PAKISTAN JOURNAL OF EMERGING SCIENCE AND TECHNOLOGIES (PJEST)

June, 2021, 1(1) ISSN 2788-8630 (On-Line)

FACE DETECTORS EVALUATION TO SELECT THE FASTEST AMONG DLIB, HAAR CASCADE, AND MTCNN

Kousar Abdul Majeed[1*], Zain Abbas[1], Maheen Bakhtyar[1], Zohaib Durrani[1],
Junaid Baber[1], Ihsan Ullah[1], Atiq Ahmed[1]
[1]Department of Computer Science and IT University of Balochistan Quetta
*Correspondence: kousarabdulmajeed3@gmail.com

ARTICLE INFORMATION

Citation: Kousar Abdul Majeed, Zain Abbas, Maheen Bakhtyar, Zohaib Durrani, Junaid Baber, Ihsan Ullah, & Atiq Ahmed. (2021). Face Detectors Evaluation to Select the Fastest among DLIB, HAAR Cascade, and MTCNN (Version 2). Pakistan Journal of Emerging Science and Technologies, 2(1), 13. http://doi.org/10.5281/zenodo.5089391

Received: 09th June-2021,

Revised and Accepted:

01-July-2021

Published On-Line

10-July-2021

***Corresponding Author:**

Kousar Abdul Majeed:

kousarabdulmajeed3@gmail.com

Original Research Article

ABSTRACT

Face detection is an important problem in computer vision research and applications are getting trending due to the advancement in the file of machine learning and computer vision. There are several algorithms and models for face recognition. However, face detection is the first step in all implementations. This research proposed a face detection method based on an enhanced Multi-Task Convolution Neural Network (MTCNN) and improves the network of MTCNN, creates a neural network model based on MTCNN using Python, and cascades to increase the accuracy of face location in difficult scenarios. In this research paper, we evaluated the performance of three famous face detector models on CPU-based machines. Experiments show that HAAR Cascade is the fastest on CPU-based machines but in the case of accuracy, MTCNN is better. MTCNN and DLIB based detectors are designed for GPU-based machines.

Keywords: Face recognition, DLIB face detector, HAAR Cascade Classifier, MTCNN (Multi-Task Learning)

Introduction:

In the multimedia digital world, image and video are becoming extremely important. In every image or video, the human face is one of the most significant elements. Detecting the human faces in an image and subsequently extracting the facial characteristics is a useful technique for a variety of applications, including computer interface, video conferencing, and so on.

Detecting human faces and extracting facial characteristics from an image is, in reality, and a difficult task that has been researched in computer vision for decades. The goal of improving face recognition rates is to extract valuable facial features. It has been a challenging task in machine learning for a long time, with applications ranging from image classification, face analysis, face identification, face tracking, and need the monitoring and detection of several faces at a distance. The integral image, classifier learning using AdaBoost, and the HaarCascade are three fundamental principles in the Viola-Jones face detector that make it possible to create an effective face detector that can execute in real-time.

Face detection is used to identify whether there are any faces in the image or not. Following Viola-Jones' pioneering work, other detection processes have been created to recognize faces efficiently and correctly [1]. The initial stage in a computerized face recognition system is face detection. It generally decides whether or not a picture has a face in it. If it does, it can track one or more facial positions in the image. The feature extraction stage includes selecting a unique feature vector from the identified face. Verification and identity are required for classification. To allow access to the desired identity, verification involves matching one face to another. Identification, on the other hand, compares a face to numerous other faces and offers numerous options for determining the face's identity [2]. Feature extraction subset feature and emotion classifier are the three phases. The feature value of the input picture, a face, is derived using Sobel edge detection and a Haar Cascades technique for feature extraction of the mouth and eyes [3]. Because of its robustness, we chose Haar cascade for face detection in this research. For feature extraction, there are two main approaches: holistic feature and local feature. The characteristics are retrieved from the entire face in the holistic feature-based method, which might be influenced by occlusion and expression variations. Local feature-based techniques, on the other hand, solve problems by considering sections of the image. They are also invariant in terms of size and rotation [4]. Filtering methods are another option to improve the accuracy of facial recognition [5].

To produce a discriminative subspace, different postural features are converted using a transformation dictionary. Face matching is done on a patch-by-patch basis rather than on a holistic level. Experimentation is carried out using the Multi-PIE, and FERET databases. When compared to existing approaches of posing issues and single-task-based baselines, the suggested technique produces better results. MTL techniques are used to improve these tasks. Face Recognition with MTL suggest that the characteristics of different postures be transformed into a discriminative subspace and that the transformations be learned simultaneously for all postures with one challenge for each posture for the MTL-based face recognition algorithm [6]. To rotate a face image, [7] creates a deep network. This multi-task framework outperforms the single-task model without the reconstruction component. Similar work [8] has been done in this way to extract robust identification traits while simultaneously synthesizing facial pictures.

