Comparing Deep Learning Models for Automated Face Labeling

Avalon Vinella
Harvey Mudd College
CS153, Spring 2022

1. Motivation

Studying how the human gaze responds to movement in film gives us valuable insight into both visual psychology and design. For example, we may want to observe how audiences focus on the actors, or what viewers first focus on after a cut has been made in editing. In this research, it would be helpful to have a set of film sequences labeled with visually important features and movement, such as human faces and editing cuts, that can assist in establishing correlations between eye movement and what is seen on-screen. Currently, the Gaze Data for the Analysis of Attention in Feature Films [1] provides hand-labeled clips from 15 films that fulfills this need. However, hand-labeling these are time intensive, and thus inefficient for generating more sequences to study. To this end, there are several existing face detection deep learning models that will help to automate the creation of these types of data sets by identifying which frames contain faces and where they are onscreen. This study compared three different pretrained models, a Haar cascade classifier, a Multi-tasked Cascaded Convolutional Network (MTCNN), and RetinaFace [2–4], in their accuracy and efficiency at face detection in film clips.

2. Method

All of the models were pretrained and used the default hyperparameters and loss functions. Each model was run on individual frames from each clip on the XSEDE supercomputer using 2 GPUs. The output of each model on a clip was represented as two binary arrays: one that indicates whether a single face appears in each frame, and one that indicates whether multiple faces appear in each frame. These results were compared directly to the handcoded labels for each clip to determine their accuracy, which was measured by the number of frames correctly labeled overall, as well as the number of frames correctly labeled as having single or multiple faces. Additionally, the runtime for each model to process all frames of a clip, without additional image annotation, was recorded. The accuracy and speed of the three models were compared quantitatively, and were additionally qualitatively analyzed by stitching the frames annotated with bounding boxes back into a video format. As each model utilizes different architectures, all of their performances should vary.

2.1. Haar Cascade Classifier

The Haar cascade classifier is a general object detection model that can be trained for face detection. It calculates kernel-like "Haar features" within an integral rendition of the original image. It then uses the Adaboost algorithm to determine which features are most prominent in the images and trains a set of "weak" classifiers to locate them. Lastly, it then combines these weak classifiers into one "strong" classifier that can then be used for object detection. Because the Haar features are very rigid and highly specific to one type

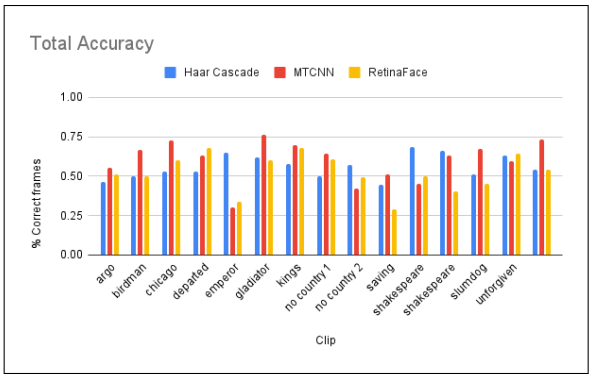
and angle of object, it is not very versatile to variation like multiple face angles. On the other hand, this model is not computationally complex and is relatively fast.

2.2. Multi-tasked Cascade Convolutional Network

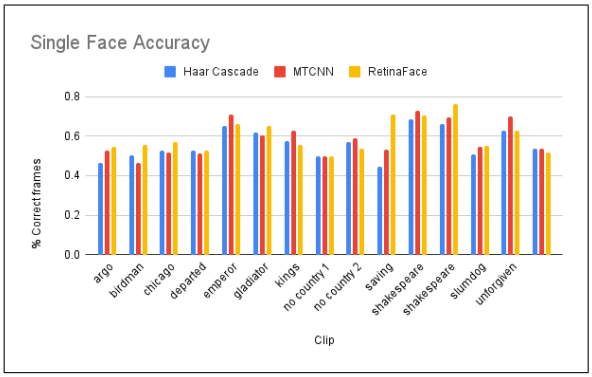
MTCNNs are designed specifically for face detection and localization, which means it is able to identify facial landmarks such as eyes and noses in addition to the overall face. It consists of three layers of networks; the first is a shallow fully convolutional network (FCN) that identifies potential areas of interest in the image. The second layer is a standard convolutional neural network (CNN) that specifies the facial bounding box and narrows down facial feature locations. Finally, it finishes with another CNN that identifies the exact location of facial features.

