Republic of Iraq

The Ministry of Higher Education
and Scientific Research

University of Technology

Computer Science Department

FACE DETECTION USING ARTIFICIAL NEURAL NETWORK

This project submitted to the department of Computer Science, Computer Science department at the University of Technology in partial fulfillment of the requirements for the Degree of B.Sc. in Computer Science – Software.

By

Muayyad Emad Najim

Khaled Waleed Khaled

Ahmed Majid Jabbar

Supervised by

Lec. Dr. Ragheed Dawood Salem

2021-2022 **Baghdad** **1443**

Table of Contents

Abstract

Face detection is an important research direction in the field of target detection, it is a computer vision problem that involves finding faces in images. It is also the initial step for many face-related technologies, for instance, face verification, face modeling, head pose tracking, gender and age recognition, facial expression recognition, and many more.

For the input image, the position of the face is returned. In order to complete the task of face detection using deep learning, data input, feature extraction and face feature detection are three steps, among which feature extraction is the most important part.

Studying the basic principles of current mainstream target detection algorithms, this project compares the characteristics of Two-stage and One-stage detection models and their application in face detection tasks, Also, MTCNN (Multi-task convolution neural network) is deeply analyzed and its implementation principle is introduced in detail. The real effect of MTCNN in face detection task is verified by experiments.

List of Equations

List of Figures

List of Abbreviations

Abbreviations	Meaning
NN(s)	Neural Networks
ANN(s)	Artificial Neural Networks
CNN(s) / ConvNet(s)	Convolutional Neural Networks
Conv	Convolution
MTCNN(s)	Multi-Task Cascaded Convolutional Networks
AI	Artificial Intelligence
DPM(s)	Deformable Part Models
P-Net	Proposal Network
O-Net	Output Network
R-Net	Refine Network
NMS	Non-Maximum Suppression
MP	Max Pooling

Chapter 1
General Introduction

1.1 Introduction to Facial Detection

Facial detection is a well-known computer vision application, widely known in today's era which can be solved using AI (Artificial Intelligence). AI is a technology which replicates neurons in the human brain and tries to mimic its learning process of identifying and solving real-world problems such as facial detection, object detection, facial recognition, etc.

The AI needs to be trained to perform such tasks in a way in which a child needs to be educated on a particular topic. Recognizing a face using various elements such as skin color, mouth, eyes, and nose requires guidance which is provided by training instances, also known as supervised learning.

1.2 Machine Learning (ML)

Machine learning is a field of study that applies the principles of computer science and statistics to create statistical models, which are used for future predictions (based on past data or *Big Data*) and identifying (discovering) patterns in data. Machine learning is itself a type of artificial intelligence that allows software applications to become more accurate in predicting outcomes without being explicitly programmed.

1.3 Deep Learning

Deep learning, also known as the deep neural network, is one of the approaches to machine learning. Other major approaches include decision tree learning, inductive logic programming, clustering, reinforcement learning, and Bayesian networks.

Deep learning is a special type of machine learning. It involves the study of ANN and ML related algorithms that contain more than one hidden layer.

1.4 Artificial Neural Networks (ANNs)

Usually simply called neural networks (NNs), are computing systems inspired by the biological neural networks that constitute animal brains.

An ANN is based on a collection of connected units or nodes called artificial neurons, which loosely model the neurons in a biological brain. Each connection, like the synapses in a biological brain, can transmit a signal to other neurons. An artificial neuron receives signals then processes them and can signal neurons connected to it. The "signal" at a connection is a real number, and the output of each neuron is computed by some non-linear function of the sum of its inputs. The connections are called edges. Neurons and edges typically have a weight that adjusts as learning proceeds. The weight increases or decreases the strength of the signal at a connection. Neurons may have a threshold such that a signal is sent only if the aggregate signal crosses that threshold. Typically, neurons are aggregated into layers. Different layers may perform different transformations on their inputs.

Signals travel from the first layer (the input layer), to the last layer (the output layer), possibly after traversing the layers multiple times.

1.5 Convolutional Neural Networks (CNNs)

In deep learning, a convolutional neural network (CNN, or ConvNet) is a class of artificial neural network (ANN), most commonly applied to analyze visual imagery. [1].

Since a facial detection task involves images/video feeds, a convolutional neural network could be a potential candidate for solving the problem at hand. There is a considerably huge difference between a human’s and a computer’s perception of an image. The human brain is trained to extract features from an image and distinguish them, whereas, computers view an image in the form of numbers, i.e., pixels. These numbers range from 0 to 255, describing the pixel intensity at every point.

1.6 Multi-task Cascaded Neural Networks (MTCNNs)

MTCNN is a method of face detection and alignment primarily based on deep convolutional neural networks [2,3] that are to mention, this technique can accomplish the assignment of face detection and alignment at the same time. In comparison with the traditional approach, MTCNN has better overall performance, can appropriately discover the face, and the speed is likewise faster, besides, MTCNN also can hit upon in real-time.

Chapter 2
Theoretical Background & Related Work

2.1 Introduction to Neural Networks history

The neural network is a computer system modeled after the human brain. In simple words, a neural network is a computer simulation of the way biological neurons work within a human brain.