Related Work

Computer vision is a topic of research that aims to create strategies to aid computers in analyzing the information of digital pictures like video and images. It's also automated processing, which includes several major procedures for image processing, recognition, and decision-making [9]. Faces are detected in a sliding window manner using new features and classifiers. Viola and Jones' remarkable work [41], AdaBoost is used with Haar cascade features in which it is used to train a cascaded face detector, which then influences various other techniques [10]. The Viola and Jones method use Haar classifiers to identify facial characteristics. In computer vision, Haar features are used to classify the intensity of pixels in a region in a detectable manner. Haar features are image rectangle areas, and classifiers are made up of two or three rectangle features that are continually inspected for features in the window. Face recognition systems that are robust to pose variations, and facial expressions [11]. The fundamental goal of these techniques is to find distinguishing characteristics. In general, these techniques may be split into two types: type one is to extract local characteristics, local appearance-based approaches are applied, and the facial picture is separated into smaller sections [46]. Type two is to extract the features centered on these places, key-points algorithms are utilized to identify the facial picture.

Techniques Based on Local Appearance It's a geometrical methodology that's also referred to as a feature methodology. In this scenario, the facial picture is represented as a series of low-dimensional characteristic vectors. To describe local features, this concentrates on essential portions of the face and employs the face's uniqueness to identify and apply several parameters, such as pixel orientations, histograms [12], geometric features, and correlation planes [47]. LBP and variants: LBP is a general-purpose texture extraction approach that may be applied to any object [51]. It excels at face recognition [47], facial expression identification, texture classification, and texture segmentation. Facial image retrieval is an interesting but difficult challenge for its practical usage in computational forensics. It's much worse when the system doesn't identify the target image and it's only in the user's mind. As a result, accurate retrieval results require the user's descriptions of the target image. As a result, factor analysis techniques such as PCA, Fisher's linear discriminant, ICA, nonnegative matrix factorization, and probabilistic PCA were applied to these datasets. Facial recognition and retrieval are significant computer vision and image processing issues. The system works by retrieving valuable features to be used in the retrieval process, similar to face recognition systems. According to studies, interactive image retrieval has the benefits of incorporating user input and enhancing retrieval performance through relevance feedback.

To extract data, [13] employed an unsupervised deep learning system called the LBP network. The topology of the LBPNet is identical to that of the convolutional neural network (CNN). LBPNet is similar to other unsupervised approaches, according to experimental findings acquired using available benchmarks (e.g., LFW and FERET). Reference [55] developed a system for resolving face recognition problems including a wide range of characteristics such as emotion, lighting, and various positions. It is based on two different techniques: K-NN and Local Binary Pattern. The local binary pattern used face recognition algorithms to its invariance to target picture rotation. For feature extraction, reference [14] presented a variation of the local binary pattern methodology which is called "multiscale local binary pattern (MLBP)." The local ternary pattern methodology [15] is another local binary pattern variant that is less noise-sensitive than the original LBP methodology. The HOG technique can define the face shape by looking at the distribution of edge direction or the light intensity gradient [2]. This methodology works by dividing the entire face picture into cells, creating a histogram of pixel edge direction, and then combining the histograms to extract the face picture's feature. Reference [16] suggested a robust face recognition system based on a mixture of several histograms of oriented gradients (HOG). The approach is known as "multi-HOG."

In addition, a face detection system has been built to determine the item that looks like a face. Faces in the Wild and dataset of a wider face with a great degree of variety in circumstances such as size, position, occlusion, expression, cosmetics, backdrop, and light were used in this study. Reference [17] created the LWF image database to research the challenge of face recognition, which contains variations in posture, facial expression, lighting, age, gender, color saturation, image quality, clothing, focus, and other parameters. 13,233 facial images were acquired from the website, representing 5749 users, with two or more different images 1680 and each image having a resolution of 250 by 250 pixels. For the classification model, the dataset was divided into ten non-repeating subsets of image pairs. There are 300 positive pairs and 300 negative pairs in each subset. When the database is only used for checking, all of the pairs are used to calculate accuracy. The database was created to research a special task that a pair of people had to complete. A pair of face images is given, and a classifier may identify the pair as “same” or “different” depending on whether or not the images depict the same person. A database was created to research a special task that a pair of people had to complete. Two procedures have been created to make use of the training data: Image-restricted and unrestricted training are also available. The identities of pictures are removed in image-restricted training. (1) Funneled images [18], (2) LFW-A employed an unknown process for alignment, and (3) deep funneled pictures [19] are the three types of aligned photographs presented. The WIDER Face [20] data set contains 32,203 photos, with 50% samples being utilized for training and 10% being utilized for validation. The poses, light, occlusion, and size of the faces in this data set vary substantially. Face detectors that have been trained on this data set have a greater performance [21].

Deep Face was trained on a large dataset of faces acquired from a sample that was significantly different from the people used to create the evaluation targets, and it outperformed existing methods with very slight adaptations [22]. The suggested method is different from the bulk of previous submissions in that it implements the deep learning (DL) model. With several significant advancements in different domains such as vision, voice, and language modeling, DL is particularly well suited for working with wide training sets [23]. The Convolutional neural network models combine the feature extractor and classifier from beginning to end. The techniques produce are useful in face representations. Many methods to improving face verification performance in an environment have been proposed, and some of them have shown positive results. Face identification makes extensive use of metric learning techniques, which are often combined with task-specific goals. MTL improves the performance at the same time. Face detection [24], Face alignment [25], pedestrian detection [26], attribute estimation [27], and other applications have all proven effective. Despite MTL's effectiveness in a variety of visual difficulties, there hasn't been much research done on it for face recognition.

