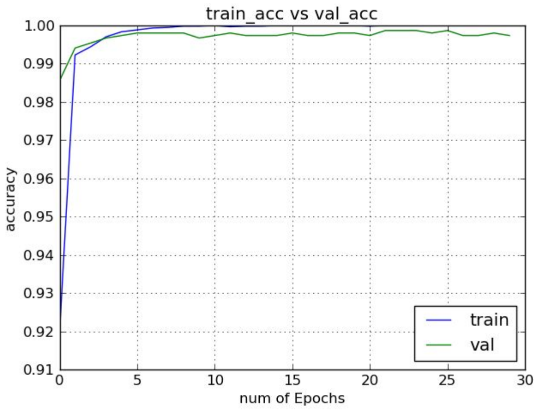


This model was trained with loss='categorical_crossentropy' and optimizer='rmsprop'.

- Training with detected faces

No. of examples	No. of epochs	Validation acc.	Test acc.
7623	30	99.74%	69.49%





Here reason of high test accuracy may be that we are using test frames from same videos, so model is overfitting.

- Training with detected faces

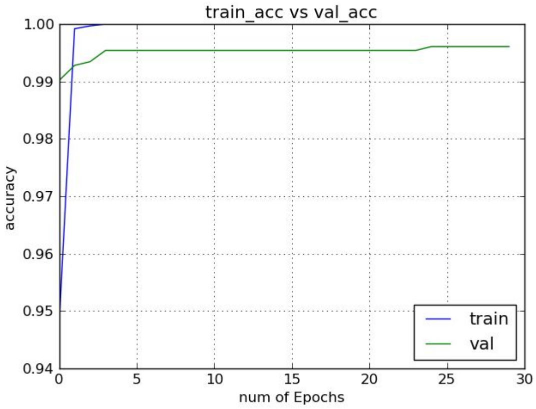
No. of examples	No. of epochs	Validation acc.	Test acc.
7623	30	99.61%	62.16%

3.2 VGG16 with tunable fc1, fc2, softmax classifier

Here we are training all layers except dense layers (fc1, fc2) with 128 nodes in each fully connected layer and 7 in softmax layer.

- Using video frames as it is (face detection not applied):

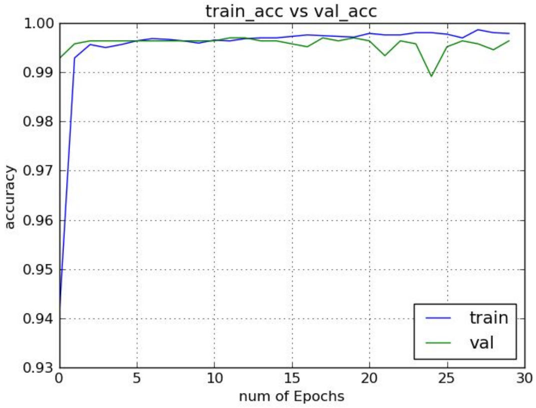
No. of examples	No. of epochs	Validation acc.	Test acc.
8297	30	99.52%	57.47%
3142	30	98.84%	51.08%



3.3 Resnet50 with only classifier trained

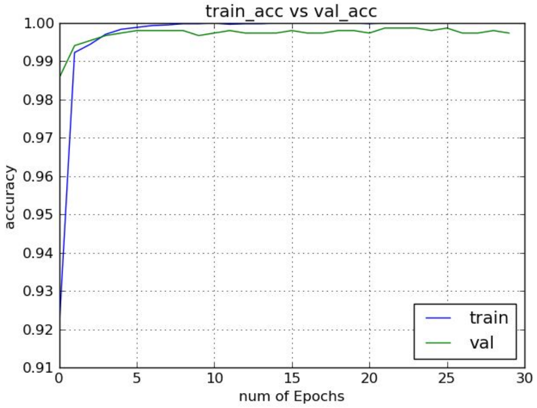
- Using video frames as it is (face detection not applied):

No. of examples	No. of epochs	Validation acc.	Test acc.
8297	30	99.31%	59.69%
3142	30	98.73%	51.08%



- Training with detected faces

No. of examples	No. of epochs	Validation acc.	Test acc.
7623	30	99.74%	75.16%

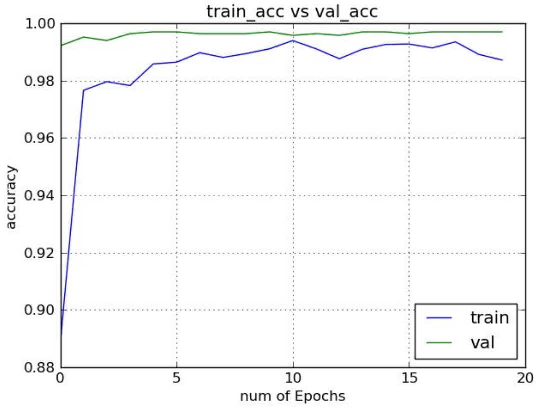


3.4 Resnet50 with added fully connected and dropout layers

- Using video frames as it is (face detection not applied):