2.3. RetinaFace

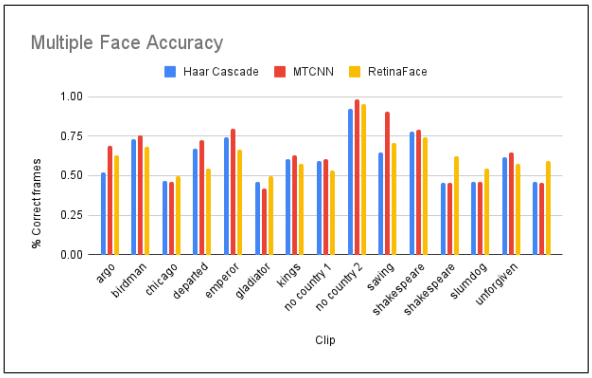
RetinaFace is a more recently developed network that performs facial detection, localization, and generates a 3D mesh of the face. It employs the most complex architecture of the three models, using a feature pyramid network (FPN) with three major stages. First, it goes through a residual network (ResNet) for preliminary feature detection. Then, it goes through a series of convolutional layers that are each connected to a layer in the preceding ResNet. This stage conducts localization, and the connection to higher resolution layers helps to perform on a larger scale that allows it to identify details like small faces in large crowds and facial landmarks. Lastly, it goes through a series of deformation convolutional networks (DCNs) that models geometric transformation; this is mostly for the 3D mesh component, which works by regressing a premade set of vertices to two dimensions and matching it to the face in the image. Because of its more complex structure, RetinaFace takes significantly longer to process images than either of the previous models but is better at identifying less detailed faces.



(a) % correctly labeled frames



(b) % frames correctly labeled with a single face



(c) % frames correctly labeled with multiple faces

Figure 1. Accuracy of each model model by clip

2.4. Code

The code for this project, including information on the specific model implementations that were used, can be accessed at https://github.com/avinella/automated-face-labeling.

Model	Avg. % correct frames	Avg. % correct single face	Avg. % correct mult. faces
Haar Cascade	60.1	56.0	60.9
MTCNN	60.0	58.6	65.2
RetinaFace	52.3	59.9	62.3

Table 1. Average accuracy for each model

3. Results

Despite the different architectures of each model and their expected outcomes, the three overall performed nearly the same in terms of accuracy (Fig. 1). The average number of accurate frames differs by less than 10%, and the average single and multiple face accuracies are even more similar (Tab. 1). No model consistently performs better than the others, although the Haar cascade classifier and MTCNN seem to overall do better than RetinaFace in frame-by-frame accuracy, and the Haar cascade classifier performs slightly worse in the single/multiple face detection. Note that the frame-by-frame accuracy also indicates how the single/multiple face errors interfere with each other; for example, if a frame with a single face is incorrectly predicted to have multiple faces, it will count as an error for both single and multiple face labels but as only one incorrect frame.

Qualitatively, the Haar cascade seems to frequent both false positives and false negatives. It was most successful in identifying faces that take up most of the frame and face mostly forwards, but often classified background elements as faces. The MTCNN generally fails with false negatives, mostly in smaller faces, and RetinaFace seemed to have little error overall. The RetinaFace output also seems to be more stable in that it will consistently track a single face across adjacent frames, while both the Haar cascade classifier and MTCNN often did not identify a face that it had already predicted in the previous frame. Because these annotated videos are inconsistent with the quantitative results, this may mean that there may be other factors that interfered with its evaluation, which will be discussed in the analysis.

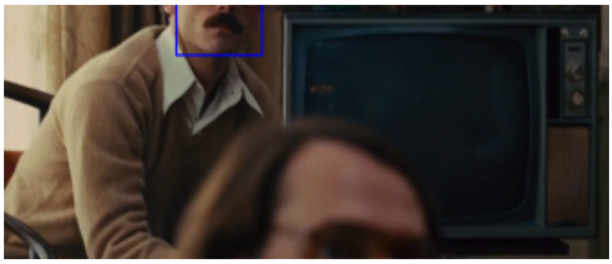


Figure 2. RetinaFace annotation of a partial face in frame from *Argo*

There are also a few edge cases in which some or all of the models failed to identify faces. First, the Haar cascade classifier struggled with angled faces that are not head-on. Even though there is a trained classifier specifically for profile views, it still generally does not detect profile faces. Second, partially obscured or cut-off faces were most often correctly identified by RetinaFace, but not the other two models (Fig. 2). Lastly, none of the models were able to identify close-up shots that only contained a few facial features and did not include the overall shape of the face.

Unlike the accuracy results, the efficiency of each model was as expected. The Haar cascade classifier is by far the fastest model, and the MTCNN performed slightly faster than RetinaFace (Fig. 3).