For many facial analysis tasks, MTL algorithms are applied. Reference [28] was the first to examine the MTL framework for machine learning in-depth. It has since been utilized to solve a variety of computer vision challenges. Reference [29] offered one of the first MTL-based facial analysis studies. The method was used to solve face recognition, feature identification, and head-pose estimation. It combined a tree model with a pool of common components, each of which symbolizes a landmark place. JointCascad is another technique [28]. Combining the training with the landmarks localization challenge enhanced the face identification algorithm. Both of these systems relied on handmade components, making it difficult to apply the Multi-Task Learning paradigm to a wide range of activities. Because the feature representations required to accomplish each job were different, multi-task learning was limited to a few sets of tasks before the introduction of deep-learning algorithms. Face detection, for example, is often performed with HOG, whereas face recognition is often performed with LBPs. Likewise, distinct handmade characteristics were developed for age, gender, landmark localization, and attribute categorization. Handwritten characteristics were replaced by deep convolutional neural network features for facial analysis tasks as deep learning progressed. This allowed a single deep convolutional neural network model to be trained for a variety of tasks, including face detection, landmark localization, facial attribute prediction, and face recognition. When designing computers to execute various jobs, it's common to create separate algorithms for each and by exchanging deep features, one may create a neural network that can concurrently fulfill all of the requirements.

MTL is interpreted as a regularization strategy for a DCNN by [30]. Because the learned parameters of a network agree with all of the tasks at hand, the MTL methodology prevents overfitting and converges to a robust solution. Machine learning [31], and computer vision [22] have both done extensive research on MTL. To study common features and task-specific models, it is appropriate to combine MTL and CNN. For combined face identification, posture estimation, and landmark localization, [25] offers a deep CNN. Several techniques for training face-related tasks have recently combined the MTL framework with deep CNNs. Reference [32] advocated employing a CNN to simultaneously estimate age and gender. Hyper Face [33] trained an MTL network for face identification, landmark localization, posture, and gender estimation by combining the intermediate layers of CNN for better feature extraction. Reference [34] proposed an MTL-restricted Boltzmann machine for learning facial features, whereas reference [25] improved landmark localization by simultaneously training it with head-pose estimation and face attribute inference. These methods can perform MTL on a small number of tasks, but they are unable to train a large number of related tasks. Individual face analysis tasks have been the subject of much investigation. DP2MFD [33], Faceness [35], Hyperface [33], Faster-RCNN [36], and other deep CNN-based face detection algorithms have considerably outperformed classic algorithms like TSM [29] and NDPFace [37]. Due to a shortage of training data, only a few approaches have exploited deep CNNs for face alignment tasks [25-38]. Recent landmark localization algorithms have mostly focused on near-frontal faces [39], which have all of the key points visible. Recent techniques such as PIFA, 3DDFA, Hyperface, and CCL have been used to investigate face alignment.

Methodology

Face Detection

In the fields of image processing and pattern recognition, human face detection has proved to be a challenging problem. This step determines whether or not the given image contains any human faces. Variations in lighting and facial expression may make face identification difficult. A face recognition system's capacity to deal with images and videos, real-time, robust lighting situations and function with faces from various viewpoints are all aspects that make it successful. Furthermore, due to the lack of limitations on the quantity, position, size, and alignment of faces in an image or video scene, facial feature extraction is a time-consuming procedure. The proper method is to gather a large number of face and non-face samples, then use machine learning methods to train a face model for categorization. In this procedure, there are two major concerns: which characteristics to extract and which learning algorithm to use. Finally, the face recognition process compares the obtained attributes from the human face to all template face databases to determine human face identification. The three main approaches required to construct a decent face recognition system are face detection, feature extraction, and face recognition [40]. To facilitate the building of a more powerful facial recognition system easier, pre-processing methods are used. Only a few of the methods used to recognize and identify face images include the Viola-Jones detector [41], histogram of directed [42], and PCA [43].

Each face has its structure, size, and shape, allowing it to be identified. Some methods for identifying the face based on size and distance include extracting the contour of the lips, eyes, or nose [44]. HOG [45], Eigenface [46], independent component analysis, linear discriminant analysis [43], scale-invariant feature transform [47], Gabor filter, local phase quantization [48], Haar wavelets, Fourier transforms [49], and local binary pattern [44] are some of the techniques used to extract face features. In the feature extraction stage, features are extracted from the environment and use to identify objects recorded in a database. Face recognition is used for identification as well as verification. A test face is compared against a group of faces to identify the most likely match throughout the identification process. During the identification stage, a test face is compared to a known face in the database to make the acceptance or rejection decision [45]. The success of correlation filters [50], convolutional neural networks [51], and k-nearest neighbor [52] has been demonstrated.

