**Chapter 3**

**SYSTEM ANALYSIS AND** **METHODOLOGY**

**3.1 Introduction**

This chapter summarizes the procedures of doing research on the proposed system which is age and gender estimation methods. The main purpose of this work is to develop a framework of Age and gender prediction with deep convolutional neural network with OpenCV.

**3.2 Description of the existing system**

The existing system is a python program without GUI and it is not stating what is precisely the age and gender to a lay man. And in order to run it, you should have the knowledge of running it on the terminal. (Which means all the activities of training, testing and the evaluation of the program must be done in the in the terminal). Another drawback of the existing system is; every time we need to run it, the network needs to be retrained which will takes at least 2 hours of our time. That why we come up with training the network with the *Caffe model framework* which we train the CNN once and run it many times.

**3.3 Analysis of the existing system**

**3.3.1 Existing method of age classification.**

In recent years, many methods have been provided in the problem of automatically extracting age related attributes from facial images. A detailed survey of such methods of such methods has been presented in (Y. Fu et. al., 2010) and more recently in (H. Han et. Al, 2013).

Early methods for age estimation are based on calculating ratios between different measurement of facial features. Once a facial features (e.g. eye, nose, mouth, chin, etc.) are localized and their sizes and distances measured, ratios between them are calculated and for classifying the face into different age categories according to he hand-crafted rule(Y. H. Kwon et. al., 1994).

More recently, (N. Ramanathan and R. Chellappa, 2006) uses a similar approach to model age progression in subject under 18 years old. As the method require accurate localization (finding location of face feature such as nose) of facial features.

**3.3.2 Existing method of Gender classification**

A detailed survey of existing methods of gender classification has been provided in (Makinen and R. Raisamo, 2008), and more recently in (D. Reid et. al., 2013).

One of the early methods for gender classification used a neural network trained on a small set of near-frontal face images (B. A. Golomb et al., 1990).

A. J. O’toole et. al. uses the combined 3D structure of the head (obtained using a laser scanner) and image intensities were used for classifying gender ( A. J. O’toole et al., 1997).

**3.4 Problems of the existing system**

Most of the methods discussed above used the FERET Dataset ( P. J. Phillips et. al. both to develop the proposed systems and to evaluate performances. FERET images were taken under highly controlled condition. Moreover, the results obtained on this Dataset suggest that it is saturated and not challenging for modern methods. It is therefore difficult to estimate the actual relative benefit of these techniques. As a consequence, (C. Shan., 2012) experimented on the popular *Labeled Faces in the Wild (LFW) (G. B. Huang et al., 2007)* Dataset, primarily used for face recognition.

**3.5 Description of the propose system**

As with age estimation, here too, we focus on the Adience dataset which contains images more challenging than those provided by *Labeled Faces in the Wild (LFW*) *(G. B. Huang et al., 2007)*, reporting performance using a more robust system, designed to better exploit information from massive example training sets.

**3.6 Advantages of the** **proposed system over the existing system**

Some of the advantages of the proposed system includes the following:

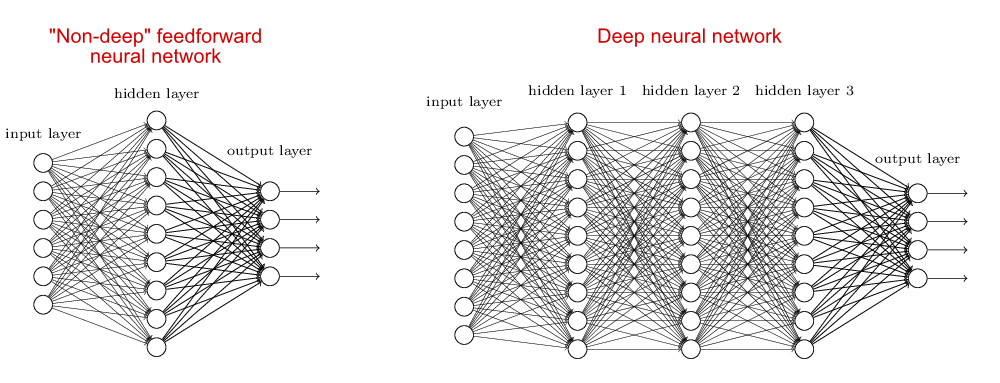
* The design of the Graphical User Interface GUI for the proposed system
* It can be able to insert image and predict it with a click.
* It eliminates the burden of writing script on the terminal to estimate age and gender.

