SELFIE CAPTURE WITH SMILE DETECTION

Nurul Nadiah Binti Zulkifli, 1161101947
Puteri Aisyah Binti Megat Muzafar, 1171100470
Nur Sabrina Syazwani Binti Mazlan, 1171302745

Faculty of Computing and Informatics, Multimedia University, Cyberjaya

ABSTRACT **—** A smile represents satisfaction and happiness. Many applications are created using smile detection technology, for example product rating, patient monitoring, image capturing, video conferencing and interactive systems. In this project, we focused on how to apply smile detection for image capturing specifically for selfie capture and to compare which methods perform better at smile detection by measuring the accuracy, precision, recall, and F1-score. Methods used for this project are Haar Feature Selection and Mouth Aspect Ratio (MAR). Both methods were tested by using datasets of images, videos and real-time webcam. The comparison for both methods was evaluated based on the evaluation metrics. The results showed that MAR is generally better at detecting a smile on non-static images compared to Haar. However, both methods did not perform well at detecting smiles for static images. Nonetheless, they showed good results on capturing selfies using the real-time webcam. Some improvements could be made to improve the performance for both methods, such as the features used and haarcascades values for Haar, as well as the threshold values set for MAR.

Keywords—Smile Detection; OpenCV; Haar Feature Selection; Mouth Aspect Ratio

1. INTRODUCTION

In this modern day and age, smartphones play a crucial role in taking pictures. Less and less people have a tendency to bring around their cameras as smartphones are generally a “one size fits all” in terms of having a lot of similar uses to those of a camera. For example, pictures taken from a smartphone could sometimes beat those of a professional camera, plus they are much cheaper too than some. In addition, smartphones are much more used due to the presence of a front and back camera that enables users to take pictures from both angles. The front camera in a smartphone is especially used for “selfies” or also known as self-portraits. It could also capture a photo full of people for a family photo, friendship photo and others.

Selfies are taken while one extends their arm at a certain position to get the best view of one-self. Then, it is continued by clicking the button on the phone to take the picture. Therefore, this sometimes limits the outcome of the picture. It is very difficult for people to hold the phone while simultaneously taking the picture. The image may get blurry due to the movement of the hand during the clicking of the button. Furthermore, sometimes the images taken do not capture the people at their best. One may be smiling while another does not. Henceforth, the computer vision that will be investigated is smile detection. With smile detection, the camera will only capture the image once everyone detected inside the image is smiling. Consequently, the perfect photo with no blurry areas and with every person smiling will be taken.

2. RELATED WORK

The study from Whitehill et al. [1] presented a practical smile detection embedded in digital cameras. They collected 63,000 real-life faces from the Internet and stored the images in a database called GENKI. They investigated different parameters including size and type of datasets, image registration using five facial features such as Gabor Energy Filters (GEF), Box Filters (BF), Edge Orientation Histograms (EOH), combination of BF and EOH, and Local Binary Patterns (LBP), facial representation, and machine learning algorithms such as Support Vector Machine (SVM) and GentleBoost. The area under the ROC curve shows that 97.9% GentleBoost classifiers trained on binary tasks are highly correlated with human estimates of smile intensity, both in still images and video. The combined feature set BF and EOH presented the best recognition performance overall training set sizes. Their study suggests high detection accuracy is achievable in real-life situations with machine learning methods.

Another related work from Huang and Fuh (2009) studied for both face detection and smile detection [2]. From their studies, they proposed a real-time smile detection system. To extract key features of the human face, the dataset from BIOID that contains a face database of 1521 grayscale images with a 384x286 resolution was analysed using AdaBoost. Patterns of object motions are also studied through various equations to obtain the displacement vector of the features. Therefore, based on those two subjects, the smile detection is executed by (1) Detecting and finding the 20 standard facial features, (2) Track position of corners of the mouth, (3) Claiming a smile if the distance of the mouth corners is larger than a specified threshold. Furthermore, they used a Normalized Cross Correlation (NCC) block to obtain the best match to a mouth region. From the proposed method, it was then experimented on a facial expression database from FGNET, comparing to a camera called Sony T300 which has a smile shutter function that takes a photo when a smile is detected. Overall, results show that their system was better than Sony T300's smile shutter function in terms of robustness, high accuracy and able to detect slight smiles.

Winal Zikril Zulkifli et. al. [3] did other studies related to smile detection in 2018. The research used a smile detection tool using OpenCV-Python to measure response in Human-Robot Interaction (HRI) with animal robot PARO for medical therapy. This study constructed a tool using OpenCV-Python to detect the number of smiles when each patient interacts with PARO. First, a collective data on human smiles was included for smile identification. This action is followed by identifying appropriate dataset processing libraries. Once the dataset library had been identified, the development continued with detection and learning. For smile detection algorithm, Adaptive Boosting (AdaBoost) was used. Using AdaBoost improves the accuracy of the learning algorithm where the output of multiple "weak classifiers" are combined into a weighted sum that represents the final output of the boosted classifier. To test for accuracy, three dataset of images which are (1) 300 images from 10 Malaysian subjects by using 5-fold cross-validation, (2) GENKI database, developed by MPLab and (3) Japanese female facial expression (JAFEE) were fed into the smile detection program. The results on program accuracy are 90%, 76% and 70% respectively.