Figure 1 shows the design of face detection and face feature extraction system. The modules, face detection, and feature extraction use in this paper.

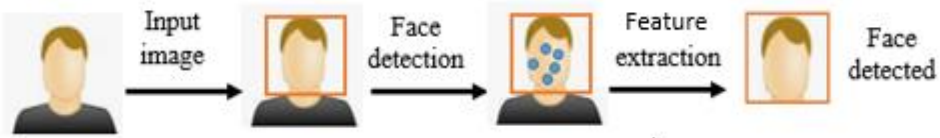

Fig 1 Face Detection and Feature Extraction model

In figure 1 Face detection is used in all facial recognition systems, however not all face detection methods are used for facial recognition. using cameras a human's facial movements convert into a digital database or face detection.

The initial step is to choose the digital image that will be used to apply the algorithms. The chosen image is referred to as the input image, as illustrated in Fig1. Face detection is the second phase, which involves determining whether or not there are any faces in a given image and, if so, returning the image position and content of each face. After the face detection, the face extraction procedure is carried out in order to offer relevant information for differentiating between various people's faces. Finally, a result is displayed, indicating that an image has been identified.

Face Detection Algorithms

a) Knowledge-based algorithms:

Describe in code the features that make up a real face, as well as the relationships between them, and make it simple to put up basic rules to differentiate between different face types. Face features are detected first in an image using programmed rules. Furthermore, covering all possible situations is challenging.

b) Feature invariant algorithms:

It includes a group of features that are similar to all faces and do not vary in response to changes in the environment, such as posture and light. Edge, intensity, size, texture, and color are all common features. These algorithms have a difficult task in identifying features in image corruption scenarios such as noise, lighting, and occlusion, as well as in complicated backgrounds.

c) Matching algorithms:

Different face formats are described by these algorithms. Face locations are usually specified based on edges, and correlation is used to identify them. They must be deformable to detect and adjust to the contours of the face. The primary disadvantage of knowledge-based algorithms is that they must cover all potential postures.

d) Appearance-based algorithms:

Create face templates using statistical and machine learning approaches such as classifiers, which include neural networks, PCA, and SVM, among others. Positive face samples, as varied as appropriate, and negative examples, with representative images that do not contain faces, are often used in the training of these algorithms.

e) Viola-Jones classical algorithm:

Viola-Jones algorithm has a higher detection speed in real-time face detection. The main characteristics of this algorithm are given below:

- By using the integral map to ensure the speed at which the feature is extracted value of the face image is extracted and, secondly, by using the Ada-Boost strong classifier, the correct rate of face detection is achieved.
- Modification of the standard Ada-Boost classifier. In the modern Ada-Boost classifier, several small decision trees construct the weak classifier and then cascade into a strong classifier. The Viola-Jones algorithms classifier for Ada-Boost cascade processing. A strong classifier is each new classifier constructed from this, which ensures a high rate of detection. Although face detection is the first step in any computerized system which solves face detection, facial expression, face tracking problems.

There are different ways of detecting faces and using these strategies, it is possible to identify faces with better accuracy. These methods, such as OpenCV, Neural Networks, Matlab, Python, and others, have the nearly same procedure for Face Detection. Face detection is the process of detecting numerous faces in an image. Using OpenCV to recognize faces, and there are a few steps to how to face detection works, which are as follows:

- The image is first imported by specifying the image's location. The image is next converted from RGB to Grayscale since faces are easier to identify in grayscale.
- The Haar-Like features method, introduced by Viola and Jones for face identification, is the next step. The position of human faces in a frame or picture is determined using this method.

Due to the integral image method, the HaarCascade features are highly efficient to calculate and give a better result for developing frontal face detectors. Researchers enhanced the simple characteristics with more variants in the ways rectangular features are combined in several follow-up studies. OpenCV is used to implement the face detection method. However, the model should obtain before the test phase of human face detection at the stage. Haar-like characteristics are used in a cascade of classifiers. The cascade classifier that comes with OpenCV 3.3.0 will be applied, thus no further training is required.

Results and Discussion

The collecting of high-resolution face expressions required the verification of data and computer expertise. The camera records information in a way that is directly connected to the camera. Automatic data entry may be made available with the support of integration, which improves the efficiency of data collecting. To complete the image segmentation, a face image collection algorithm relies on OpenCV and a frontal face recognition algorithm, data transformation, analysis of the changed data, and image classification by analysis of the parameters were implemented. The entire system design is separated into the image area. To use OpenCV's

"HaarCascade frontal face" file for image pre-storage, a C++ environment is required. The target file's information is identified, evaluated, and transforming the image into digital information. The digital image data is converted into an easy-to-store and-retrieve coding format. Using the OpenCV function cvHaarDetectObjects, which identifies an image from the target area, the pointer is used to execute cascade classifiers with varied parameters to retrieve different region information.

Above mentioned all detectors are used for Python-based implementation. Several hundred images are obtained by different cameras that include Webcam, CCTV 4K, and Mobile cameras. The image size is between 300 KB to 3.4 MB. All face recognition demos available online are modeled on a webcam that runs very fast.

