**Methods and Techniques**

We use the following models in our code. These models contain a convolutional neural network for face detection, age detection, and gender detection. These models are trained using the Adience Benchmark Datasets.

* opencv\_face\_detector.pbtxt
* opencv\_face\_detector\_uint8.pb
* age\_deploy.prototxt
* age\_net.caffemodel
* gender\_deploy.prototxt
* gender\_net.caffemodel

We use both live video and a picture as input. The picture undergoes image preprocessing before being introduced into the trained model, as the model is trained for a certain size, scale, alignment of image. After preprocessing the image runs through each of the networks for face, age, and gender detection. Only three convolutional layers and two fully-connected layers with a modest number of neurons make up the age and gender network.

The first two convolution layers are followed by a rectified linear operator (ReLU), a max-pooling layer, and a local response normalization layer each. The third convolution layer is followed by a rectified linear operator (ReLU) and a max-pooling layer. The first two fully-connected layers are followed by ReLu and a dropout layer. A soft-max layer receives the output of the last fully connected layer, which assigns a probability to each class. The prediction is made by selecting the class with the highest probability for the test image in question.

For face detection, we have a .pb file- this is a protobuf file (protocol buffer); it holds the graph definition and the trained weights of the model. We can use this to run the trained model. And while a .pb file holds the protobuf in binary format, one with the .pbtxt extension holds it in text format. These are TensorFlow files. For age and gender, the .prototxt files describe the network configuration and the .caffemodel file defines the internal states of the parameters of the layers. Both the caffe models (age\_net and gender\_net) simply define the internal states of the parameters/gradients of those layers.