EECS 432: Final Project

Human (Face) Detection and Recognition

By

Chenyang Meng

Department of Mechanical Engineering

March 21, 2018

Contents

Abstract

With crimes on the rise all around the world, video surveillance is becoming more and more important at present. Due to the lack of human resources to monitor the increasing number of surveillance cameras manually, computer vision algorithms are applied to teach the computer extracting the useful information that people want, such as detecting and identifying people and vehicles. With such help of the computer, people do not have to put efforts in the monitoring and searching. In this project, a program is developed to detect people and recognize them. Algorithms such Histogram of Gradients, Support Vector Machine, non-maximum suppression, Haar-like features, cascade classifiers, and Local Binary Pattern Histogram are applied to achieve the goal of tracking and recognizing people in a video or a sequence of videos. The results are not perfect, and improvements are necessary for future development.

1 Introduction

Computer vision is becoming important day by day with that human detection and recognition for application such as identifying people under video surveillance. Nowadays, many closed-circuit television cameras are applied in the public places such as bank, hotel, and etc. for the safety purposes. Though people do not have to monitor the area with the help of surveillance cameras, they still have to spend time trying to identify someone in the video. And sometimes this could cost a huge amount of time to search by people and results are not good. If a computer could identify a person by itself, it will save lots of human efforts.

In this project, a program is developed which can detect a person and his/her face in a given video and then recognize him/her by using features in the pre-trained model of that person. If the person is detected in the video for the first time, a label number will be assigned to that person. And that person can be tracked and recognized in any other given videos. Conventional algorithms and methods are used in the project. Histogram of Oriented Gradients (HOG) is a robust algorithm for object detection with help of a support vector machine classification (SVM). And non-maximum suppression is applied to select the best results of detection. Haar features and cascade classifiers are used for face detection. Local Binary Pattern Histogram (LBPH) features from pre-trained models are used to recognize people, which is considered as the new method used in this project for face recognition.

The improvements are needed since the results are not 100% perfect. There are still many false positive cases occurs in the videos, either on body or face. The computer cannot detect a person's body if he/she presents only parts of the body in the videos. The computer cannot recognize side faces when people are passing by. The computer still needs database to recognize people.

2 Problem Definition

The input of the program is a video or a sequence of videos. The main objective of the project is to detect and recognize people and track them in videos. There are some constrains of the project when computer provides results. A computer can successfully detect a person for the most cases but only recognize a person who has pre-trained LBPH feature model. If a person is not pre-trained to create a database, he/she will not be recognized by computer.

There are some assumptions of the project too. In most of cases, a person is facing the camera, so his/her frontal face is successfully detected and recognized by computer.

There are also some challenges of the project. First, detecting a person's body by presenting only parts of the body is a great challenge. Also, detecting and recognizing a person's side face is still a challenge. Last but not least, reducing the false positive cases remains is a challenge.

3 Methodology

There are two parts in this section, including approaches to the project and algorithms used in the project. Section 2.1 gives both overview and details on approaches to the project and how the program works. Section 2.2 provides the algorithms used in the programs, which LBPH is a new method used for face recognition.

3.1 Approaches

Figure 1 and Figure 2 illustrate the approaches to the project. Figure 1 shows the flow chart of the overview of the project.