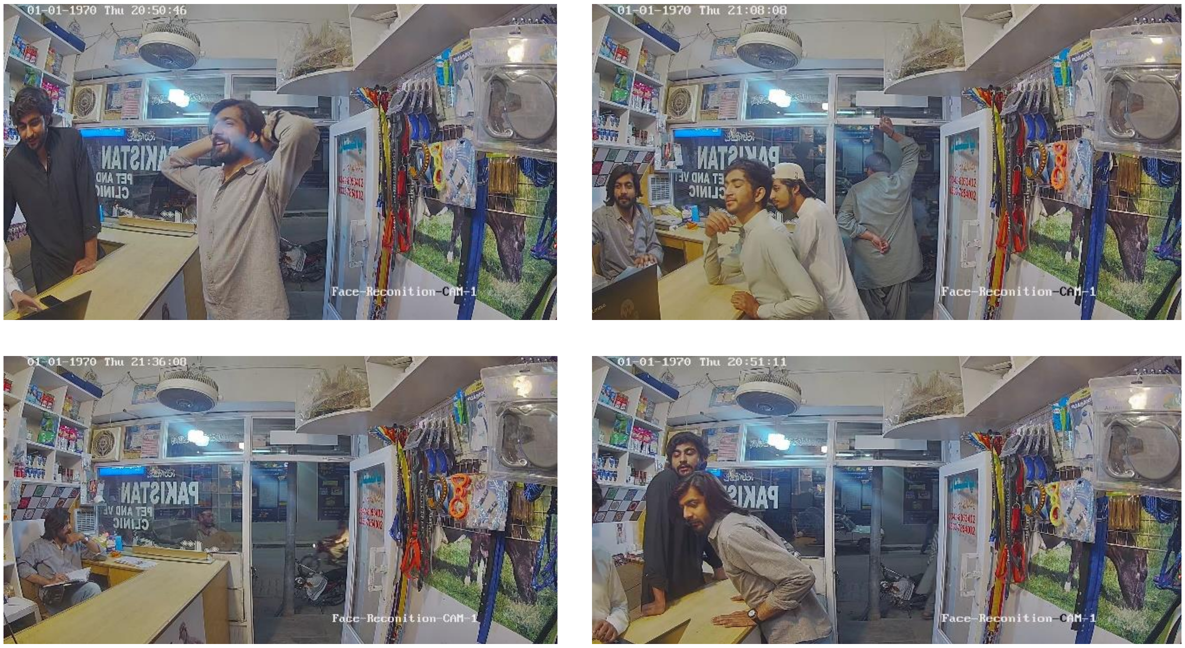

Fig 2 Face recognition model installed in a shop.

Figure 2 shows four random pictures of the shop where the face recognition module is installed. In the real world, webcams are not used due to the poor quality of images. Therefore CCTV or HD cameras are installed in an optimal location for better face reading. The same demo programs give very low accuracy and speed when used with HD and CCTV cameras.

The system has been tested and evaluated and the following is how the database's data was gathered. The camera captured images of a group of people from which it is used to detect the images. Three algorithms (DLIB facial detector, Haar-Cascade face detector, and MTCNN facial detector) attempted to recognize faces based on these images. The MTCNN technique works by using multi-scale transformation to produce many rectangular frames of various sizes in the face data, performing sliding detection on the face picture, and extracting and detecting the characteristics of the rectangular frames.