4. Analysis

Some of the results of this study are expected. For the most part, the three models all struggle to identify faces that were not fully contained within the frame. This makes sense, as they require the overall face of the shape or the ability to localize

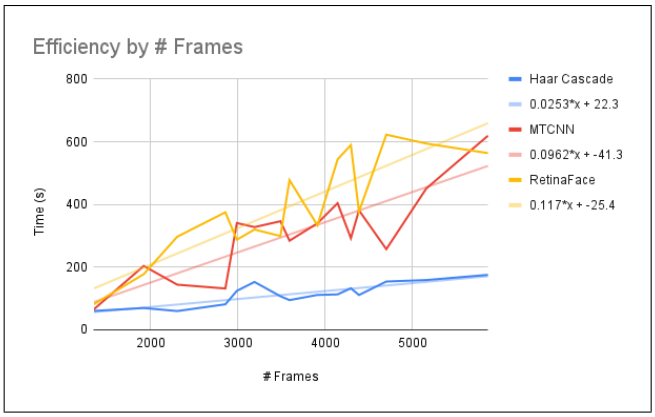


Figure 3. Runtime in seconds for each model processing frames

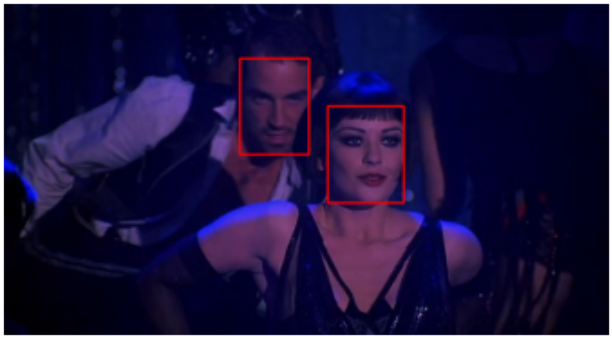


Figure 4. MTCNN annotation of two visible faces in frame that is labeled as single face in ground truth

facial landmarks in relationship to each other, and this type of problem will likely continue to prove difficult even with additional fine tunings of the models.

On the other hand, the similar results for the accuracy of each model is certainly surprising. The Haar cascade classifier is very brittle because it looks for and trains on very specific features, and it should have significantly underperformed compared to the other two. On the other extreme, RetinaFace is very good at predicting less visible or smaller faces, and thus should have performed better. However, by examining some examples of model annotated and hand-labeled images, it is clear that the ground truth data may be too ambiguous to the point that "good" models correctly detected faces that were not indicated in the ground truth.

There are two cases where this notably occurs. First, it is unclear how detailed or foregrounded a face must be to be counted. In many clips, background faces were not hand labeled, but RetinaFace in particular correctly identified them. However, this results in a mismatch when comparing to the ground truth, even though the model itself was not incorrect.

Another ambiguous case in the provided ground truth concerns instances where a face moves from being occluded to fully visible. These frames often do not have this "new" face labeled until it is nearly fully visible. For example, in Fig. 4, two faces are correctly identified by the MTCNN. In the context of the clip, the male figure is moving from behind the woman's head, where he is not visible, to the side where he is visible. However, this frame is still labeled as only having a single face in the ground truth; the frame following this one is labeled as having multiple. This leads to an incorrect model label, even though it identified the faces in frame. Similarly, camera pans that move from one face to another often have a few frames in which neither face is fully in frame, and both the Haar cascade and MTCNN will often not be able to detect either. However, the ground truth does not account for this transition and will generally label that section of the clip as containing a face. This distinction in the labeling of partial faces is not qualitatively determined and creates an inconsistency with the models' ability to recognize partially occluded faces, which again causes an often correct prediction to be counted as a mislabel. Thus, an overall mediocre accuracy of otherwise well-performing models may not be indicative of the model's face detection inaccuracy, but instead of ambiguity within the data set.

Nevertheless, there are certainly improvements and additions to the models that will improve their performance. As all of the tested models were pretrained and used off-the-shelf, they will likely

benefit from additional hyperparameter finetuning. Moreover, as both the Haar cascade classifier and the MTCNN struggled to label the same face from frame to frame, it can be improved by some method that compares adjacent frames, such as optical flow or keypoint matching. This should greatly improve the number of false negatives from both models.

5. Conclusion

Quantitatively, it seems that each model was similarly accurate in its face detection such that the higher efficiency of the Haar cascade classifier may be desirable despite the slight decrease in accuracy. On the other hand, the qualitative analysis of the output annotated videos indicates that the MTCNN has a better balance of accuracy and speed. If speed is not a concern, then RetinaFace qualitatively seems to be more accurate than the other two. However, because of inconsistencies between the qualitative and quantitative results, as well as within the ground truth labels, no strong conclusions can be made about which model overall performs best for labeling film clip data. It would be recommended to both customize each model for this data set and further specify the ground truth labels before moving forward with using their automated labels to supplement gaze data sets.

References

[1] Katherine Breeden and Pat Hanrahan. Gaze data for the analysis of attention in feature films. *ACM Trans. Appl. Percept.*, 14(4), sep 2017. 1

[2] Jiankang Deng, Jia Guo, Yuxiang Zhou, Jinke Yu, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-stage dense face localisation in the wild, 2019. 1

[3] 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*, volume 1, pages I–I, 2001. 1

[4] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, oct 2016. 1