As per ***Dr. Robert Hecht-Nielsen***, the inventor of one of the first neurocomputers, a neural network or **artificial neural network (ANN) is**

"...a computing system made up of a number of simple, highly interconnected processing elements, which process information by their dynamic state response to external inputs."

ANN is a group of algorithms that are used for machine learning (or precisely deep learning). Alternatively, think like this – **ANN is a form of deep learning, which is a type of machine learning, and machine learning is a subfield of artificial intelligence**

Neural networks are not a new invention, its history dates back to 1943 when McCulloch and Pitts described how the basic processing elements of the brain worked, and demonstrated it using electrical circuits [4]. Neural Networks (NN) in the machine learning domain refers to Artificial Neural Networks, as opposed to biological neural networks of the brain. NN have evolved to become a major building block for machine learning algorithms. What is remarkably with NN is their ability to learn from data. An example that is hard or even impossible to program by hand, is to recognize handwritten numbers in images. By giving many examples of differently written numbers, together with their true value, the network can learn a function that maps an image to a number. When exposing a trained NN to new examples, the network is capable of predicting correct number with high precision. The basic building block of a neural network is the single node perceptron. A perceptron takes an input X of size n, and has one output y. A weight w is associated with each input, and the there is a bias associated with the perceptron.

$$y = f(\sum_{i=1}^{i=n} w_i x_i + bias).$$

Equation 2.1: Perceptron Equation. [5]

2.2 Neural network-based face detection

Early in 1994 Vaillant et al. [6] applied neural networks for face detection. In their work, they proposed to train a convolutional neural network to detect the presence or absence of a face in an image window and scan the whole image with the network at all possible locations. In 1996, Rowley et al. [7] presented a retinally connected neural network for upright frontal face detection. The method was extended for rotation invariant face detection later in 1998 [8] with a "router" network to estimate the orientation and apply the proper detector network. In 2002 Garcia et al. [9] developed a neural network to detect semi-frontal human faces in complex images; in 2005 Osadchy et al. [10] trained a convolutional network for simultaneous face detection and pose estimation. It is unknown how these detectors perform in today's benchmarks with faces in uncontrolled environments. Nevertheless, given recent break-through results of CNNs [11] for image classification [12] and object detection [13], it is worth to revisit the neural network-based face detection. One of the recent CNN based detection method is the R-CNN by Girshick et al. [14] which has achieved the state of-the-art result on VOC 2012. R-CNN follows the "recognition using regions" paradigm. It generates categoryindependent region proposals and extracts CNN features from the regions. Then it applies class-specific classifiers to recognize the object category of the proposals. Compared with the general object detection task, uncontrolled face detection presents different challenges that make it impractical to directly apply the R-CNN method to face detection. For example, the general object proposal methods may not be effective for faces due to small-sized faces and complex appearance variations.

2.3 Face Detection History

Face detection are essential to many face applications, such as face recognition and facial expression analysis. However, the large visual variations of faces, such as occlusions, large pose variations and extreme lightings, impose great challenges for these tasks in real world applications. The cascade face detector proposed by Viola and Jones [15] utilizes Haar-Like features and AdaBoost to train cascaded classifiers, which achieves good performance with

real-time efficiency. However, quite a few works [16, 17, 18] indicate that this kind of detector may degrade significantly in real-world applications with larger visual variations of human faces even with more advanced features and classifiers. Besides the cascade structure [19, 20, 21], introduce deformable part models (DPM) for face detection and achieve remarkable performance. However, they are computationally expensive and may usually require expensive annotation in the training stage. Recently, convolutional neural networks (CNNs) achieve remarkable progresses in a variety of computer vision tasks, such as image classification [22] and face recognition [23]. Inspired by the significant successes of deep learning methods in computer vision tasks, several studies utilize deep CNNs for face detection. Yang et al [24] train deep convolution neural networks for facial attribute recognition to obtain high response in face regions which further yield candidate windows of faces. However, due to its complex CNN structure, this approach is time costly in practice. Li et al. [25] use cascaded CNNs for face detection, but it requires bounding box calibration from face detection with extra computational expense and ignores the inherent correlation between facial landmarks localization and bounding box regression.

2.4 MTCNN History

MTCNN is a deep cascaded multi-task framework which exploits the inherent correlation between detection and alignment to boost up their performance. The framework of MTCNN leverages a cascaded architecture with three stages of carefully designed deep convolutional networks to predict face and landmark location in a coarse-to-fine manner. In addition, a new online hard sample mining strategy that further improves the performance in practice.

This method was first proposed in 2016 by Kaipeng Zhang et al. in their paper ‘Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks’, IEEE Signal Processing Letters, Volume: 23 Issue: 10.

Chapter 3

Proposed Work & Implementation

3.1 Introduction

CNN is composed of two main components:

1. **Feature Learning:** In reference to face detection, the feature learning task of CNN would involve learning how different parts of the face look depending on height, width, and other features.

2. **Classification:** The classification task assigns a probability for an entity in the image depending on the object to be predicted; In this case, the human face.