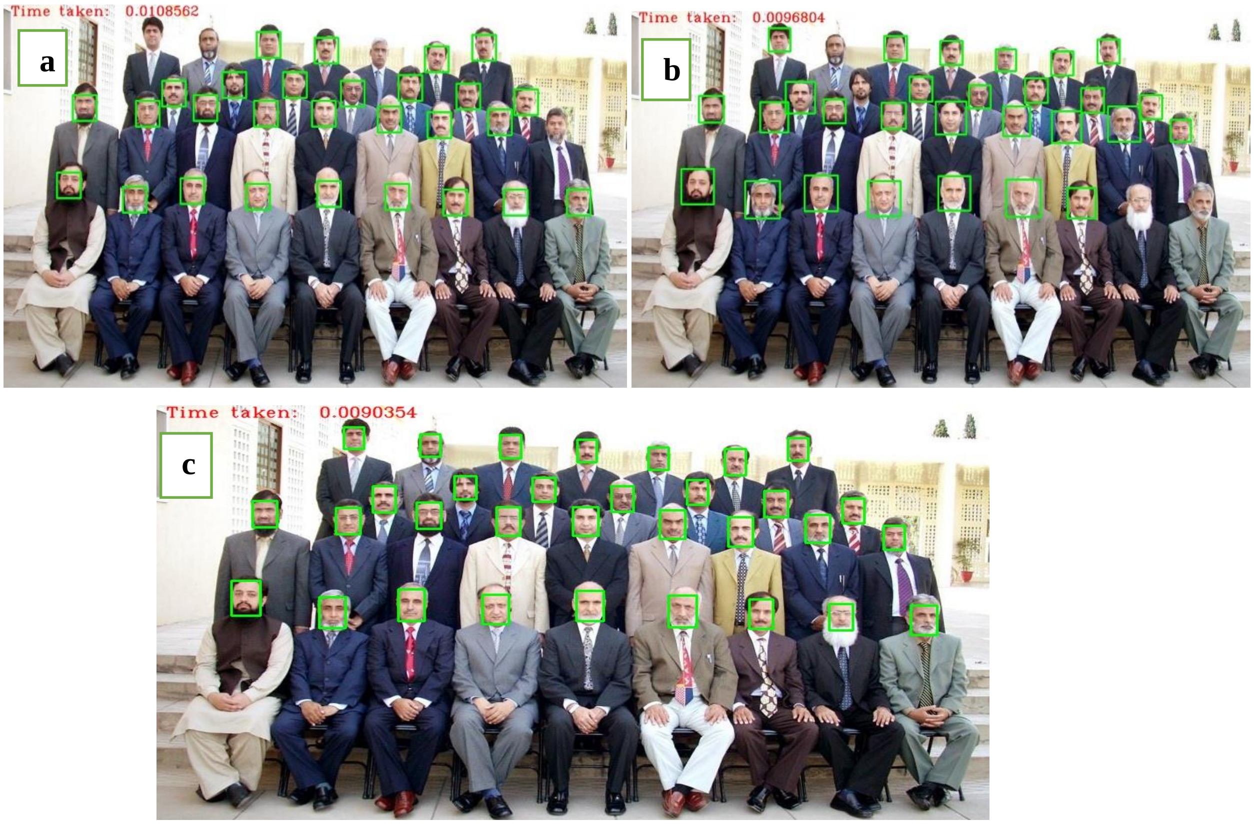

Fig 3 Qualitative detection accuracy of face detectors, a) shows Dlib, b) shows Haar Cascade, and c) shows MTCNN

Figure 3 shows the detection accuracy of face detectors used for experiments. All detectors are run based on their default configuration. It can be seen that MTCNN accuracy is better than the other two detectors on the default configuration. However, DLIB configuration if changed can get all faces to detect but with many false positives. The DLIB achieves good precision on default configuration with low recall.

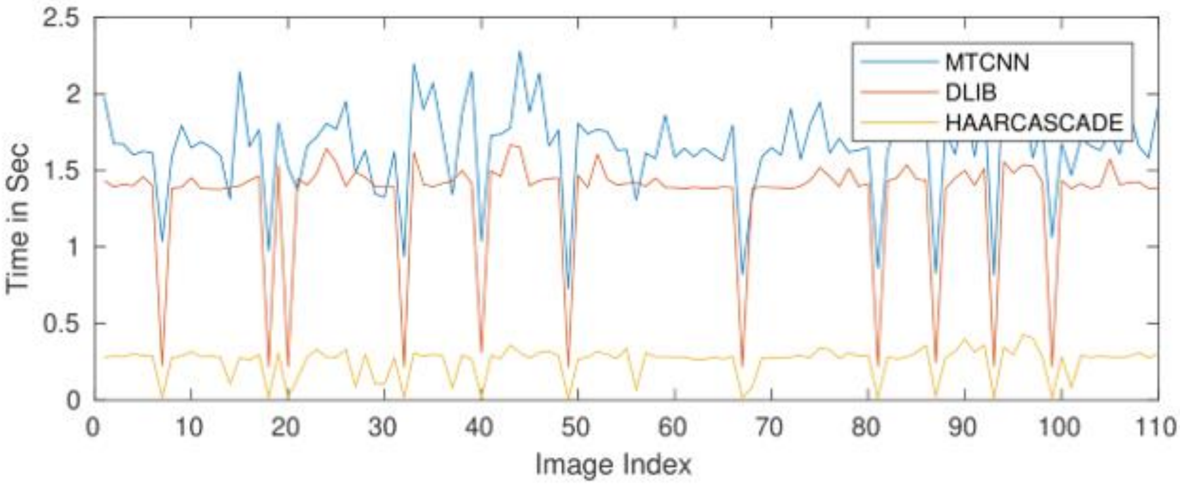

Fig 4 Detection time of face detectors.

Figure 4 shows the graph of different detectors. This graph is generated by using the images with different resolutions and sizes which is obtained from different cameras. The experiment is conducted to detect the faces on the default configuration.

The graph in figure 4 also shows that the HAAR cascade is the fastest on the default configuration. However, experiments also show that HAAR cascade precision is low compared to DLIB and MTCNN. The DLIB detector uses HOG based detector for CPU computation.

Conclusion

In this research, famous three face detectors are evaluated against the efficiency matrix, where time is the prime parameter. The detector includes HAAR cascade, DLIB, and MTCNN. All these detectors are available for all famous languages such as C, C#, Matlab, and Python. For experiments, we have used Python-based implementation. Experiments show that the HAAR cascade has better performance on the default configuration. However, the precision of the HAAR cascade is lower compared to DLIB but with better recall. The DLIB detector is widely used for face recognition applications as lower recall misses small faces on the default configuration.

Author's Contribution: J.B did the language and grammatical edits, Critical revision manage. M.B, Conceived the idea, designed the work, K.A.M executed simulated work, data analysis, and interpretation of data.

