**Comparative Study of Face Detection Models**

Saurabh Aggarwal, Michael Cruz, Rafeeq Shodeinde

December 6, 2023

**Abstract**

With the proliferation of facial detection applications in various domains, the evaluation of pretrained face detection models has become imperative to ensure their reliability and effectiveness in real-world scenarios. This paper presents a comprehensive analysis of diverse pretrained face detection models, emphasising their performance metrics in terms of precision, recall, and processing speed. By benchmarking these models across varied datasets that simulate real-world conditions, we discern their strengths, weaknesses, and potential suitability in some applications. Our research aims to guide developers, researchers, and industry practitioners in making informed decisions about model selection and customization. The outcomes of this evaluation have direct implications for the design and implementation of facial detection systems, influencing fields such as security, human-computer interaction, and social media. Through this work, we want to provide insights that performance of these models.

**1. Introduction**

In recent years, the rapid advancement of computer vision technologies has propelled the field of facial detection to the forefront of research and application development. The ability to automatically identify and locate faces within images or video streams has found widespread applications in diverse domains, including security surveillance, human-computer interaction, social media, and even healthcare. As the demand for sophisticated facial detection systems continues to grow, the evaluation of pretrained face detection models becomes a critical task, influencing the efficiency and reliability of these applications.

The fundamental challenge addressed by face detection models lies in the inherent complexity of the task – distinguishing and localising faces in images that exhibit variations in scale, pose, illumination, and occlusion. As a result, the development of accurate and robust face detection models has become an area of intense research, with numerous pretrained models emerging as state-of-the-art solutions and the evaluation of these models is essential to assess their real-world applicability, reliability, and performance across diverse scenarios. Being able to gain and understand the nuances of pretrained models is crucial for guiding developers, researchers, and industry practitioners in making informed decisions about model selection and customization for specific applications.

This paper aims to systematically evaluate and compare the performance of various pretrained face detection models, shedding light on their strengths, weaknesses, and suitability for specific use cases. By employing rigorous benchmarking and analysis of key metrics, such as accuracy, precision, and processing speed, we seek to provide valuable insights into the practical implications of adopting different pretrained models.

**2. Method**

To conduct a comprehensive evaluation of pretrained face detection models, we selected a diverse set of widely recognized models, each representing distinct methodologies in the realm of computer vision. The chosen models include the classical Haar cascade classifier, the robust dlib CNN model, Multi-task Cascaded Convolutional Networks (MTCNN), dlib HOG, RetinaFace, and OpenCV CNN model. This ensemble was deliberately curated to encompass a spectrum of approaches, ranging from traditional feature extraction based methods to state-of-the-art deep learning architectures. For benchmarking and comparative analysis, we utilised a rich dataset comprising images sourced from the Labelled Faces in the Wild (LFW) dataset and other image datasets. Additionally, to simulate dynamic and challenging scenarios, we incorporated video data from the Condensed Movies Dataset, further augmenting the scope of our evaluation.

Amongst the pretrained face detection models publicly available, the dlib CNN model emerges as a pivotal player, harnessing the power of Convolutional Neural Networks (CNNs) to excel in facial feature extraction and localization. CNNs have become instrumental in computer vision tasks, particularly in face detection, due to their capacity to automatically learn hierarchical representations from raw pixel data. This model is built on the principles of Max-Margin Object Detection (MMOD), a framework designed to optimise the accuracy of object detection tasks. At its core, the MMOD-based architecture of the dlib CNN model transforms face detection into a binary classification problem. The network is trained to discern between positive instances, where a face is present, and negative instances, where no face is detected.

The OpenCV’s Deep Neural Network (DNN) module stands out for its robust facial detection capacity in both image and video. It is capable of dealing with diverse orientations and expressions in high resolution stills. In spite of a few challenges like motion blur and variable lighting, the accuracy still is observed to be high. It has the ability to process real-time video streams which makes it a good candidate for video surveillance and live audience analysis.

RetinaFace is a highly efficient and accurate facial detection algorithm. RetineFace is designed to recognize and locate human faces within digital images. It employs a convolutional neural network (CNN), particularly adept at analysing visual imagery. What sets RetineFace algorithm apart from other such video detection algorithms is the ability to recognize not just faces but also key facial landmarks, such as the eyes, mouth, and nose etc. This makes it very suitable for applications requiring highly precise facial recognition, like biometric authentication, or video/photo editing.