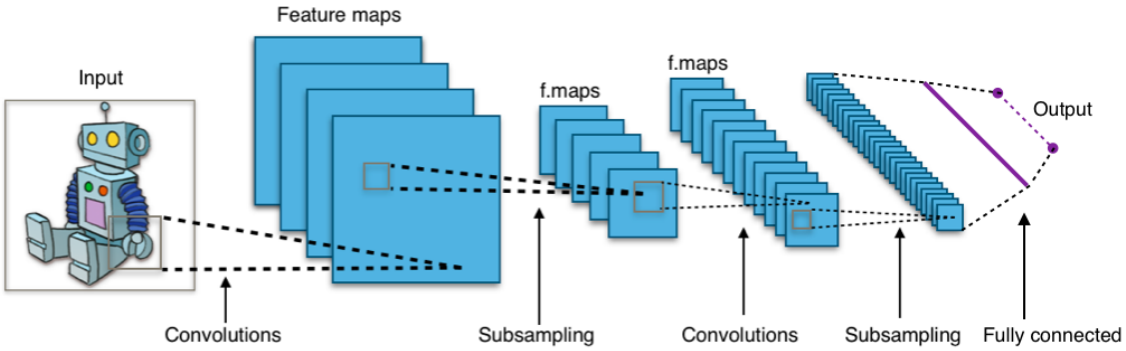

Figure 3.1: Typical CNN architecture. [1]

3.2 Proposed Work

By default, the MTCNN bundles a face detection weights model.

MTCNN implements the face area detection and face key point detection together, and its subject framework is similar to cascade. The whole can be divided into a three-layer network structure of Proposal Network (P-Net), Refine Network (R-Net), and Output Network (O-Net) [26,27]. It is a multi-task neural network model for face detection tasks which mainly uses three cascaded networks and the idea of candidate boxes plus classifiers.

MTCNN consists of 3 neural network cascades, particularly p-net, r-net, and o-internet. With the intention to attain face reputation on a unified scale, the authentic picture must be scaled to one-of-a-kind scale to shape a photo pyramid before the usage of these networks.

The whole concept of MTCNN can be explained in **three stages** out of which, in the third stage, **facial detection and facial landmarks** are performed simultaneously. These stages consist of various CNNs with varying complexities.

A simple explanation of the three stages of MTCNN can be as follows:

1. In the first stage the MTCNN creates multiple frames which scan through the entire image starting from the top left corner and eventually progressing towards the bottom right corner. The information retrieval process is called **P-Net (Proposal Net)** which is a shallow, fully connected CNN.

2. In the second stage all the information from P-Net is used as an input for the next layer of CNN called as **R-Net (Refinement Network)**, a fully connected, complex CNN which rejects a majority of the frames which do not contain faces.

3. In the third and final stage, a more powerful and complex CNN, known as **O-Net (Output Network)**, which as the name suggests, outputs the facial landmark position detecting a face from the given image/video.

3.3 Experiments

In this section, we first evaluate the effectiveness of the proposed hard sample mining strategy. Then we compare our face detector and alignment against the state-of-the-art methods in Face Detection Data Set and Benchmark (FDDB) [28], WIDER FACE [29], and Annotated Facial Landmarks in the Wild (AFLW) benchmark [30]. FDDB dataset contains the annotations for 5,171 faces in a set of 2,845 images. WIDER FACE dataset consists of 393,703 labeled face bounding boxes in 32,203 images where 50% of them for testing into three subsets according to the difficulty of images, 40% for training and the remaining for validation. AFLW contains the facial landmarks annotations for 24,386 faces and we use the same test subset as [31]. Finally, we evaluate the computational efficiency of our face detector.

3.4 Approach

We are using MTCNN framework which is a deep cascaded multi-task framework that exploits the inherent correlation between detection and alignment to boost up their performance. The framework of MTCNN leverages a cascaded architecture with the three stages of carefully designed deep convolutional networks to predict face and landmark location in a coarse-to-fine manner.

The MTCNN is trained on the following three tasks:

1. Face classification. A two-class classification problem: Face or not face.

2. Bounding box regression: For each candidate window, find offset from the nearest ground truth. Bounding box coordinates consist of left, top, width, and height.

3. Landmark position localization: The network outputs landmark positions. Training data has ground truth data, and during training the

Euclidean loss is minimized.