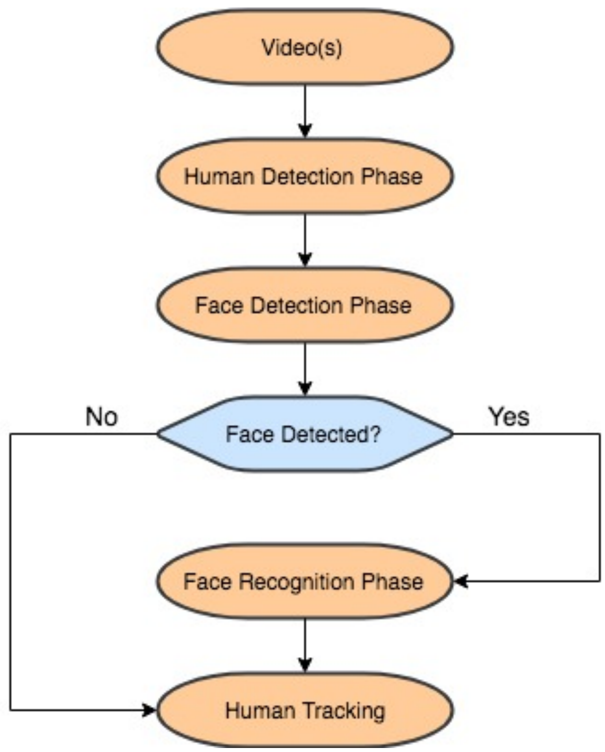

Figure 1. Overview of the Project.

The input to the program is a video or sequence of videos. There are three phases, including human detection phase, face detection phase, and face recognition phase. After receiving a video or sequence of videos, the program executes the human detection phase firstly and the face detection phase secondly. If the face is detected in the face detection phase, then the program will execute the face recognition phase and track the human. If the face is not detected, the program will just track the human.

Figure 2 demonstrates the data flow chart, which gives details about how the program works.

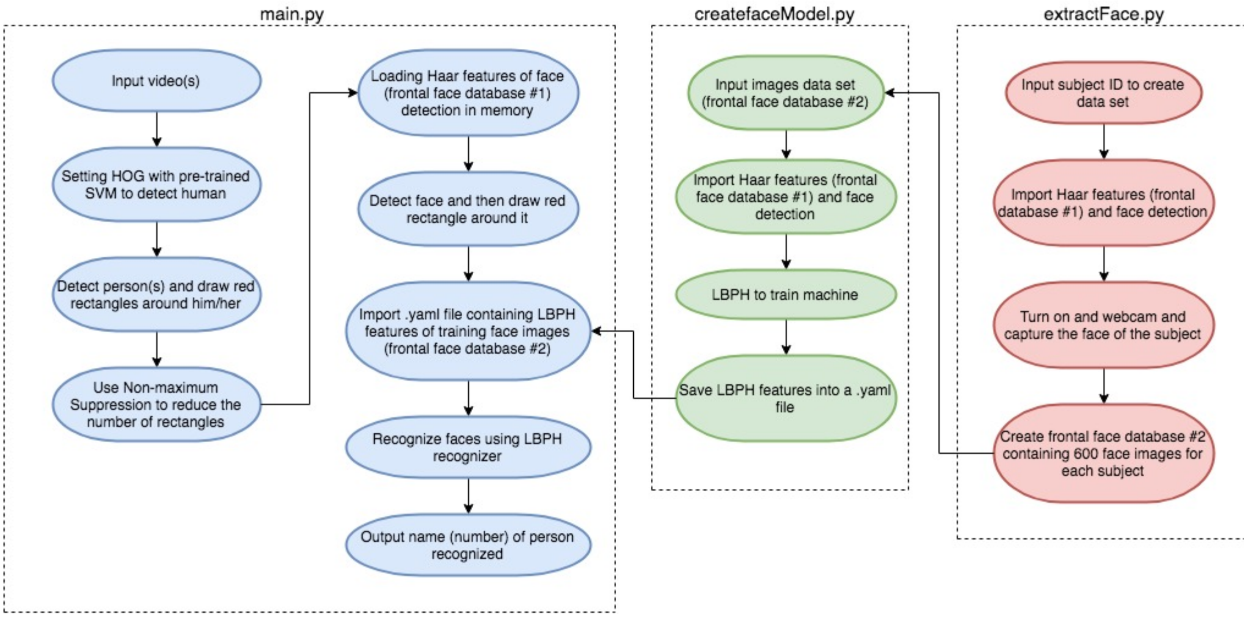

Figure 2. Details about How the Program Works.