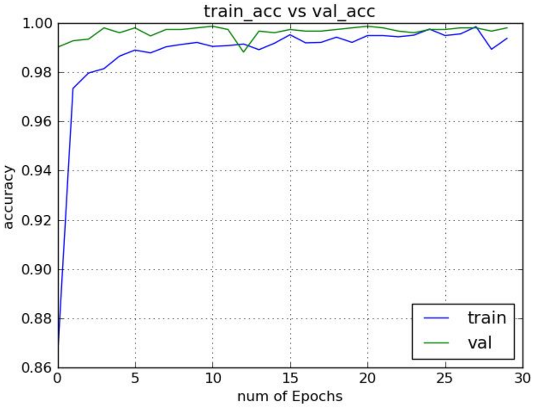
No. of examples	No. of epochs	Validation acc.	Test acc.
8297	20	99.52%	57.95%
3142	30	98.84%	52.88%

Here reason of high test accuracy may be that we are using test frames from same videos, so model is overfitting.



- Training with detected faces

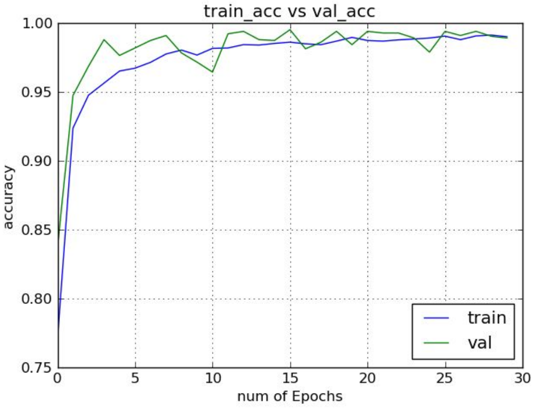
No. of examples	No. of epochs	Validation acc.	Test acc.
8297	30	99.80%	71.54%



3.5 Inceptionnet with adding avg pooling and two dense layers

- Using video frames as it is (face detection not applied):

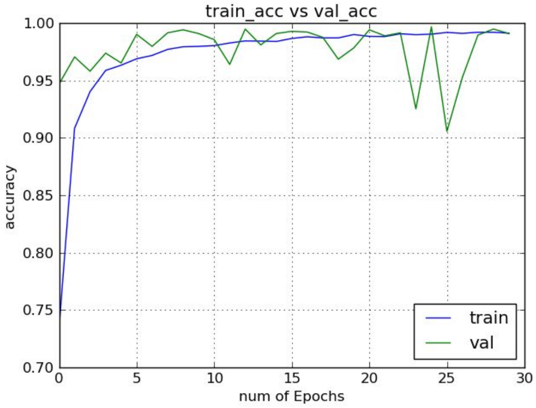
No. of examples	No. of epochs	Validation acc.	Test acc.
8297	30	99.52%	56.88%
8297	50	99.45%	61.40%
3142	30	98.54%	54.98%



Here reason of high test accuracy may be that we are using test frames from same videos, so model is overfitting.

- Training with detected faces

No. of examples	No. of epochs	Validation acc.	Test acc.
8297	30	99.08%	69.13%



4. Challenges faced:

- Faced problem while cropping the desired speaker (discarding audience or other person) when there are more than one person in a frame using **yolo,** because yolo detects all objects present.
- In case of one person in a frame,to identify whether it’s desired speaker or not.
- With large dataset during training, job was getting killed.

5. Conclusion:

- We got high accuracies when only classifier was trained in pretrained models.
- We got high accuracies when we train and test on cropped frames.
- For all results please check this: https://docs.google.com/document/d/1HOCt7g3Ti6NLR7eoBKQVBFyvM--CoPMYkvOLoohNsQk/edit?usp=sharing

Refrences:

1.https://www.superdatascience.com/opencv-face-detection/

2.Darknet: Open Source Neural Networks in C. https://pjreddie.com/darknet/

3. Building powerful image classification models using very little data. https://blog.keras.io/building-powerful-image-classification-models-using-very-little-data.html

4. Keras documentation https://keras.io/applications/

5.Exploring Neurons || Transfer Learning in Keras for custom data - VGG-16 https://www.youtube.com/watch?v=L7qjQu2ry2Q&t=18s https://github.com/anujshah1003/Transfer-Learning-in-keras---custom-data