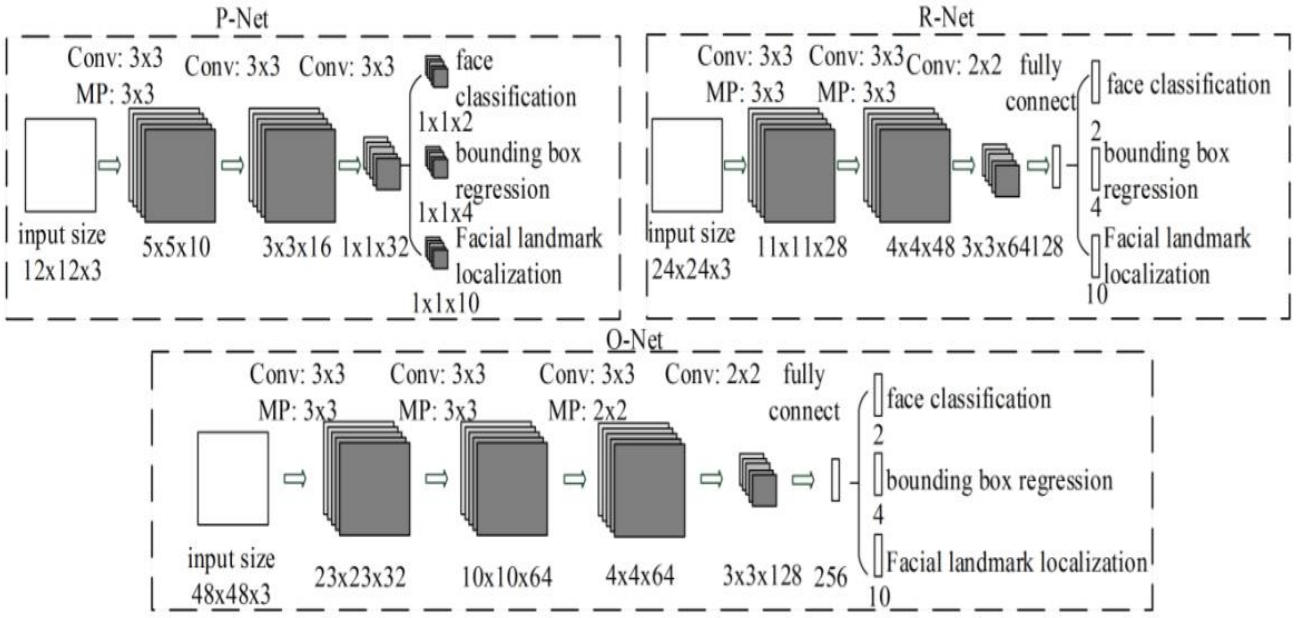

Figure 3.2: The architectures of P-Net, R-Net, and O-Net, the step size in convolution and pooling is 1 and 2, respectively. [2]

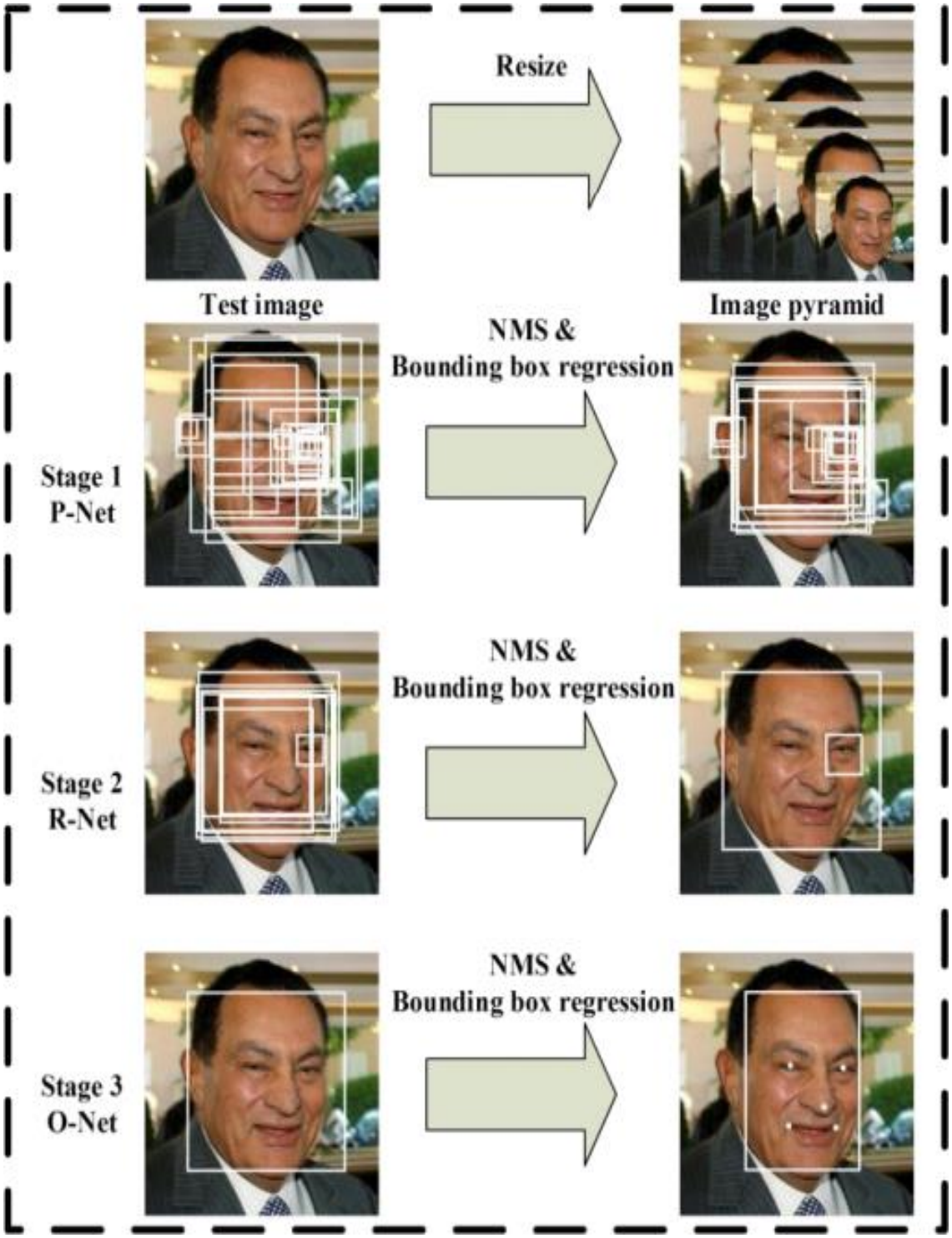

Figure 3.3: Pipeline of our cascaded framework that includes three-stage multi-task deep convolutional networks. Firstly, candidate windows are produced through a fast Proposal Network (P-Net). After that, we refine these candidates in the next stage through a Refinement Network (R-Net). In the third stage, The Output Network (O-Net) produces final bounding box and facial landmarks position. [2]