Another quality of RetinaFace is its scalability and efficiency. It can easily be used for real-time applications like video surveillance etc, because of its high processing capabilities.

Histogram of Oriented Gradients, is a widely used algorithm in the domain of computer vision for object detection, with a notable application in face detection as well. HOG involves analysing the distribution and direction of gradients or edge directions in an image. The image is basically divided into small connected cells. These histograms effectively capture the shape and texture information of the captured cell. Subsequently, these histograms are normalised over larger, overlapping blocks of cells to enhance accuracy and adjust for variations in illumination and shadow. This process makes the algorithm robust even in different lighting conditions. In the context of face detection, the HOG algorithm analyses these descriptors to identify patterns that resemble facial features.

The Haar Cascade algorithm for face detection was published in 2018 and is one of the most widely used face detection algorithms today. This algorithm is based on the work of Viola and Davis, who proposed a technique that utilised a cascade classifier algorithm incorporating Haar’s like features to solve the problem of facial detection . By boosting this algorithm via AdaBoost, Viola Jones cascade was able to achieve a high detection rate at the cost of an increased false negative rate. Using this method as a basis, the Haars Cascade algorithm added three weak classifiers to overcome the weaknesses of the previous model. One weak classifier “One is a decision node based on human skin hue histogram matching because the skin tone for a specific race of people has a relatively determined distribution. The second and the third weak classifiers are based on eyes and mouth detections. Because eyes and mouth detections are also implemented with Haar-like features based cascade classifiers, both of them have a sufficiently high detection rate, satisfying conditions of weak classifiers.(Cuimei & Zhiliang, 2018)”.

Multi-task Convolutional Neural Network (MTCNN) is a powerful algorithm for face detection as well as facial expression recognition. The MTCNN algorithm is implemented with three key parts. “[(1)](https://ieeexplore.ieee.org/document/#deqn1-2) a Proposal Network (P-Net) for generating a list of the candidate windows. Then, we use it for classifying face and non-face and estimate bounding box regression vectors to face location, and non-maximum suppression (NMS) candidate merge. [(2)](https://ieeexplore.ieee.org/document/#deqn1-2) a Refine Network (R-Net), unlike P-Net, doesn't obtain the region proposals but rejects a lot of false candidates. [(3)](https://ieeexplore.ieee.org/document/#deqn3) It is similar to R-Net, called O-Net.(Xiang & Zhu, 2017)” By utilising a deep neural network and inherent correlation between detection and alignment, the model produced low validation accuracy, but still accomplishes relatively fast detection speeds.

**3. Experiments**

**3.1 Dataset**

For our evaluation, we harnessed the dynamic nature of video datasets and the richness of image datasets to comprehensively assess the pretrained face detection models. Video datasets, such as the Condensed Movies dataset, containing a variation of movies produced in different years, served as good resources, simulating real-world scenarios with varying lighting conditions, facial poses, and occlusions. This dynamic environment enabled us to gauge the models' robustness in tracking faces across frames, potentially useful in practical applications like surveillance and video analytics. In place of a singular emphasis on a specific dataset, our evaluation strategy is centred on a judicious random sampling across various image datasets. This approach was intentionally designed to give a difference of scenarios into our assessment, encompassing an array of facial expressions, lighting conditions, and contextual nuances. By drawing from multiple datasets, our evaluation aimed to present a nuanced and representative analysis of pretrained face detection models in diverse real-world applications.

**3.2 Evaluation metrics**

Throughout the evaluation, our analysis hinged on three key performance metrics—accuracy, precision, and processing time—each providing unique insights into the effectiveness of pretrained face detection models. Accuracy served as a fundamental measure of the models' ability to correctly identify faces, offering a comprehensive view of their overall performance. To calculate the processing time of the models for different data input, each algorithm took for completion, a time average was calculated. For some models the Positive predictive value (PPV) was used to measure the accuracy of true positives detected while taking into account false positives. PPV is defined as:

**3.3 Analysis and discussions**

**3.3.1 Haar Cascade**

The main advantage to using the Haar Cascade algorithm is fast execution when detecting faces. Compared to other algorithms, Haars Cascade performed face detection faster than most algorithms across all mediums. When detecting faces in images, on a dataset of 20 images, each image was processed on average 0.204 seconds per image. For a 2:35 minute video, the algorithm only took 8.11 seconds to detect faces in every frame. The impressive speed of the algorithm comes at the cost of its effectiveness in detecting all faces in an image or a frame. On average, the Haars Cascade algorithm had an accuracy of 61% on facial detection when considering all the faces present in an image, versus the faces correctly detected by the algorithm. However, the PPV for the algorithm was 98.7% meaning there were few false negatives among the faces correctly predicted by the model. Due to the manner in which Haar Cascade was implemented, only faces which are directly front facing can be detected, which drastically limits the algorithm's capability, especially in video and real time settings. When experimenting on video data, the algorithm was unable to continuously detect faces from frame to frame, especially if the subjects turned in a direction to which they were no longer directly facing the recording aparatas.

**3.3.2 MTCNN**

The main performance benefit of the MTCNN is the ability to detect the majority if not all faces in all mediums during experimentation. Taking into account different scales and aspect ratios, the algorithm correctly identified most faces in images and videos, for all faces with the condition that facial features must be in full focus and unblurred. While performing face detection on image data, MTCNN had an 80% accuracy rate in detecting true positives. Likewise, on video data, the algorithm could maintain bounding boxes around the subjects regardless of their orientation in the frame. However, MTCNNs have a high computational cost and require a much longer time period to detect faces in all mediums. In the case of images, the algorithm took on average 1.7 seconds to process each image. Subsequently, a 2:35 minute video took 41 minutes and 17 seconds to process. MTCNNs are unable to be used on real-time data because frame processing takes too long which results in rendering with very low throughput. Additionally, the MTCNN produces a large number of double bounding boxes. This is most likely due to the non-max suppression algorithm employed having a threshold value which is too high.

**3.3.3 Dlib CNN**

In terms of the accuracy measure for the Dlib model, the recorded value ranged from 90-95% for the video and image inputs. This model was designed to handle a robust set of scenarios and occlusions and especially fitted to run efficiently on the GPU card. This brings up the consideration for the computational speed of this model asides other things. The model was evaluated on both CPU and GPU to see the difference in performance which was rather obvious. The CPU processing time for a two and half minute video was ranging 6 minutes to finish processing and a little over 8 minutes when the frames of the videos were being shown simultaneously.The GPU on the other hand was able to display the frames of the video, while applying the face detector, as it would a normal video. There were no lags which were occurring when the CPU was used. There were points where the CPU was able to improve on speed by resizing to smaller frames before passing as the inputs, but would still be jumpy and not match the GPU results. Along with accuracy, the dataset of 20 images were used for evaluating the performance measure of precision, exhibiting high precision overall for the dataset. The sample of the image shown in section 3.4 provides results for the performance of the MMOD CNN model trained on faces.

**3.3.4 HOG**

The Histogram of Oriented Gradients (HOG) algorithm stands out for its effectiveness in facial recognition, especially in scenarios with high-quality imaging and distinct facial features. In our assessment, HOG exhibited an 80-90% accuracy rate for facial recognition in both videos and images. Although it doesn't match the accuracy of more sophisticated models, HOG's minimal computational requirements and efficiency in real-time processing makes it a valuable tool in resource-constrained environments.

**3.3.5 RetinaFace**

RetinaFace is a state-of-the-art facial detection model known for its high accuracy and efficiency. In our analysis, RetinaFace reported an accuracy of 85-95% for facial recognition in both images and videos. In many cases the accuracy was observed as high as 99%. This high level of accuracy can be attributed to its deep learning framework, which allows it to effectively handle a wide range of facial orientations, expressions, and lighting conditions. RetinaFace performs really well in scenarios involving challenging angles and partial occlusions, making it a robust choice for diverse applications. Its performance in detecting faces in high-resolution images and videos is particularly commendable, and it is capable of identifying faces with varying scales and aspect ratios. The advanced features of RetinaFace, for instance its ability to detect facial landmarks (such as eyes, nose, and mouth), enhance its precision in facial recognition tasks. However, the high accuracy comes with the cost of increased computational resources and time, making it more suitable for systems where processing power is not a major constraint. Said that, RetinaFace is not an ideal candidate for realtime facial detection application.