There are three major stages in the project, including extracting face stage, creating face model stage, and main detecting and recognizing human stage. In the extracting face stage (extractFace.py in Figure 2), database of 600 sample images is taken for each subject (person). The input is subject ID to create a dataset to corresponding subject. The program uses Haar features of frontal face from OpenCV library to detect subject face and turns on the webcam to capture the sample images of the subject. In the creating face model stage (createfaceModel.py in Figure 2), input is database of the subject sample images. The program uses Haar features of frontal face to detect face and Local Binary Pater Histogram (LBPH) to train the machine. At last, the program saves LBPH features into a .yaml file, which is the matrix model used in the main detecting and recognizing human stage. In the main detecting and recognizing human stage (main.py), input is a video or a sequence of videos. The program sets the Histogram of Gradients (HOG) with pre-trained Support Vector Machine (SVM) to detect a person. After detecting the person, the program draws red rectangles around him/her. The program uses non-maximum suppression to reduce the number of rectangles and leave only the best one. After body detection, the program loads the Haar features of the frontal face to detect face of the person and draw the green rectangle around the face. After the face detection, the program imports the matrix model of LBPH features to recognize the face and puts the label above the face recognized.

3.2 Algorithms

3.2.1 Histogram of Gradients (HOG) [1].

The Histogram of Gradients (HOG) is a feature descriptor used in computer vision and image processing for object detection. The HOG descriptor can be described by the distribution of intensity gradients in an image. The image is divided into blocks with overlap. And each block should consist cells with pixels in each cell. The gradient orientation is quantized into 9 bins (0 degree to 180 degree). The descriptor contains histogram of gradient of every pixel. HOG is contrast-normalized by calculating intensity over blocks to provide better accuracy. And this value is used to normalize all the cells within the block. The normalized results provide a better performance on variation in illumination and intensity. HOG descriptor has several advantages over some other descriptors. Since it operates on the local cells, HOG descriptor is invariant to geometric and photometric transformation, except for object orientations. Also, coarse spatial sampling, fine orientation sampling, and strong local photometric normalization permits the individual body movement of pedestrians to be ignored so long as they maintain a roughly upright position. Thus, the HOG descriptor is robust for the human detection.

3.2.2 Support Vector Machine (SVM) [2]

Support Vector Machine (SVM) is supervised learning models with associated learning algorithms that can analyze data used for classification and regression. SVM can efficiently perform on linear and non-linear classification. When it is not possible to classify data linearly in the same dimensional space, SVM approaches into higher dimensional space so that the classification can be done linearly. As Figure 3(a) shown, the data to be classified is non-linear. In order to classify data linearly, the SVM approaches to higher dimensional space, as Figure 3(b) shown. For the linear classification, SVM constructs a hyperplane or a set of hyperplanes in high or infinite-dimensional spaces. Linear classification is easier and faster in SVM, comparing to non-linear classification. And SVM is used as combination with HOG mentioned in previous section.

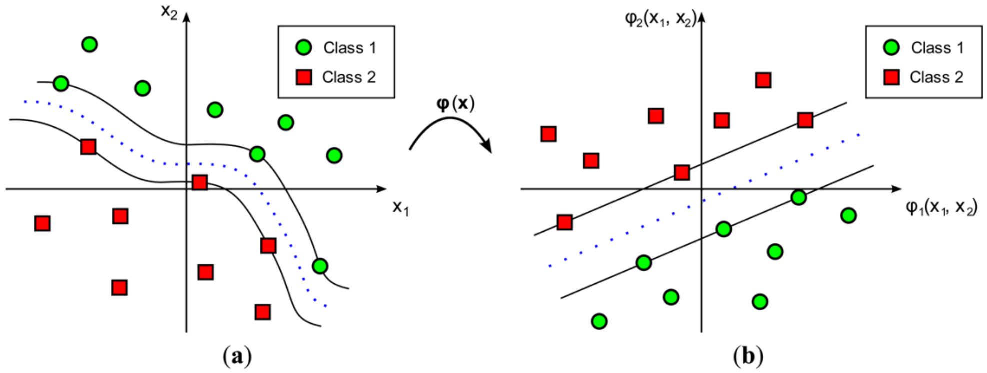

Figure 3. Representation of a Support Vector Machine classifier with a nonlinear kernel. Function φ(·) is the nonlinear transformation mapping vectors from (a) the input space to (b) the feature space [2].