**3.7 Methodology**

The study begun with a review of related literature on the development of Age and Gender for developing the proposed system using image recognition technology. The literature research was accomplished through Google Scholar, Institute of Electrical and Electronics Engineers (IEEE). And the related works and researches done in recent years regarding the application of DL in image were studied in order to get an idea of developing the Age and Gender detection system. *Python* Programming language was chosen as software platform because of its flexibility and capability to support different DL packages.

**3.7.1 Difference Between Neural Networks & Deep Neural Networks**

In [machine learning](https://en.wikipedia.org/wiki/Machine_learning), a **convolutional neural network** (**CNN**, or **ConvNet**) is a class of deep,[feed-forward](https://en.wikipedia.org/wiki/Feedforward_neural_network) [artificial neural network](https://en.wikipedia.org/wiki/Artificial_neural_network) that have successfully been applied to analyzing visual imagery. CNNs use a variation of [multilayer perceptrons](https://en.wikipedia.org/wiki/Multilayer_perceptron) designed to require minimal [preprocessing](https://en.wikipedia.org/wiki/Data_pre-processing). They are also known as **shift invariant** or **space invariant artificial neural networks** (**SIANN**), based on their shared-weights architecture and translation invariance characteristics.



*figure 3.1. Showing differences between non-deep NN and Deep NN*

Convolutional networks were [inspired](https://en.wikipedia.org/wiki/Mathematical_biology) by [biological](https://en.wikipedia.org/wiki/Biological) processesin which the connectivity pattern between [neurons](https://en.wikipedia.org/wiki/Artificial_neuron) is inspired by the organization. Individual [neurons](https://en.wikipedia.org/wiki/Cortical_neuron) respond to stimuli only in a restricted region of the [visual field](https://en.wikipedia.org/wiki/Visual_field). CNNs use relatively little pre-processing compared to other image classification algorithms. This means that the network learns the filters that in traditional algorithms were hand-engineered. This independence from prior knowledge and human effort in feature design is a major advantage. They have applications in [image and video recognition](https://en.wikipedia.org/wiki/Computer_vision), [recommender systems](https://en.wikipedia.org/wiki/Recommender_system) and [natural language processing](https://en.wikipedia.org/wiki/Natural_language_processing).

