Link: <https://github.com/pyannote/pyannote-video>

1. What is pyannote?

Ans - It is online github repo for face detection, tracking and clustering in videos. It also contains repo for audio as well as other media formats.

Doc Getting started file

2. How it does what it does?

Ans -

**Standard pipeline for Shot boundary**

**Shot boundary detection -----> Shot threading --------> Segmentation into scenes(frames)**

It starts with first defining the length of scene or video which is going to take as it’s basic unit which it calls as **Shot boundary selection**. You can define video frame height and width. There are other customization options which I do not understand right now e.g. threshold, min-match and lookahead.

Those boundaries(converted into json format) are further divided into frames. The length of the shot boundary will determine the number of frames per second or the total number of frames it is going to divide the video into.

**Standard pipeline for Face Recognition and Processing**

**Face tracking---------> Feature Extraction------->Face clustering**

They input the video, text(subtitles) as well as the audio(mp3) file. They also input the json file of tracking which we converted earlier.

They encode video into 64-encoding format to finally decode it into ascii format.

What is embedding in overall scenario?

After extracting the embedding - face track hierarchical agglomerative clustering is applied. Now, we are applying Face clustering - which is followed by making a timeline of when the faces appear into the whole video.

pyannote/\_init\_.py file and video file

What are trackers? What are trackers identifiers?

**Documentation for installation in ubuntu**

1. First, you need to download **get-pip.py** file from website- <https://bootstrap.pypa.io/get-pip.py>

2. Open your ubuntu terminal and go to the folder where your file is downloaded. I downloaded the get-pip.py file in **Downloads** folder. Go over to the folder from your root directory

3. Install pip:

sudo apt-get install python3-pip

The above command will install pip in the folder. sudo is used to override the firewall as it creates error in installing pip. Since, we are working on python 3.5x versions, we are setting pip in Python 3.5 folder in the system.

sudo update-alternatives --install /usr/bin/python python/usr/bin/python3 10

Setting default python version as 3.5x in ubuntu.

4. Installing virtual wrapper

sudo -H pip3 install virtualenvwrapper

We use the above command to install virtualenvwrapper package which provides tools for creating python virtual environments on Ubuntu Linux. Since, we are using python3.5x, we will always use pip3 in place of pip over all commands.

5. We need to add our environment variables in the newly created environment

sudo gedit ~/.bashrc

This command opens the bashrc file in your home directory.

export VIRTUALENVWRAPPER\_PYTHON=/usr/bin/python3.5  
export WORKON\_HOME=$HOME/.virtualenvs  
export PROJECT\_HOME=$HOME/python\_projects  
source /usr/local/bin/virtualenvwrapper.sh

Place these commands at the end of your bashrc file and save it. The above commands define the python version to be used in your virtual environment. You can set the directory of your project in project\_home

6. Creating a new virtual environment

mkvirtualenv pyannote

This creates virtual environment with name ‘pyannote’. This command also installs default packages into your virtual environment. You will now enter the ‘pyannote’ virtual environment.

Note: If you want to deactivate the virtual environment, you can use the command

deactivate

Note: If you want to again login into the virtual environment, you can use the command

workon pyannote

7. Install **pyannote-video** and it’s dependencies

pip3 install cmake

Pip3 install gcc

pip3 install -c menpo ffmpeg=3.1.3

pip3 install opencv

pip3 install boost

pip3 install pyannote-video

8. Downloading dlib models:

git clone https://github.com/pyannote/pyannote-data.git  
git clone https://github.com/davisking/dlib-models.git  
bunzip2 dlib-models/dlib\_face\_recognition\_resnet\_model\_v1.dat.bz2  
bunzip2 dlib-models/shape\_predictor\_68\_face\_landmarks.dat.bz2

9. Downloading videos and setting up local jupyter notebook

git clone https://github.com/pyannote/pyannote-data.git  
git clone https://github.com/pyannote/pyannote-video.git  
pip3 install jupyter  
jupyter notebook --notebook-dir="pyannote-video/doc"

**Installing through anaconda in ubuntu**

1. This command is to install anaconda in ubuntu.

<https://www.continuum.io/downloads>

First download the anaconda installer in your downloads folder. Install the Python 3.6x version and 64 bit.

Now, go to your ubuntu terminal and reach to your home directory

bash ~/Downloads/Anaconda3-4.4.0-Linux-x86\_64.sh

Execute the above command(this command assumes that your download is in ‘Downloads’ folder). Choose the default settings and say ‘Yes’ and press enter to all the default questions. Don’t change the home directory of anaconda.

Now, restart the terminal (this is important). This updates your bashrc file.

2. Creating a conda environment

conda create -n pyannote python=3.5 anaconda

source activate pyannote

3. Install pyannote-video and its dependencies:

conda install cmake

conda install gcc  
conda install -c menpo ffmpeg=3.1.3

conda install opencv  
conda install boost  
pip install pyannote-video

4. Download dlib models:

git clone https://github.com/pyannote/pyannote-data.git  
git clone https://github.com/davisking/dlib-models.git  
bunzip2 dlib-models/dlib\_face\_recognition\_resnet\_model\_v1.dat.bz2  
bunzip2 dlib-models/shape\_predictor\_68\_face\_landmarks.dat.bz2

## 

## 

5. To execute the "Getting started" notebook locally, download the example video and pyannote.video source code:

git clone https://github.com/pyannote/pyannote-video.git  
jupyter notebook --notebook-dir="pyannote-video/doc" --NotebookApp.iopub\_data\_rate\_limit=10000000000

Now, sequentially executing blocks by ‘Ctrl+Enter’

Note: If you will run the notebook in mozilla, you may get an error **“No Video with Supported Format and Mime Type Found”** . It does not support ogg format. So, run the notebook in Google Chrome.

6. In the ‘pylab inline’ block(first block) in jupyter notebook, update this block with

%pylab inline

import os

os.chdir(“/home/abhishek”)

!ls

You have to change the directory as per your pc name.

E.g. if your pc name is jarvis, then

os.chdir(“/home/jarvis”)