

Artificial intelligence

Age Estimation and Gender Classification

Instructor: Dr. Rawan Ghnemat

By: Fadia ElMoubarak

20150109

## Problem Defenition

Two of the key facial attributes that play an important role in a person's day to day life is age and gender. Classifying the age and gender of a person from an unconstrained image is important in our time especially intelligent applications, such as human computer interaction. To achieve both age and gender detection and classification using convolutional neural networks. In deep learning, CNN, or ConvNet is a class of deep neural networks, most commonly applied to analyzing visual imagery. They are also known as shift invariant or space invariant artificial neural networks (SIANN), based on their shared-weights architecture and translation invariance characteristics.

A convolutional neural network consists of an input and an output layer, as well as multiple [hidden layers](https://en.wikipedia.org/wiki/Multilayer_perceptron#Layers). The hidden layers of a CNN typically consist of a series of convolutional layers that convolve with a multiplication or other [dot product](https://en.wikipedia.org/wiki/Dot_product). The activation function is commonly a [RELU layer](https://en.wikipedia.org/wiki/Rectifier_(neural_networks)), and is subsequently followed by additional convolutions such as pooling layers, fully connected layers and normalization layers, referred to as hidden layers because their inputs and outputs are masked by the activation function and final [convolution](https://en.wikipedia.org/wiki/Convolution).

Though the layers are colloquially referred to as convolutions, this is only by convention. Mathematically, it is technically a sliding dot product or [cross-correlation](https://en.wikipedia.org/wiki/Cross-correlation). This has significance for the indices in the matrix, in that it affects how weight is determined at a specific index point.

### 1.1. Gender Prediction

They have framed Gender Prediction as a classification problem. The output layer in the gender prediction network is of type softmax with 2 nodes indicating the two classes “Male” and “Female”.

### 1.2. Age Prediction

Age Prediction should be approached as a Regression problem since we are expecting a real number as the output. However, estimating age accurately using regression is challenging. Even humans cannot accurately predict the age based on looking at a person. However, we have an idea of whether they are in their 20s or in their 30s. Because of this reason, it is wise to frame this problem as a classification problem where we try to estimate the age group the person is in.

## Objectives:

* Build the first CNN for the gender classification.
* Build the second CNN for the estimate age identification.
* Integrate both CNNs.
* Test the results.

## Expected result:

The result that is expected from this project is to give the right gender and age of the inputted image with a good accuracy.

This project was created using the sequential type of model. This model is used to help efficiently create the layers one by one. Due to the Limited time frame that we had, we were not able to implement the Extreme Learning Machine part. In this project, it was implemented using two Convolutional Neural Networks as said before. The first Convolutional Neural Network is used to identify the gender of the person in the picture that was inputted through feature recognition. The second Convolutional Neural Network will be used to find the estimated age of the person in the picture.