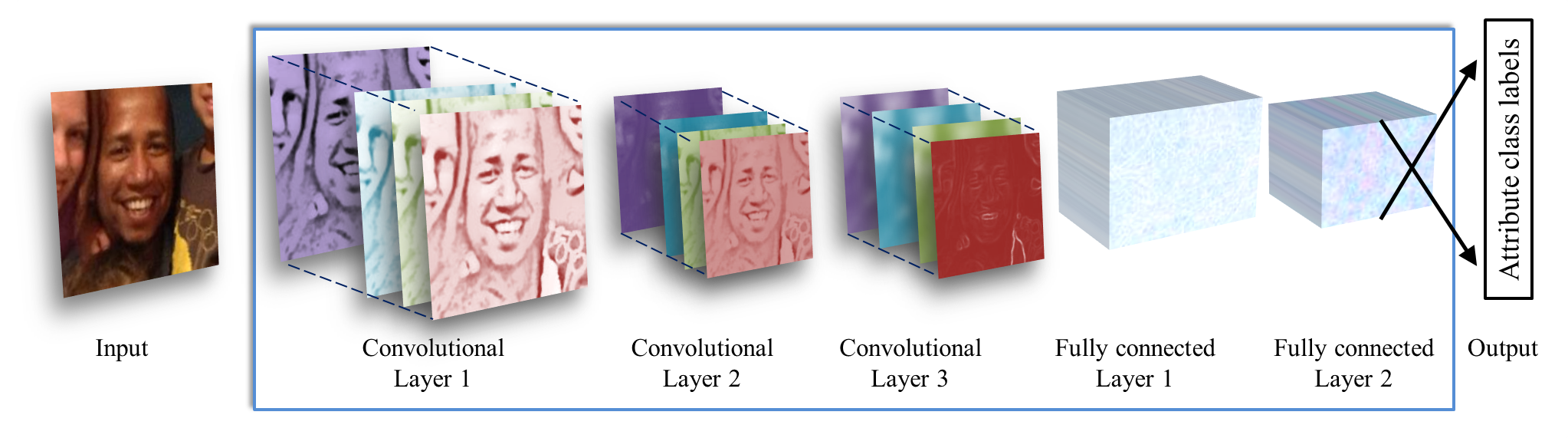
Deep learning is a class of [machine learning](https://en.wikipedia.org/wiki/Machine_learning) [algorithms](https://en.wikipedia.org/wiki/Algorithm) that use a cascade of many layers of [nonlinear processing](https://en.wikipedia.org/wiki/Nonlinear_filter) units for [feature extraction](https://en.wikipedia.org/wiki/Feature_extraction) and transformation. Each successive layer uses the output from the previous layer as input. The algorithms may be [supervised](https://en.wikipedia.org/wiki/Supervised_learning) or [unsupervised](https://en.wikipedia.org/wiki/Unsupervised_learning) and applications include pattern analysis and classification are based on the learning of multiple levels of features or representations of the data, are part of the broader machine learning field of learning representations of data, learn multiple levels of representations that correspond to different levels of abstraction; the levels form a hierarchy of concepts.

These definitions have in common multiple layers of nonlinear processing units and the supervised or unsupervised learning of feature representations in each layer, with the layers forming a hierarchy from low-level to high-level features. The composition of a layer of nonlinear processing units used in a deep learning algorithm depends on the problem to be solved. Layers that have been used in deep learning include hidden layers of an [artificial neural network](https://en.wikipedia.org/wiki/Artificial_neural_network) and sets of complicated [formulas](https://en.wikipedia.org/wiki/Propositional_formula).

At each layer, the signal is transformed by a processing unit, like an artificial neuron, whose parameters are iteratively adjusted through training.

**3.7.2 Network architecture**

The network architecture is used throughout our experiments for both age and gender classification it is illustrated in Figure below. The network comprises of only three convolutional layers and two fully-connected layers with a small number of neurons. Age classification on the Adience set requires distinguishing between eight classes; gender only two. This, compared to, e.g., the ten thousand identity classes used to train the network used for face recognition in (Y. Sun. et al, 2014).

figure 3.2. *Illustration of our CNN architecture. The network contains three convolutional layers, each followed by a rectified linear operation and pooling layer. The first two layers also follow normalization using local response normalization (A. Krizhevsky et al., 2012). The first Convolutional Layer contains 96 filters of 7×7 pixels, the second Convolutional Layer contains 256 filters of 5×5 pixels, The third and final Convolutional Layer contains 384 filters of 3 × 3 pixels. Finally, two fully-connected layers are added, each containing 512 neurons. See Figure 2 for a detailed schematic view.*