3.5 Flowchart and Pseudocode

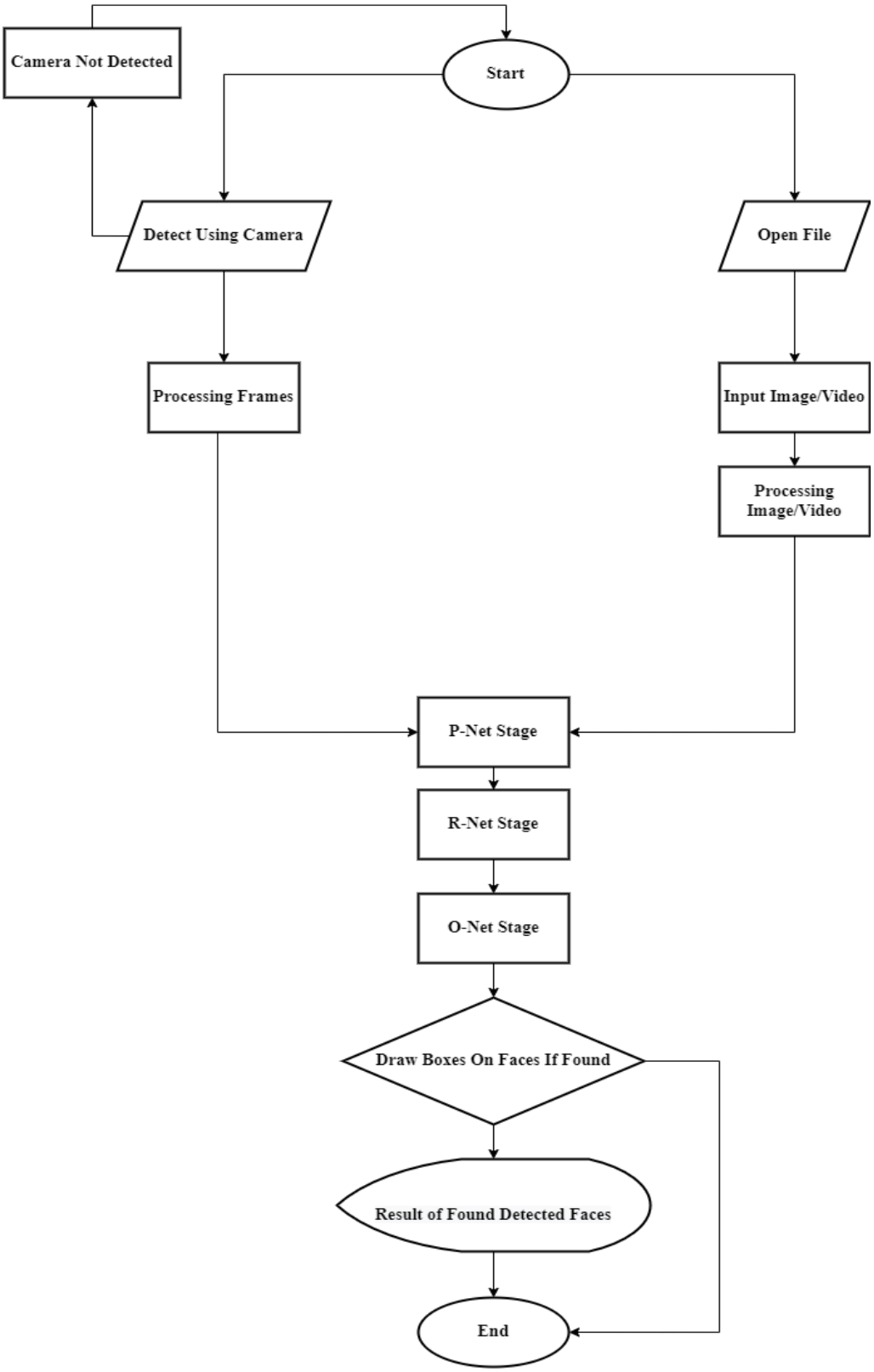

Figure 3.4: A flowchart diagram of the proposed program

```
def main_func ():
    # calling a function to select an image
    filename = select_file()
    # reading the input image/video
    pixels = pyplot.imread(filename)
    # create the detector, using default weights
    detector = MTCNN()
    # detect faces in the image
    faces = detector.detect_faces(pixels)
# calling the main function
main_func()
# function to output the image with box drawn on detected faces
def draw_faces(filename, result_list):
        # load the image
        data = pyplot.imread(filename)
        # plot the image
        pyplot.imshow(data)
        # get the context for drawing boxes
        ax = pyplot.gca()
        # plot each box
        for result in result_list:
                # get coordinates
                x, y, width, height = result['box']
                # create the shape
                rect = Rectangle((x, y), width, height, fill=False, color='red')
                # draw the box
                ax.add_patch(rect)
        # show the plot
        pyplot.show()
    # calling draw_faces function
draw_faces(filename, faces)
```