# Description of implementation:

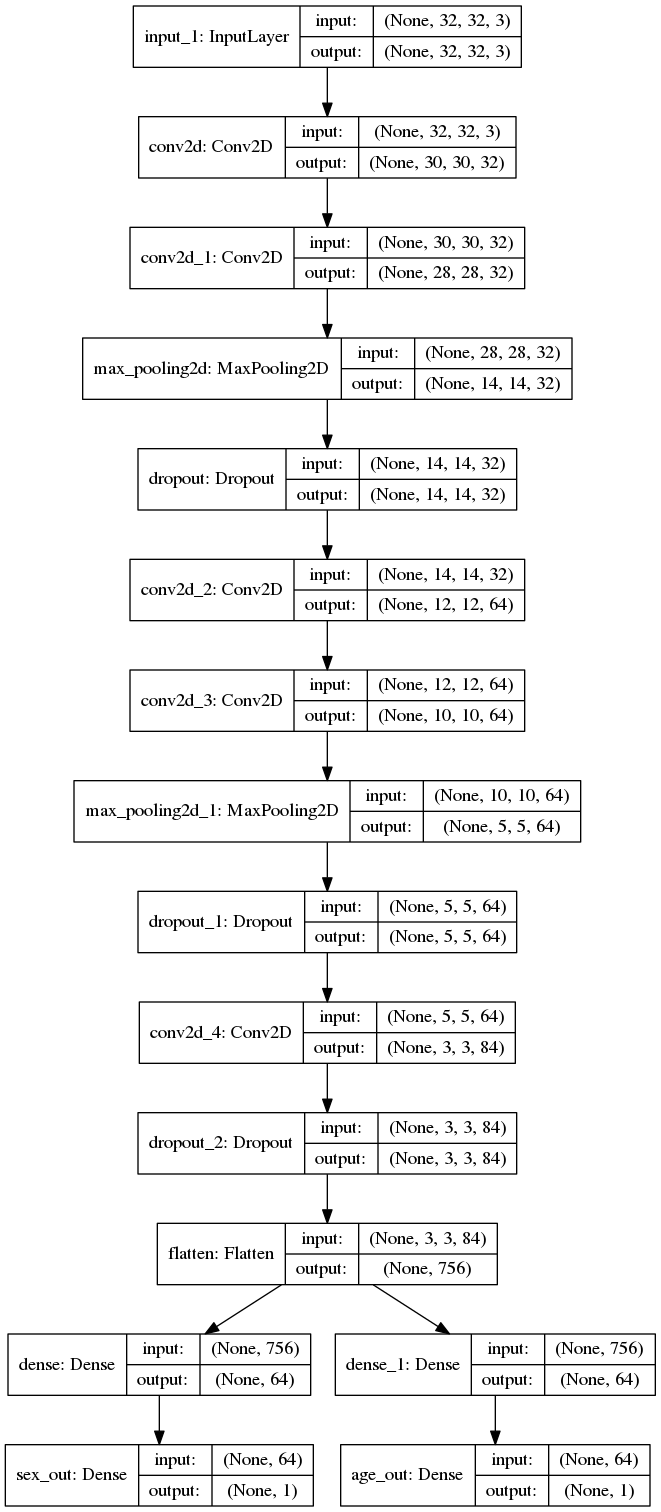
I took the database from UTKFace Kaggle data set, it is a big one with over than 23700 photos of people distributed randomly from both genders and ages from 1 up to 99. Each file has a unique name containing a shortcut to know its details, it is named like this: age\_gender\_modificationDatetime.jpg.

The gender is 1 for females and 0 for males. the ages are split into 4 categorize, 0-18, 19-30, 31-80, 81++, and a 5th category for anything unknown.

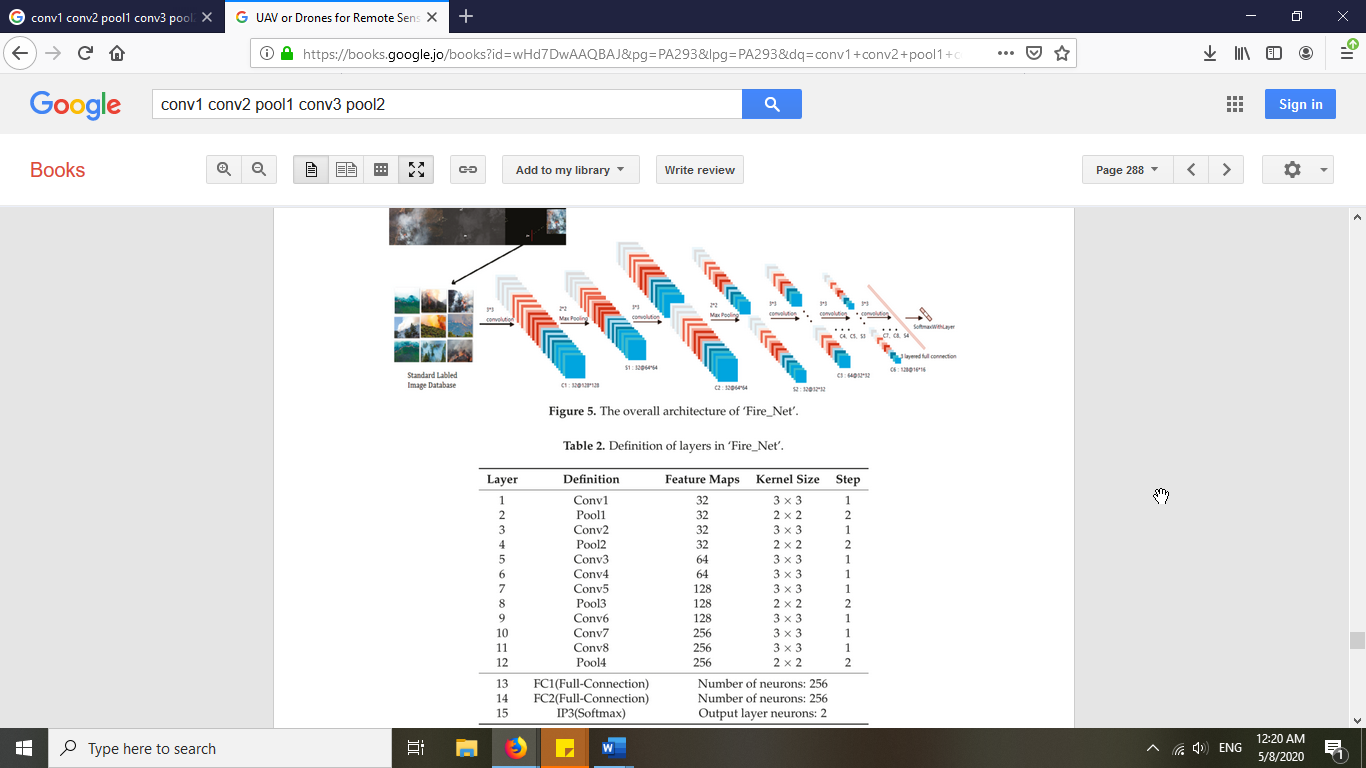
I resized each photo to 64\*64, and split the data 20/80 for testing and training.