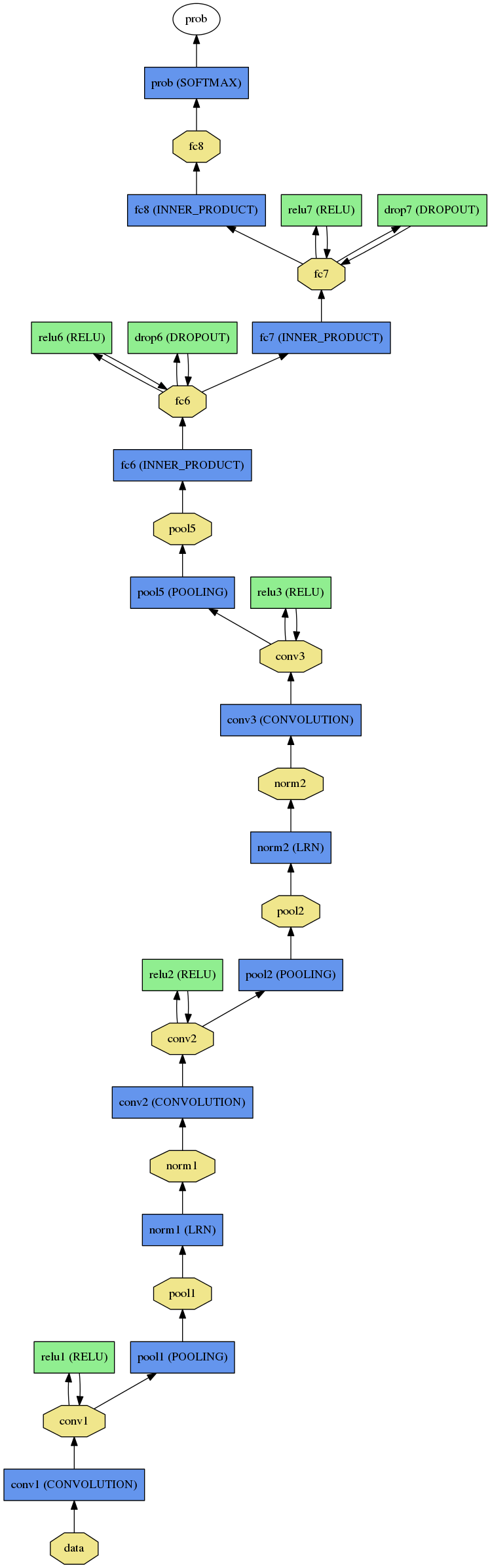
All three color channels are processed directly by the network. Images are first re-scaled to 256 × 256 and a crop of 227 × 227 is fed to the network. The three subsequent convolutional layers are then defined as follows.

1. 96 filters of size 3×7×7 pixels are applied to the input in the first convolutional layer, followed by a rectified linear operator (ReLU), a max pooling layer taking the maximal value of 3 × 3 regions with two-pixel strides and a local response normalization layer *(A. Krizhevsky et al., 2012).*
2. The 96 × 28 × 28 output of the previous layer is then processed by the second convolutional layer, containing 256 filters of size 96 × 5 × 5 pixels. Again, this is followed by ReLU, a max pooling layer and a local response normalization layer with the same hyper parameters as before.
3. Finally, the third and last convolutional layer operates on the 256 × 14 × 14 blob by applying a set of 384 filters of size 256 × 3 × 3 pixels, followed by ReLU and a max pooling layer.

The following fully connected layers are then defined by:

1. A first fully connected layer that receives the output of the third convolutional layer and contains 512 neurons, followed by a ReLU and a dropout layer.
2. A second fully connected layer that receives the 512-dimensional output of the first fully connected layer and again contains 512 neurons, followed by a ReLU and a dropout layer.
3. A third, fully connected layer which maps to the final classes for age or gender.

Finally, the output of the last fully connected layer is fed to a soft-max layer that assigns a probability for each class. The prediction itself is made by taking the class with the maximal probability for the given test image.