3.6 The Implementation and Results

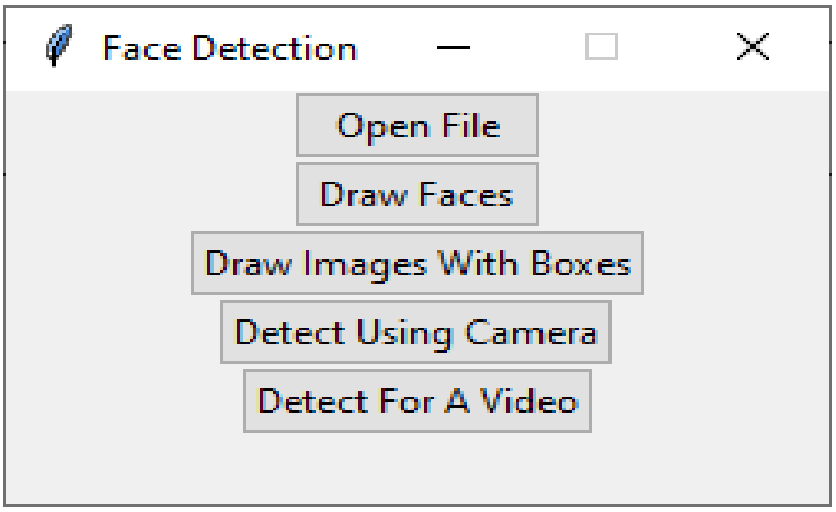

Figure 3.5: Main Menu of the user interface.

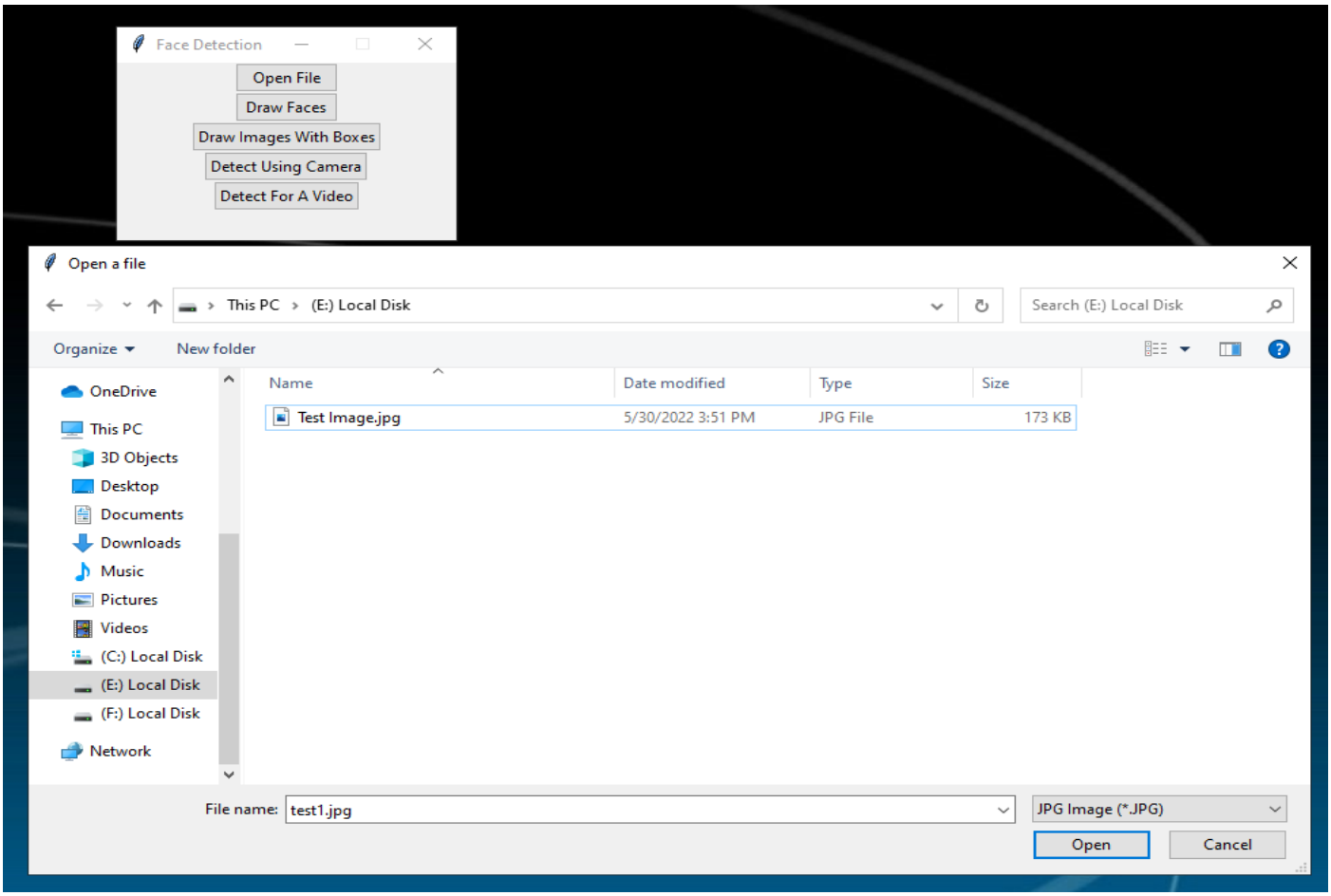

Figure 3.6: Opening a file dialog to select a media file to put it to test.