Funding: The publication of this article was funded by no one.

Conflicts of Interest: The authors declare no conflict of interest.

Acknowledgment: The authors would like to thank the Computer Science and IT Department for arrangement of hardwares and assistance with the collection of data.

REFERENCES

[1] P. Viola and M. Jones, “Robust real-time face detection,” in Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001, 2005.

[2] A. Benzaoui, H. Bourouba, and A. Boukrouche, “System for automatic faces detection,” in 2012 3rd International Conference on Image Processing Theory, Tools and Applications (IPTA), 2012. https://doi.org/10.1109/IPTA.2012.6469545

[3] D. Yang, P. W. C. P. Abeer Alsadoon, A. K. Singh, and A. Elchouemi, “An Emotion Recognition Model Based on Facial Recognition in Virtual Learning Environment”,” Procedia Computer Science, vol. 125, pp. 2–10, 2018. https://doi.org/10.1016/j.procs.2017.12.003

[4] V. Mohanraj, M. Vimalkumar, M. Mithila, and V. Vaidehi, “Robust face recognition system in video using hybrid scale invariant feature transform,” Procedia Comput. Sci., vol. 93, pp. 503–512, 2016. https://doi.org/10.1016/j.procs.2016.07.240

[5] A. Vinay et al., “Face recognition using filtered EOH-sift,” Procedia Computer Science, vol. 79, pp. 543–552, 2016. https://doi.org/10.1016/j.procs.2016.03.069

[6] A. H. Abdulnabi, G. Wang, J. Lu, and K. Jia, “Multi-task CNN Model for Attribute Prediction,” arXiv [cs.CV], 2016.https://doi.org /10.1109/TMM.2015.2477680

[7] C. Ding, C. Xu, and D. Tao, “Multi-task pose-invariant face recognition,” IEEE Trans. Image Process., vol. 24, no. 3, pp. 980–993, 2015. https://doi.org /10.1109/TIP.2015.2390959

[8] J. Yim, H. Jung, B. Yoo, C. Choi, D. Park, and J. Kim, “Rotating your face using multi-task deep neural network,” in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[9] G. Bradski and A. Kaehler, Learning OpenCV: Computer vision with the OpenCV library. O’Reilly Media, Inc. ", 2008.

[10] S. Liao, A. K. Jain, and S. Z. Li, “A fast and accurate unconstrained face detector,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 2, pp. 211–223, 2016.https://doi.org /10.1109/TPAMI.2015.2448075.

[11] Y. Kortli, M. Jridi, A. A. Falou, and M. Atri, “Face recognition systems: A survey,” Sensors (Basel), vol. 20, no. 2, p. 342, 2020. https://doi.org/10.3390/s20020342

[12] A. Vinay et al., “Face recognition using filtered EOH-sift,” Procedia Computer Science, vol. 79, pp. 543–552, 2016. https://doi.org/10.1016/j.procs.2016.03.069

[13] M. Xi, L. Chen, D. Polajnar, and W. Tong, “Local binary pattern network: A deep learning approach for face recognition,” in 2016 IEEE International Conference on Image Processing (ICIP), 2016.

[14] K. Bonnen, B. F. Klare, and A. K. Jain, “Component-based representation in automated face recognition,” IEEE trans. inf. forensics secur., vol. 8, no. 1, pp. 239–253, 2013.

[15] J. Ren, X. Jiang, and J. Yuan, “Relaxed local ternary pattern for face recognition,” in 2013 IEEE International Conference on Image Processing, 2013.

[16] M. Karaaba, O. Surinta, L. Schomaker, and M. A. Wiering, “Robust face recognition by computing distances from multiple histograms of oriented gradients,” in 2015 IEEE Symposium Series on Computational Intelligence, 2015.

[17] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, “Labeled faces in the wild: A database for studying face recognition in unconstrained environments,” 2008.

[18] G. B. Huang, V. Jain, and E. Learned-Miller, “Unsupervised joint alignment of complex images,” in 2007 IEEE 11th International Conference on Computer Vision, 2007.

[19] G. Huang, M. Mattar, H. Lee, and E. G. Learned-Miller, “Learning to align from scratch,” 2012, pp. 764–772.

[20] S. Yang, P. Luo, C. C. Loy, and X. Tang, “WIDER FACE: A face detection benchmark,” in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[21] H. Jiang and E. Learned-Miller, “Face detection with the faster R-CNN,” in 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017), 2017.

[22] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, “DeepFace: Closing the gap to human-level performance in face verification,” in 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014.

[23] Y. Bengio, Learning Deep Architectures for AI. Hanover, MD: now, 2009.

[24] C. Zhang and Z. Zhang, “Improving multiview face detection with multi-task deep convolutional neural networks,” in IEEE Winter Conference on Applications of Computer Vision, 2014. https://doi.org /10.1109/WACV.2014.6835990

[25] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, “Facial landmark detection by deep multi-task learning,” in Computer Vision – ECCV 2014, Cham: Springer International Publishing, 2014, pp. 94–108.