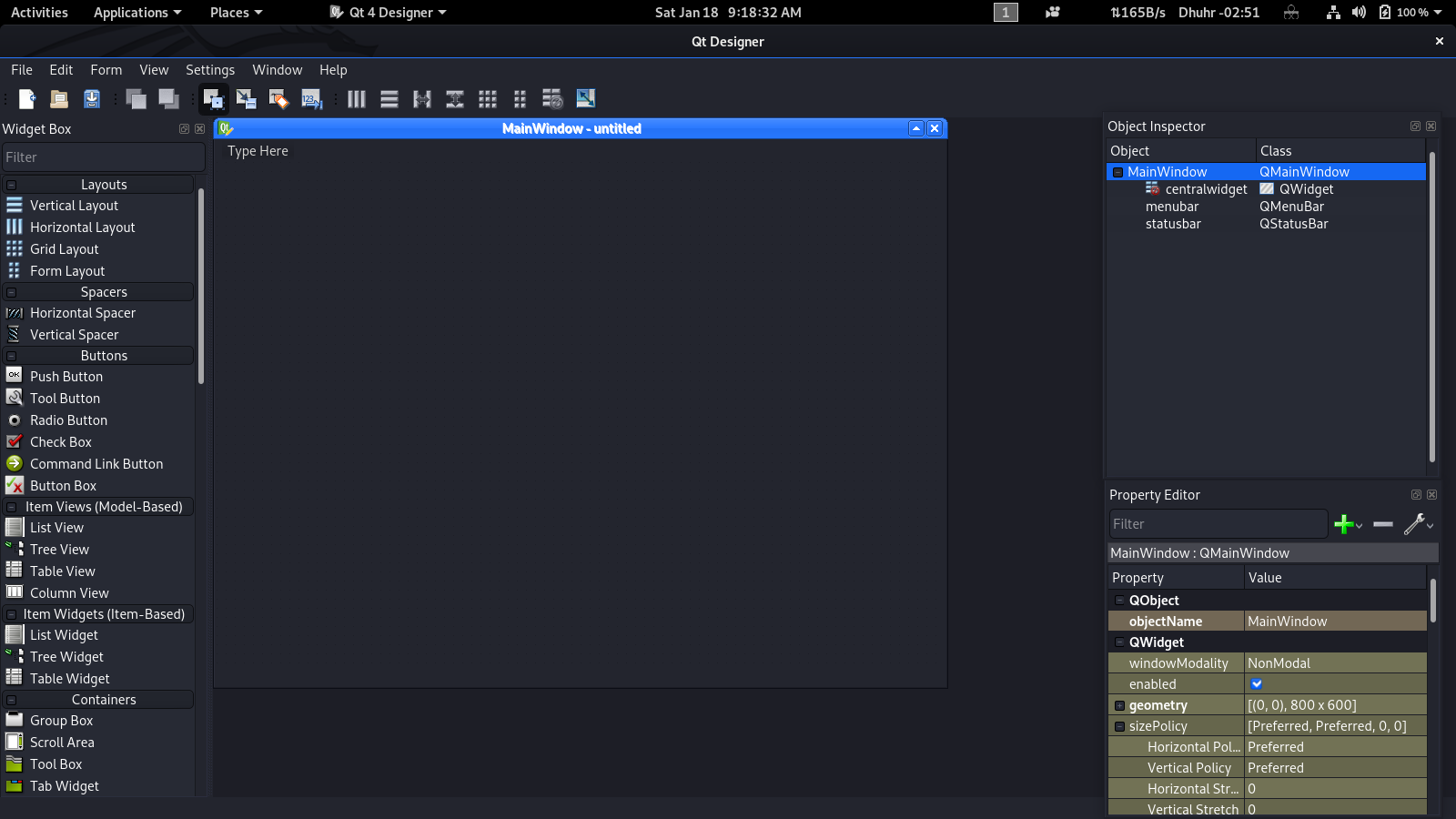
*Figure 3.3 Full schematic diagram of our network architecture. Please see text under Figure 2. for more details.*

**3.7.3 PyQT5 for GUI Design**

Using Qt **Designer**. Qt **Designer** is the Qt tool for designing and building graphical user interfaces. It allows us to design widgets, dialogs or complete main windows using on-screen forms and a simple drag-and-drop interface.