Figure 0.7: The sample image we use.

Figure 3.8: Drawing rectangle over the detected faces in the sample image.

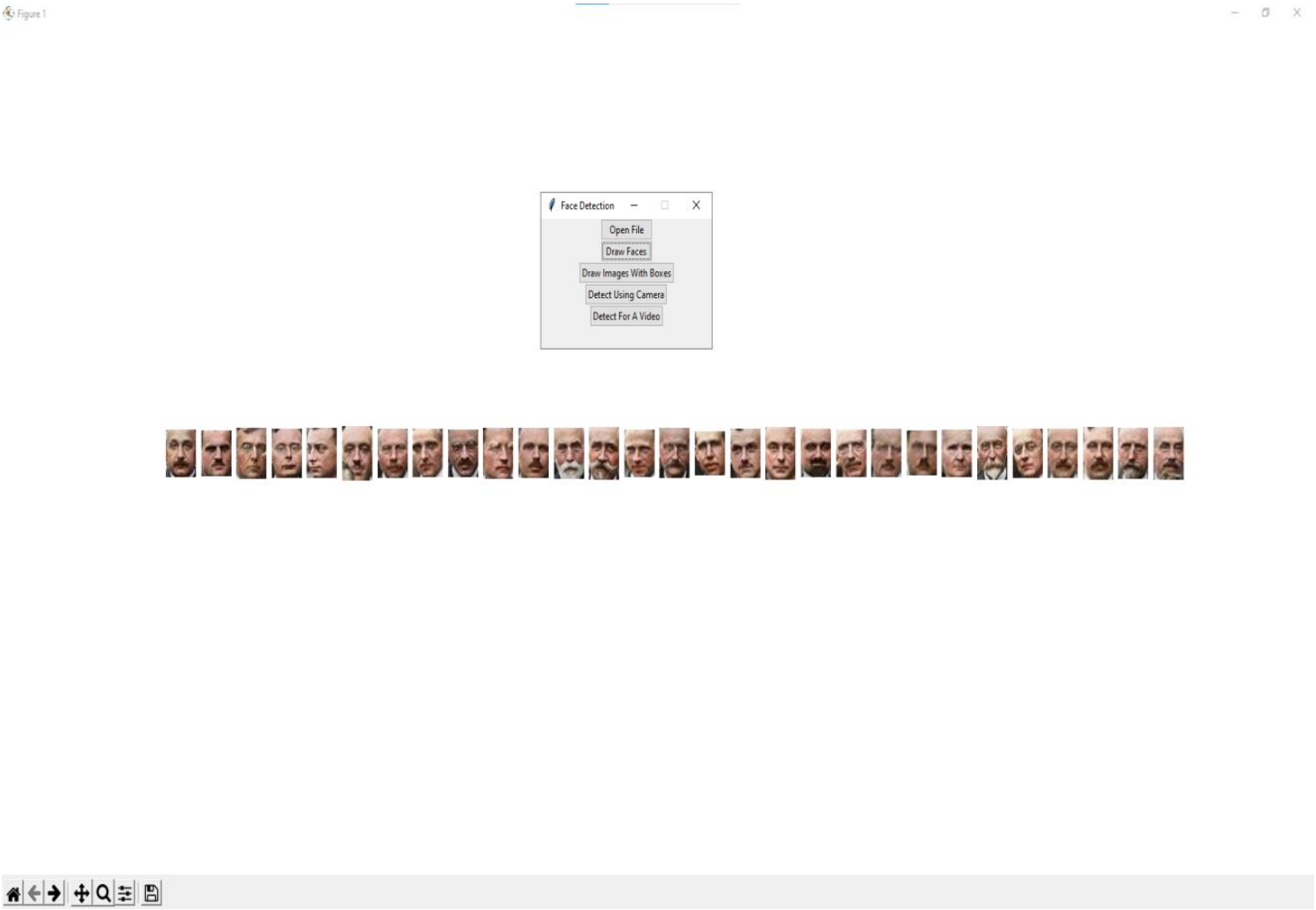

Figure 3.9: Extracting the detected faces from the sample image.

Chapter 4

Conclusion & Future Work

4.1 Conclusion

Face detection is a computer vision problem that involves finding faces in photos.

It is a trivial problem for humans to solve and has been solved reasonably well by classical feature-based techniques, such as the cascade classifier. More recently deep learning methods have achieved state-of-the-art results on standard benchmark face detection datasets. One example is the Multi-task Cascade Convolutional Neural Network or MTCNN for short.