[26] Y. Tian, P. Luo, X. Wang, and X. Tang, “Pedestrian detection aided by deep learning semantic tasks,” in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015. https://doi.org/ 10.1007/978-981-10-3002-4_17

[27] A. H. Abdulnabi, G. Wang, J. Lu, and K. Jia, “Multi-Task CNN Model for Attribute Prediction,” IEEE Trans. Multimedia, vol. 17, no. 11, pp. 1949–1959, 2015.https://doi.org/ 10.1109/TMM.2015.2477680

[28] R. Caruana, “Multitask Learning,” in Learning to Learn, Boston, MA: Springer US, 1998, pp. 95–133. https://doi.org/ 10.1007/978-3-030-37599-7_50

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

[30] Www.deeplearningbook. [Online]. Available: http://www.deeplearningbook. [Accessed: 17-Jun-2021].

[31] P. Gong, J. Zhou, W. Fan, and J. Ye, “Efficient multi-task feature learning with calibration,” in Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD ’14, 2014. https://doi.org/10.1145/2623330.2623641

[32] G. Levi and T. Hassncer, “Age and gender classification using convolutional neural networks,” in 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2015.

[33] R. Ranjan, V. M. Patel, and R. Chellappa, “Hyperface: A deep multi-task learning framework for face detection. landmark localization, pose estimation, and gender recognition.” 2016.

[34] M. Ehrlich, T. J. Shields, T. Almaev, and M. R. Amer, “Facial attributes classification using multi-task representation learning,” in 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2016.

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

[36] H. Jiang and E. Learned-Miller, “Face detection with the Faster R-CNN,” arXiv [cs. CV], 2016.

[37] A. Kumar, R. Ranjan, V. Patel, and R. Chellappa, “Face alignment by Local Deep Descriptor Regression,” arXiv [cs. CV], 2016.

[38] S. Liao, A. K. Jain, and S. Z. Li, “A fast and accurate unconstrained face detector,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 2, pp. 211–223, 2016.https://doi.org/ 10.1109/TPAMI.2015.2448075

[39] S. Ren, X. Cao, Y. Wei, and J. Sun, “Face alignment at 3000 FPS via regressing local binary features,” in 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014.

[40] A. Vinay, D. Hebbar, V. S. Shekhar, K. B. Murthy, and S. Natarajan, “Two novel detector-descriptor-based approaches for face recognition using sift and surf,” Procedia Comput. Sci, vol. 70, pp. 185–197, 2015. https://doi.org/10.1016/j.procs.2015.10.070

[41] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, 2005. https://doi.org/10.1109/CVPR.2001.990517

[42] A. Alfalou, Y. Ouerhani, and C. Brosseau, “Road mark recognition using HOG-SVM and correlation,” in Optics and Photonics for Information Processing XI, 2017. https://doi.org/10.1117/12.2273304

[43] H. J. Seo and P. Milanfar, “Face verification using the LARK representation,” IEEE trans. inf. forensics secure., vol. 6, no. 4, pp. 1275–1286, 2011. https://doi.org/ 10.1109/TIFS.2011.2159205

[44] T. Napoléon and A. Alfalou, “Pose invariant face recognition: 3D model from single photo,” Opt. Lasers Eng., vol. 89, pp. 150–161, 2017. https://doi.org/10.1016/j.optlaseng.2016.06.019

[45] Q. Wang, D. Xiong, A. Alfalou, and C. Brosseau, “Optical image authentication scheme using dual-polarization decoding configuration,” Opt. Lasers Eng., vol. 112, pp. 151–161, 2019. https://doi.org/10.1016/j.optlaseng.2018.09.008

[46] Y. W. Y. Jia and C. H. M. Turk, “Fisher non-negative matrix factorization for learning local features,” 2004, pp. 27–30.

[47] Vinay, D. Hebbar, V. S. Shekhar, K. N. B. Murthy, and S. Natarajan, “Two novel detector-descriptor based approaches for face recognition using SIFT and SURF,” Procedia Comput. Sci., vol. 70, pp. 185–197, 2015. https://doi.org/10.1016/j.procs.2015.10.070

[48] S. U. Hussain, T. Napoléon, and F. Jurie, “Face Recognition using Local Quantized Patterns,” in Proceedings of the British Machine Vision Conference 2012, 2012.

[49] F. Smach, J. Miteran, M. Atri, J. Dubois, M. Abid, and J.-P. Gauthier, “An FPGA-based accelerator for Fourier Descriptors computing for color object recognition using SVM,” J. Real-Time Image Process., vol. 2, no. 4, pp. 249–258, 2007.

[50] A. Alfalou and C. Brosseau, “Understanding correlation techniques for face recognition: From basics to applications,” in Face Recognition, InTech, 2010. https://doi.org/10.1016/j.optcom.2013.07.071

[51] F. Schroff, D. Kalenichenko, and J. Philbin, “FaceNet: A unified embedding for face recognition and clustering,” in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[52] I. L. Kambi Beli and C. Guo, “Enhancing face identification using local binary patterns and k-nearest neighbors,” Journal of Imaging, vol. 3, no. 3, p. 37, 2017. https://doi.org/10.3390/jimaging3030037