**3.3.6 OpenCV DNN**

The OpenCV model is specifically designed to run efficiently on the CPU resulting in a smooth flowing video stream. The model performed very fast and very well, processing the two and half minute video in a little over 3 minutes while not lagging or at least not as visible to the human eyes. It was good enough to watch the movie while detection was applied. A downside to the dlib model when applied to videos were instances where false negatives arise. Certain frames of the videos will highlight areas not containing faces with bounding boxes affecting the evaluation results of the model. The accuracy measured for this model ranged 70-80% of faces present in a frame of the video. For the image dataset, a noticeable factor was the non presence of the false negatives that were occurring in the video files when they were parsed.

**3.4 Results**

**DLIB CNN Face Detector OpenCV DNN**

|  |  |
| --- | --- |
|  |  |

**MTCNN Haars Cascade**

|  |  |
| --- | --- |
|  |  |

**RetinaFace** **HOG**

|  |  |
| --- | --- |
|  |  |

**4. Conclusion**

In conclusion, this study offers a nuanced comparative analysis of various pretrained face detection models, emphasising the importance of aligning model selection with specific application needs. High-accuracy models like Dlib CNN and RetinaFace, with accuracies ranging from 85-95% and 90-95% respectively, are ideal for precision-intensive tasks but require high computational resources, making them unfit for real time processing. On the other hand, the Haar Cascade and HOG algorithms, despite lower accuracies of 61% and 80-90%, offer rapid processing and efficiency, suitable for real-time applications with limited computational capacity. MTCNN and OpenCV DNN however present their own balance of accuracy and efficiency, highlighting the diverse range of options available. This comparative study highlights the trade-offs between accuracy, speed, and computational demands, guiding developers, researchers, and industry practitioners in selecting the most appropriate face detection model for their specific applications, thereby enhancing the effectiveness and reliability of facial detection systems in various real-world scenarios.

**5. Contribution**

* HaarCascade\_and\_MTCNN\_detection.ipynb file which does facial detection in images, videos, and real-time rendering.
* Researching different datasets and algorithms.
* Finding performance metrics for the models.
* Writing the report.

**6. References**

L. Cuimei, Q. Zhiliang, J. Nan and W. Jianhua, "Human face detection algorithm via Haar cascade classifier combined with three additional classifiers," 2017 13th IEEE International Conference on Electronic Measurement & Instruments (ICEMI), Yangzhou, China, 2017, pp. 483-487, doi: 10.1109/ICEMI.2017.8265863.

J. Xiang and G. Zhu, "Joint Face Detection and Facial Expression Recognition with MTCNN," 2017 4th International Conference on Information Science and Control Engineering (ICISCE), Changsha, China, 2017, pp. 424-427, doi: 10.1109/ICISCE.2017.95.2

Davis E King, Max-Margin Object Detection, arXiv:1502.00046v1

D.E.King. Dlib-ml: A machine learning toolkit. Journal of Machine Learning Research 10:1755-175.

M. Bain et. al, Condensed Movies: Story Based Retrieval with Contextual Embeddings, <https://github.com/m-bain/CondensedMovies>, arXiv:2005.04208v2

J. Deng, J. Guo, E. Ververas, I. Kotsia and S. Zafeiriou, "RetinaFace: Single-Shot Multi-Level Face Localisation in the Wild," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 2020, pp. 5202-5211, doi: 10.1109/CVPR42600.2020.00525.

M. E. Wibowo, A. Ashari, A. Subiantoro and W. Wahyono, "Human Face Detection and Tracking Using RetinaFace Network for Surveillance Systems," IECON 2021 – 47th Annual Conference of the IEEE Industrial Electronics Society, Toronto, ON, Canada, 2021, pp. 1-5, doi: 10.1109/IECON48115.2021.9589577.

P. Kumar, S. L. Happy and A. Routray, "A real-time robust facial expression recognition system using HOG features," 2016 International Conference on Computing, Analytics and Security Trends (CAST), Pune, India, 2016, pp. 289-293, doi: 10.1109/CAST.2016.7914982.

D. Monzo, A. Albiol, A. Albiol and J. M. Mossi, "A Comparative Study of Facial Landmark Localization Methods for Face Recognition Using HOG descriptors," 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 2010, pp. 1330-1333, doi: 10.1109/ICPR.2010.1145.