3.2.3 Haar Cascade [3]

Haar cascade includes Haar-like features and cascade classifiers. Haar-like features are image features used in object recognition. Haar-features are extracted by using sliding windows of rectangular blocks. These features are single valued and are calculated by subtracting the sum of pixel intensities under the white rectangles from the black rectangles, as shown in Figure 4. Methods of integral image and AdaBoost are used in extracting Haar-like features to reduce the computation time.

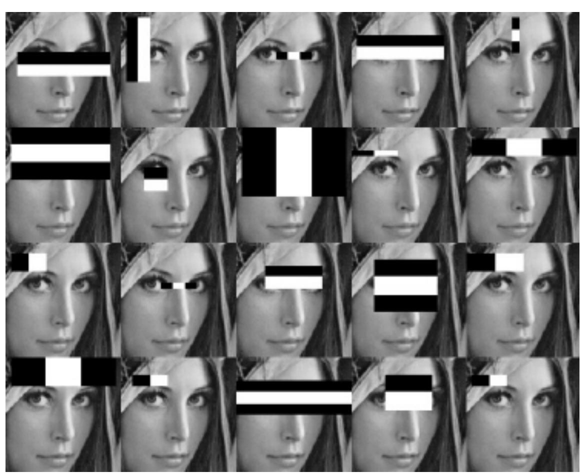

Figure 4. Haar-like Features Examples for Face Detection [3].

Haar-like features are used for object detection. In the detection phase, a window is moved over the input image and for each sub-section, Haar-like features are calculated. This difference is then compared with a learned threshold value to detect the presence of an object. In order to accurately detect the object, a large number of Haar-like features are required. This means that a large number of positive and negative samples are needed for this algorithm. The positive samples have the presence of the object, and negative samples do not have the presence of the object. The features cannot detect object on their own, but they form a strong classifier when they are used together. When a computer tries to detect a face in an image, most region of the image is non-face. Instead of applying all the features in the window at once, most informative features are used to firstly check whether the region can potentially have a face (False positive will be no issue here). This is the concept of cascade of classifiers, which groups the features into different stages. In this way, if the features in the window fail the first stage (which contains the most informative features), the rest of features are not necessary to apply and the computer knows this window does not contain any face. By using cascade of classifiers, the computation time reduces distinctively. In this project, Haar-like features of frontal face are obtained from OpenCV face detection library.

3.2.4 Local Binary Pattern Histogram (LBPH) * [4][5][6]

Local Binary Pattern (LBP) is an efficient texture operator which labels the pixels of an image by thresholding the neighborhood of each pixel and consider the result as binary number. This operator works with eight neighbors of a pixel, using the value of this center pixel as a threshold. If a neighbor pixel has a higher gray value (or the same gray value) than the center pixel. A one is assigned to that pixel. Otherwise a zero is assigned to that pixel. The LBP code for the center pixel is produced by concatenating the eight ones or zeros to a binary code, as Figure 5 shown.

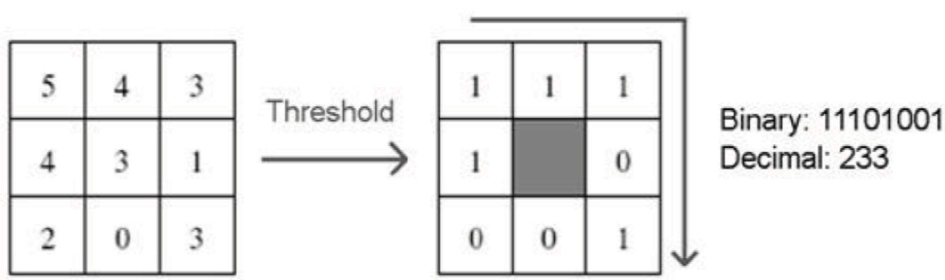

Figure 5. The LBP Operator [4].

As Figure 6 shown, an image can be divided into mxn regions and each region can be divided into hxh texture units (TU). Extracting the LBP of each TU, region can be described by the LBPH.