The best number of epochs = 100, and batch\_size = 128. The activation = 'relu' but the last layer in the gender is 'sigmoid’, the optimizer = 'adam', the loss = ‘mean\_squared\_error’for the age and ‘binary\_crossentropy’ for the gender. Finally, the metrics = 'accuracy'. All of this combination to give me the best possible accuracy = 89%, which is really good.

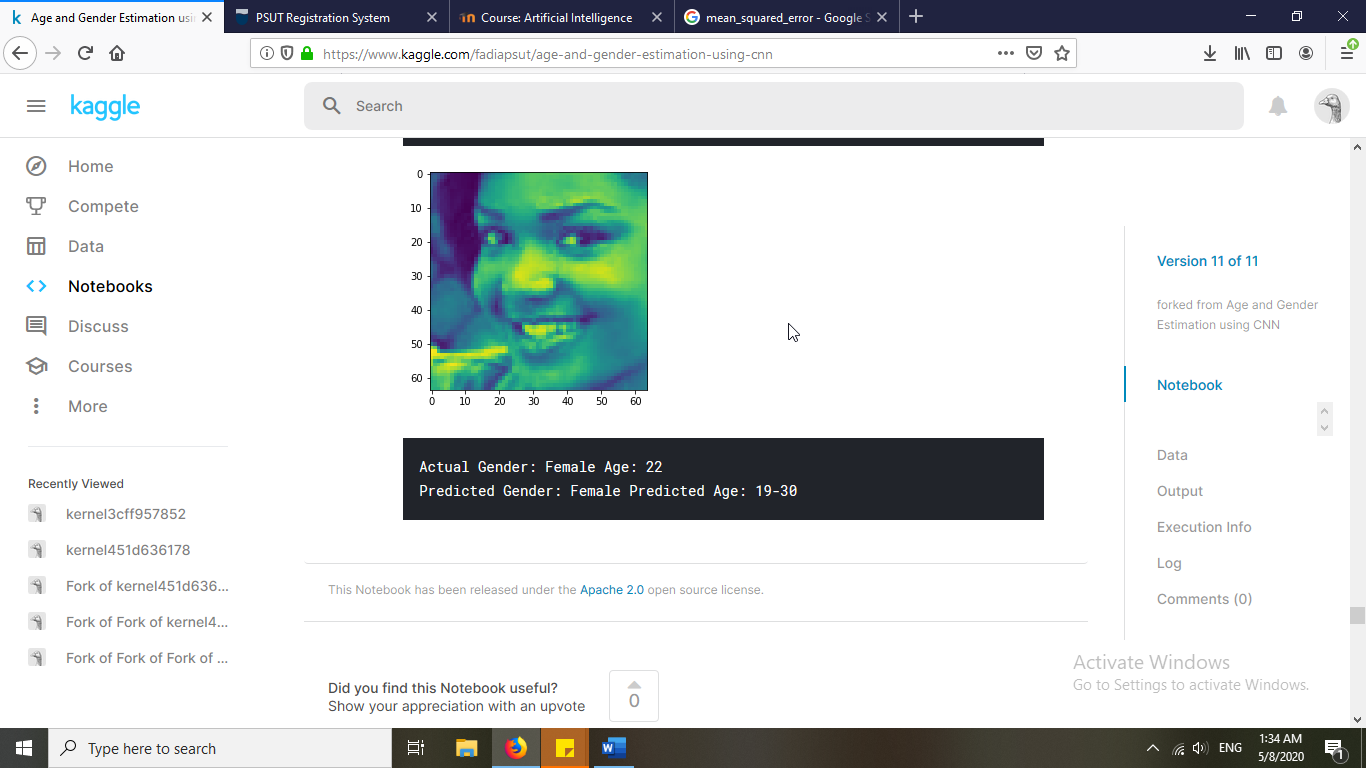
The previous model looked like this:

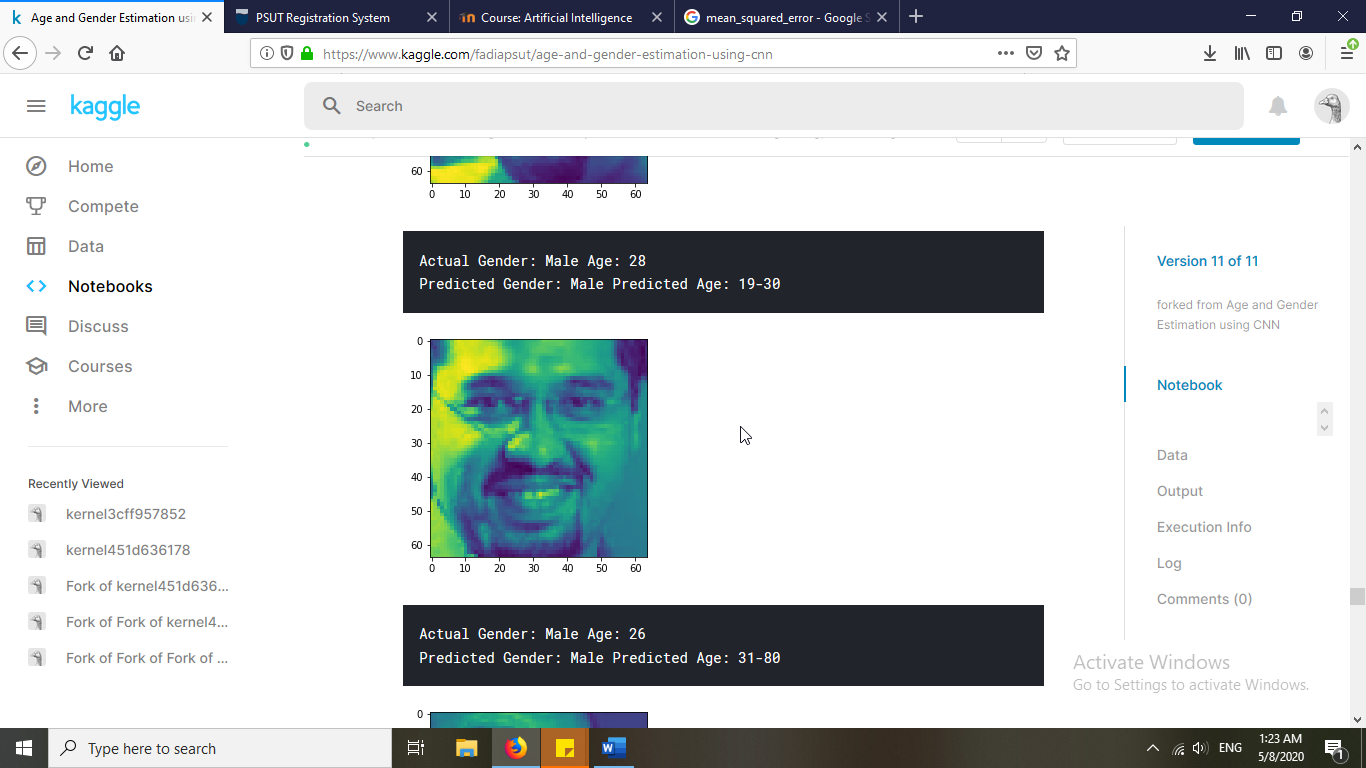


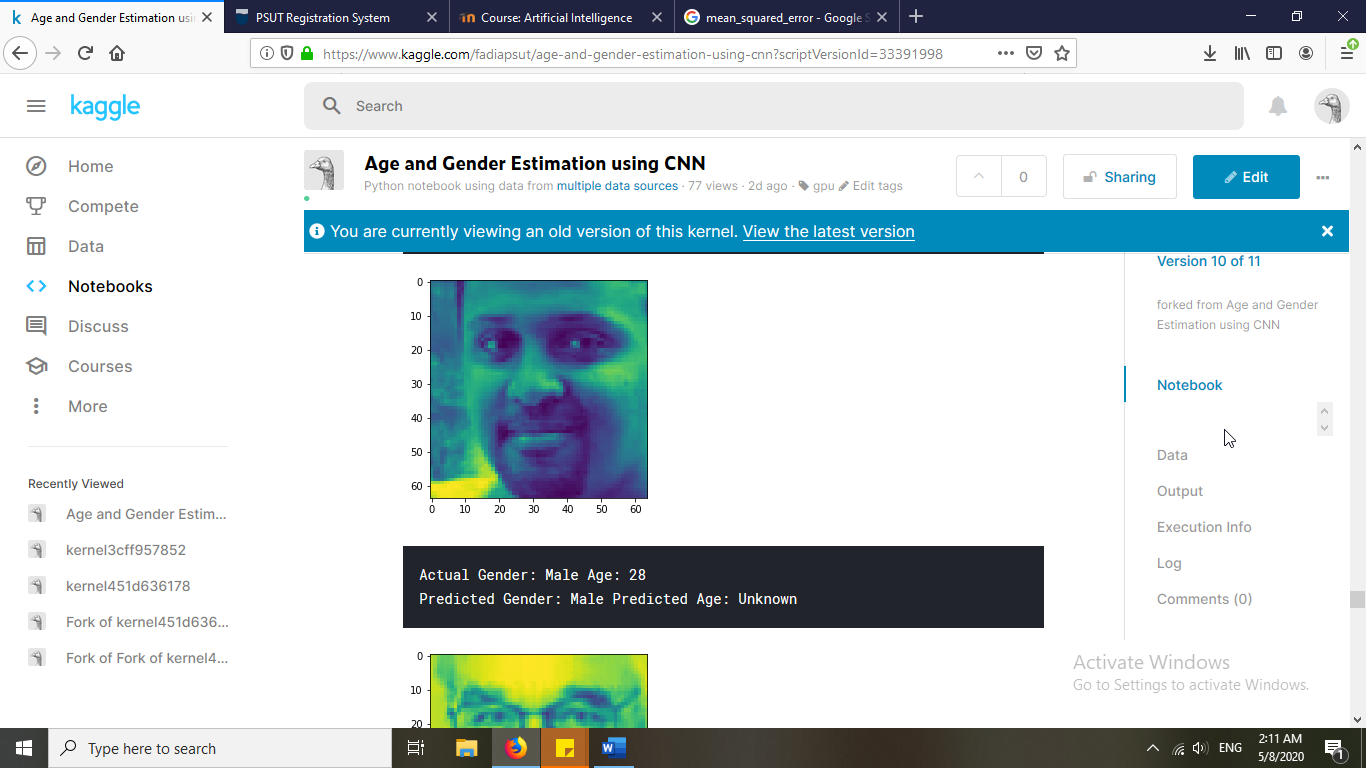
I used a model similar to the one used in the fire alarms, to detect the images of a smoke or fire.



1. **Test results**  
   Most of the time, the results are true (89%), but when it is wrong, it is mainly for elderly or babies especially males, and sometimes it is unknown (instead of wrong).   
   Here is a good and a wrong result:







<https://www.kaggle.com/fadiapsut/age-and-gender-estimation-using-cnn?scriptVersionId=33392691>

# Similar documentations

Download paper here: <https://towardsdatascience.com/predict-age-and-gender-using-convolutional-neural-network-and-opencv-fd90390e3ce6>

# References

* Definition of Convolutional neural network  
  <https://en.wikipedia.org/wiki/Convolutional_neural_network>
* Age and Gender Classification Using Python  
  <https://www.learnopencv.com/age-gender-classification-using-opencv-deep-learning-c-python/>
* Age and Gender Classification Using Wide Convolutional Neural Network and Gabor Filter   
  <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.722.9654&rep=rep1&type=pdf>
* Predict Age and Gender using Convolutional Neural Network   
  <https://towardsdatascience.com/predict-age-and-gender-using-convolutional-neural-network-and-opencv-fd90390e3ce6>
* Remote sensing applications book  
  <https://books.google.jo/books?id=wHd7DwAAQBAJ&pg=PA293&lpg=PA293&dq=conv1+conv2+pool1+conv3+pool2&source=bl&ots=6OFIrC5-S7&sig=ACfU3U0GkZzkF7UazCizZf48lIKvJiGkNQ&hl=en&sa=X&ved=2ahUKEwihi-OE1aLpAhUx6KYKHahkAxYQ6AEwAHoECAYQAQ#v=onepage&q&f=true>