PyQt is a python binding of the open-source widget-toolkit Qt, which also functions as a cross-platform application development framework. Qt is a popular C++ framework for writing GUI applications for all major desktop, mobile, and embedded platforms (supports Linux, Windows, MacOS, Android, iOS, Raspberry Pi, and more).

PyQt is developed and maintained by [Riverbank Computing](https://riverbankcomputing.com/news), a company based in England, whereas Qt is developed by a Finnish firm called The [Qt Company](http://www.qt.io/company).

*Figure 3.4 Qt Designer Main Window Home Screen*

**3.8 Method of data collection**

**Data collection**: Data collection plays an important role in training any deep neural network (DNN). In this project, the aim was to label data for two separate tasks: age and gender estimation. Collecting labeled data for some tasks, such as real age estimation, is much more challenging compared to popular classification or detection problems. This disparity is due to the fact that human error in estimating real age is large (sometimes greater than the computer vision estimations) and one cannot rely on human annotators to label faces with their corresponding real age. However, this project is done with a large dataset of faces with their corresponding age and gender labels. To our knowledge, this dataset is the largest or among the largest in either the academic or commercial world.

**3.8.1 ADIENCE DATASET**

1. Adience Database is large scale dataset, which was collected to capture all the variations in appearance, noise, pose, lighting and more, and can be expected to serve as images taken without careful preparation or posing. The Adience set consist of images automatically uploaded to flickr from smartphone devices. These images were uploaded without prior manual filtering. The entire Adience collection includes roughly 20k images of 2284 subjects. However, this database doesn’t have the accurate age annotation, i.e. age is annotated in a range form like 0-2, 4-6, 8-13, 15-20, 25-32, 38-43, 48-53, 60+. (E. Eidinger et al, 2014)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **0-2** | **4-6** | **8-13** | **15-20** | **25-32** | **38-43** | **48-53** | **60+** | **Total** |
| **Male** | 745 | 928 | 934 | 734 | 2308 | 1294 | 392 | 442 | 8192 |
| **Female** | 682 | 1234 | 1360 | 919 | 2509 | 1056 | 433 | 427 | 9411 |
| **Both** | 1427 | 2162 | 2294 | 1653 | 4897 | 2350 | 825 | 869 | 19487 |

**Table 3.1** Adience image dataset distribution (number of images for each gender and age range)

**3.8.2 Dataset pre-processing**

The dataset pre-processing step included converting human readable data, such as images into computer readable binary data. In this project, the images were converted to **Caffe Model** format data as the input for training and testing process.

**3.9 Caffe Model**

*Caffe (Convolutional Architecture for Fast Feature Embedding) model* is a deep learning framework made with expression, speed, and modularity in mind. It is developed by Berkeley AI Research ([BAIR](http://bair.berkeley.edu/)) and by community contributors.We choose caffe in this project because of the following:

1. Speed: makes Caffe perfect for research experiments and industry deployment. Caffe can process over 60M images per day with a single NVIDIA K40 GPU\*. That’s 1 ms/image for inference and 4 ms/image for learning and more recent library versions and hardware are faster still. We believe that Caffe is among the fastest *ConvNet* implementations available.
2. Expressive architecture: Which encourages application and innovation. Models and optimization are defined by configuration without hard-coding.
3. Extensible code: fosters active development. In Caffe’s first year, it has been forked by over 1,000 developers on [github](https://github.com/BVLC/caffe) and had many significant changes contributed back.
4. Community: Caffe already powers academic research projects, startup prototypes, and even large-scale industrial applications in vision, speech, and multimedia.

This project method is implemented using the Caffe open-source framework the training was performed based on (Y. Jia. Et al., 2014). Prediction running times can conceivably be substantially improved by running the network on image batches.