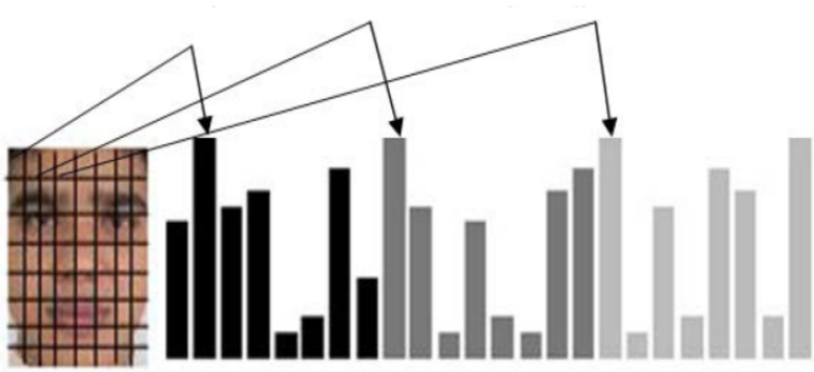

Figure 6. Face image divided into 64 regions, with for every region a histogram [5].

In this project, LBPH is used new method for the face recognition process. The computer trains the LBPH features with the dataset of frontal face of two people and uses the trained LBPH model to recognize the faces. LBP is rotation invariant and can be extract from different size of texture units [6].It also has many advantages such as that it has high discriminative power, computational simplicity, invariant to grayscale changes, and robust performance.

4 Results

This section demonstrates the results of the program.

Two people have been trained in this project. Thus, the first person appears in the video for the first time will be labeled as 1 and second person appears in the video for the first time will be labeled as 2. If the two people appear again another video, they will be still labeled as corresponding 1 and 2. Figure 7, Figure 8, and Figure 9 show the results corresponding to three videos. Most of them are correct detection and recognition. However, there are still many false positive cases and miss detection cases. The details about the results provided by program will be discussed in the next Discuss section.

Figure 7. Results for Video 1: Single Person 1 Appears in the Video.

Figure 8. Results for Video 2: Single Person 2 Appears in the Video.