4.2 Future work

- Improving the face detection for bad lightening conditions.
- Speeding up the process of face detection.
- Face detection using a thermal camera.
- Face detection on web pages for the purpose of online classrooms attendance.
- Distorted faces detection.
- Mask detection to detect if the person is wearing a mask on.

References

[1] Wikipedia contributors. "Convolutional neural network." Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 24 May. 2022. Web. 1 Jun. 2022.

[2] Zhang K, Zhang Z, Li Z, et al. Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks [J]. IEEE Signal Processing Letters, 2016, 23(10):1499-1503.

[3] Xiang J, Zhu G. [IEEE 2017 4th International Conference on Information Science and Control Engineering (ICISCE) - Changsha (2017.7.21-2017.7.23)] 2017 4th International Conference on Information Science and Control Engineering (ICISCE) - Joint Face Detection and Facial Expression Recognition with MTCNN[C] // International Conference on Information Science & Control Engineering. IEEE Computer Society, 2017:424-427.

[4] Jack D. Cowan. Discussion: mcculloch-pitts and related neural nets from 1943 to 1989. Bulletin of Mathematical Biology, 52(1):73 – 97, 1990.

[5] Mauro Castelli Leonardo Vanneschi. Perceptron.

[6] R. Vaillant, C. Monrocq, and Y. Le Cun. Original approach for the localisation of objects in images. IEE ProceedingsVision, Image and Signal Processing, 1994.

[7] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. In Computer Vision and Pattern Recognition, 1996.

[8] H. A. Rowley, S. Baluja, and T. Kanade. Rotation invariant neural network-based face detection. In Computer Vision and Pattern Recognition, 1998.

[9] C. Garcia and M. Delakis. A neural architecture for fast and robust face detection. In Pattern Recognition, 2002. Proceedings. 16th International Conference on, 2002.

[10] M. Osadchy, Y. L. Cun, M. L. Miller, and P. Perona. Synergistic face detection and pose estimation with energy-based model. In In Advances in Neural Information Processing Systems (NIPS), 2005.

[11] Y. LeCun and Y. Bengio. Convolutional networks for images, speech, and time series. The handbook of brain theory and neural networks, 1995.

[12] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge, 2014.

[13] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge, 2009.

[14] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. arXiv preprint arXiv:1311.2524, 2013.

[15] P. Viola and M. J. Jones, "Robust real-time face detection. International journal of computer vision," vol. 57, no. 2, pp. 137-154, 2004

[16] B. Yang, J. Yan, Z. Lei, and S. Z. Li, "Aggregate channel features for multi-view face detection," in IEEE International Joint Conference on Biometrics, 2014, pp. 1-8.

[17] M. T. Pham, Y. Gao, V. D. D. Hoang, and T. J. Cham, "Fast polygonal integration and its application in extending haar-like features to improve object detection," in IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 942-949.

[18] Q. Zhu, M. C. Yeh, K. T. Cheng, and S. Avidan, "Fast human detection using a cascade of histograms of oriented gradients," in IEEE Computer Conference on Computer Vision and Pattern Recognition, 2006, pp. 1491-1498.

[19] M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool, "Face detection without bells and whistles," in European Conference on Computer Vision, 2014, pp. 720-735.

[20] J. Yan, Z. Lei, L. Wen, and S. Li, "The fastest deformable part model for object detection," in IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 2497-2504.

[21] X. Zhu, and D. Ramanan, “Face detection, pose estimation, and landmark localization in the wild,” in IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 2879-2886.

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in Advances in neural information processing systems, 2012, pp. 1097-1105.

[23] Y. Sun, Y. Chen, X. Wang, and X. Tang, “Deep learning face representation by joint identification-verification,” in Advances in Neural Information Processing Systems, 2014, pp. 1988-1996.

[24] S. Yang, P. Luo, C. C. Loy, and X. Tang, “From facial parts responses to face detection: A deep learning approach,” in IEEE International Conference on Computer Vision, 2015, pp. 3676-3684.

[25] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, “A convolutional neural network cascade for face detection,” in IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 5325-5334.

[26] L. H. Ma, H. Y. Fan, Z. M. Lu, D. Tian, Acceleration of multi-task cascaded convolutional networks, IET Image Process., 14 (2020), 2435–2441.

[27] X. Zhao, S. Lin, X. Chen, C. Ou, C. Liao, Application of face image detection based on deep learning in privacy security of intelligent cloud platform, Multimed. Tools Appl., 79 (2020), 205–210.

[28] V. Jain, and E. G. Learned-Miller, “FDDB: A benchmark for face detection in unconstrained settings,” Technical Report UMCS-2010-009, University of Massachusetts, Amherst, 2010.

[29] S. Yang, P. Luo, C. C. Loy, and X. Tang, “WIDER FACE: A Face Detection Benchmark”. arXiv preprint arXiv:1511.06523.

[30] M. Köstinger, P. Wohlhart, P. M. Roth, and H. Bischof, “Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization,” in IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2011, pp. 2144-2151.

[31] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, “Facial landmark detection by deep multi-task learning,” in European Conference on Computer Vision, 2014, pp. 94-108.