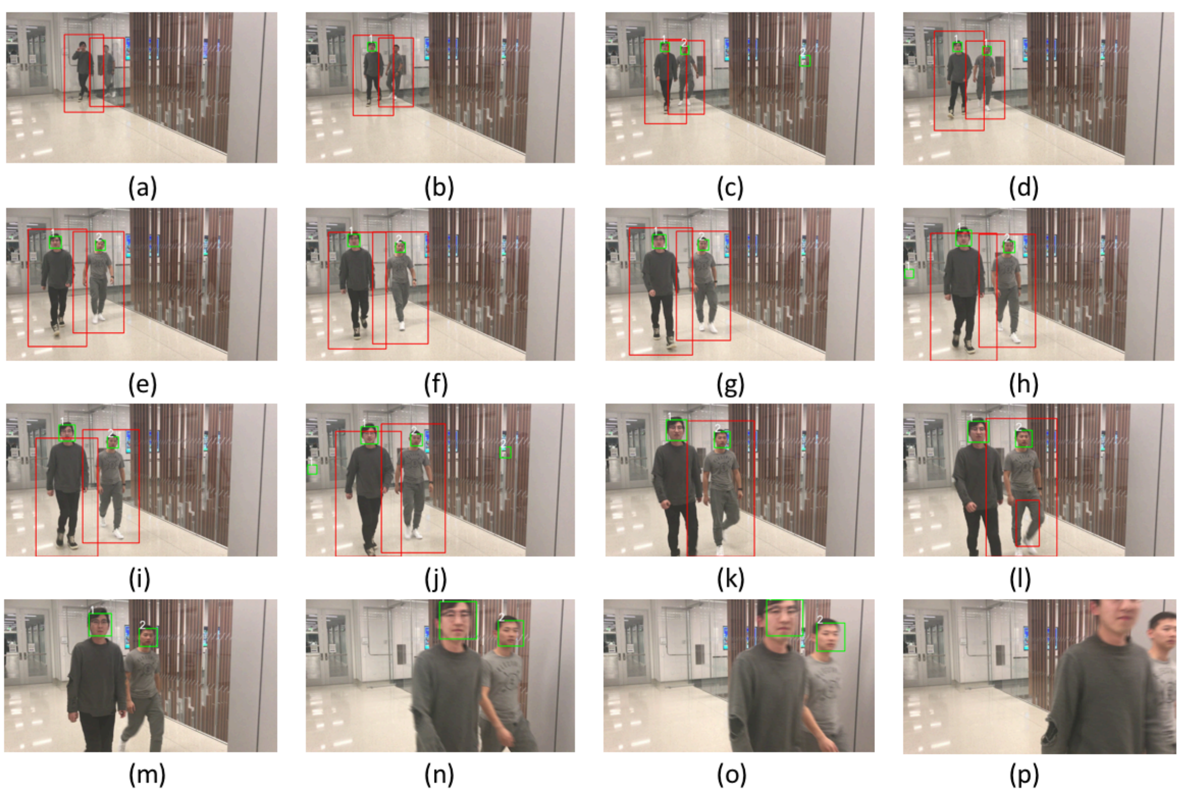

Figure 9. Results for Video 3: Both People Appear in the Video.

5 Discussion

There are three input videos in the project. Two people are pre-trained to create LBPH feature model. The person 1 appears individually in the video 1, and person 2 appears individually in the video 2. And both people appear again in the video 3.

For the video 1, as Figure 7 in the previous section shown, the computer detects person 1 at the beginning of the video and draws the red rectangle around his body. Since person 1 shows only the side face and there is a window between the person and camera, the face is not detected. The face is not detected and recognized until Figure 7(c). Once the face is detected, computer draws a green rectangle around his face. Since person 1 is the first one who appears in video 1. He is labeled as subject 1 by computer. From Figure 7(e) to 7(h), two false positive cases of face occur. When person 1 moves to Figure 7(j), his body is no longer detected because his full body is not presented in the video. In the Figure 7(m) and Figure 7(n), although the face is a little obscure due the fast movement, the face can still be detected and recognized.

For the video 2, as Figure 8 shown, the computer detects person 2 at the beginning of the video and draws the red rectangle around his body. The computer detects and recognizes his face in Figure 8(b) and draws the green rectangle around his face. Since person 2 is the second person who appears in the video sequence, he is labeled as subject 2 by computer. False positive cases occur several times in video 2, either on body or face. Again, although person 2's face is obscure due to fast movement from Figure 8(m) to 8(p), the face can still be detected and recognized.

For the video 3, as Figure 9 shown, the computer detects both person 1 and person 2 simultaneously at the beginning of the video. In the Figure 9(b), there is a miss detection on person 2's face. In Figure 9(c), a false positive case occurs that computer labels person 2 as subject 1. Both people are detected and recognized correctly from Figure 9(c). Since both of them already appear once in the previous videos, the computer still labels them as subject 1 and subject 2 instead of subject 3 and subject 4. In Figure 9(k), person 1 does not present the full body in the video but person 2 still present the full body. Thus, computer only detect person2' body. Once both of people do not present full body in the video, the computer no longer detects their bodies.

No matter in which video as mentioned previously, the computer can successfully track the bodies of person 1 and person 2 until one or both of them no longer fully present in the video. And computer can successfully track the faces of person 1 and person 2 until one or both of them pass by the camera. However, there are still some miss detection cases, such as Figure 8(a) and Figure 9(b). What's more, false positive cases like Figure 7(h), Figure 8(k), and Figure 9(d) occur many times.

Some improvements are needed in order to make the computer more robust and reliable. First necessary improvement is reducing the miss detection and false positive detection, which is quite a challenge. And detecting people who present only parts of the body is also an improvement. Instead of using HOG, the computer could use deformable parts model (DPM) algorithm to detect people who present parts of the body in the video. Moreover, it would be a great improvement if the computer can detect and recognize people's side face when they pass by the camera. Since the methods and algorithms used in the computer is conventional, using neural network and unsupervised learning to perform the human face detection and recognition is considered as advanced improvements. In such way, pre-training process is no longer necessary, and people do not have to put efforts in data acquisition process.

6 Conclusion

This project mainly focuses on tracking and recognizing a person in a video. The computer takes a video or a sequence of videos as input and outputs the labels on whose faces are seen in the videos. There are some limitations of the computer. It can only recognize a person with pre-trained model. And pre-trained model is created based on the frontal face sample images. Algorithms such as HOG, SVM, and non-maximum suppression are used for human detection. Haar features and cascade classifiers are used for face detection. LBPH features is used for face recognition. HOG, SVM, non-maximum suppression, Haar features and cascade classifiers are taught during the lecture. And LBPH is the new method implemented in the project to create robust LBPH models for recognition purposes. LBP has many advantages such as that it has high discriminative power, computational simplicity, invariant to grayscale changes, invariant to rotation and robust performance. For the most of time, person 1 and person 2 can be detected, recognized, and tracked correctly. Miss detection cases and false positive cases can still occur. In order to make computer robust and reliable, some improvements are necessary including reducing the miss detection and false positive detection, detecting people who present only parts of the bodies, detecting side faces when people pass by. For some advanced improvements, neural network and unsupervised learning can be used so the computer does not need to pre-trained model anymore.

7 Reference

[1] Dalal, Navneet, and Bill Triggs. "Histograms of oriented gradients for human detection." *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. Vol. 1. IEEE, 2005.

[2] Ruiz-Gonzalez, Ruben, et al. "An SVM-based classifier for estimating the state of various rotating components in agro-industrial machinery with a vibration signal acquired from a single point on the machine chassis." *Sensors* 14.11 (2014): 20713-20735.

[3] Bukis, Audrius, et al. "Survey of face detection and recognition methods." *Proceeding of International Conference on Electrical and Control Technologies, Kaunas, Lithuania*. Vol. 20. 2011.

[4] Ojala, Timo, Matti Pietikainen, and David Harwood. "Performance evaluation of texture measures with classification based on Kullback discrimination of distributions." *Pattern Recognition, 1994. Vol. 1-Conference A: Computer Vision & Image Processing., Proceedings of the 12th IAPR International Conference on*. Vol. 1. IEEE, 1994.

[5] Rahim, Md Abdur, et al. "Face recognition using local binary patterns (LBP)." *Global Journal of Computer Science and Technology* (2013).

[6] Mehta, Rakesh, and Karen O. Egiazarian. "Rotated Local Binary Pattern (RLBP)-Rotation Invariant Texture Descriptor." ICPRAM. 2013.

[7] Timo, Ahonen. "Face recognition with local binary patterns." *Euro. Conf. on Computer Vision*. 2004.

[8] Nugroho, Anto Satriyo, Arief Budi Witarto, and Dwi Handoko. "Support vector machine." *Teori dan Aplikasinya dalam Bioinformatika, Ilmu Komputer. com, Indonesia* (2003).

[9] Suykens, Johan AK, and Joos Vandewalle. "Least squares support vector machine classifiers." *Neural processing letters*9.3 (1999): 293-300.

[10] Kasinski, Andrzej, and Adam Schmidt. "The architecture and performance of the face and eyes detection system based on the Haar cascade classifiers." *Pattern Analysis and Applications* 13.2 (2010): 197-211.

[11] Bradski, Gary, and Adrian Kaehler. "OpenCV." *Dr. Dobb's journal of software tools* 3 (2000).

[12] Viola, Paul, and Michael Jones. "Rapid object detection using a boosted cascade of simple features." *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*. Vol. 1. IEEE, 2001.

Appendix A: Program codes

Python 3.6 and OpenCV 3.3 are used for this project. Python libraries such Python Image Library, Imutils, and numpy are used.

There are three major programs in the project including extractFace.py, createFaceModel.py, and main.py. Figure 2 provides the details about how these programs work with each other.

Details about program can be view on github.

Appendix B: Course Feedback

In EECS432, professor introduce us lots of useful theories and algorithms for many topics in the computer vision field. This course provides me with a broad overview and deep details on topics such as image segmentation, object detection, object recognition, and etc. Since I major in mechanical engineering with robotics specializations, I become interested in the computer vision after taking this course and want to participate some computer vision related projects in the future. However, I think this course is too theoretical. I really hope professor can provide some homework or practices which correspond to the content professor teaches during the lectures so we could have some practical experience in using the theories and algorithms.