3. APPROACH

OpenCV is an open source computer vision library for commercial and research use. It is one of the most widely used libraries in image processing. The OpenCV was chosen for its extensive library, simple usage and extensive user network. Our study is to compare two existing algorithms which Haar Feature Selection [4] and Mouth Aspect Ratio [5].

3.1. Mouth Aspect Ratio (MAR) Technique

3.1.1. Library Required

- numpy: Used for fast matrix calculations and manipulations.
- dlib: Library containing the facial landmarks.
- cv2: The OpenCV library used for image manipulation and saving.
- scipy.spatial: Used to calculate the Euclidean distance between facial points.
- imutils: Library to access video streams.

3.1.2. Facial Landmark Detector in dlib

This study will be using the facial landmarks detector from dlib to get the mouth coordinates. The facial landmark detector is an API implemented inside dlib. It produces 68 x- y-coordinates that map to specific facials including eyes, jaws, nose and mouth structures. We will focus on the mouth which can be accessed through point range [49,…, 68]. There are twenty coordinates for mouth structure.

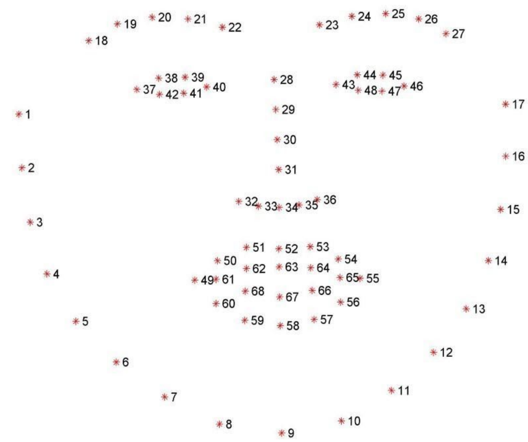


Figure 1: Facial landmarks index template

3.1.3. MAR Computation

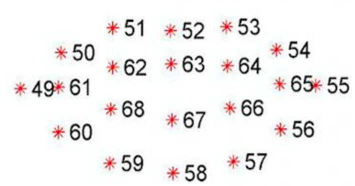


Figure 2: Mouth structure coordinates cropped from Figure 1

In this case, MAR is simply defined as the relationship of the points shown below:

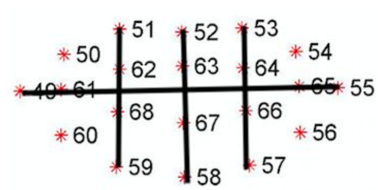


Figure 3: The mouth part extracted from Figure 2

This technique proposes to compute the distance between point 49 and point 55 as D, and calculate the average of the distances between:

- point 51 and point 59
- point 52 and point 58
- point 53 and point 57

Let's call it L, using the same naming structure:

$$MAR = \frac{\| point\ 51 - point\ 59 \| + \| point\ 52 - point\ 58 \| + \| point\ 53 - point\ 57 \|}{3 * \| point\ 49 - point\ 55 \|}$$

Hence, MAR = L/D

The ratio for the smile detector will be influenced by how an individual will be smiling. Smiling with the mouth closed will increase the distance between point 49 and point 55 and decrease the distance between the top and bottom points. So, L will decrease. Smiling with mouth open will lead to D decreasing and L increasing.

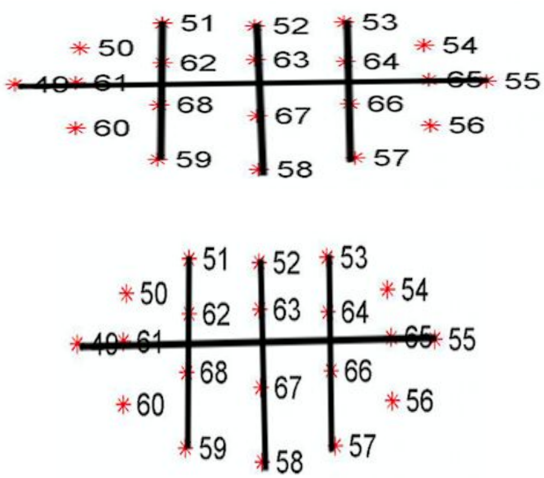


Figure 4: Smiling with the mouth closed (up) vs smiling with the mouth opened (bottom)

Based on this theory, this study will set the threshold values for smiling with mouth closed as value1 and smiling with mouth opened as value2. The detection of a person smiling will be detected when MAR is less than value1 (smile wider) or greater than value2 (smile bigger with teeth).

Table 1: Threshold values set for mouth aspect ratio

< value2	*Value2* <= value <= *value1*	*> value1*
Smile wider	Neutral face	Smile bigger with teeth

3.2. Haar Feature Selection Technique

3.2.1. Library Required

- numpy: Used for fast matrix calculations and manipulations.
- cv2: The OpenCV library used for image manipulation and saving.

3.2.2. Face Detection using Haar Feature Selection

The dataset of images will be including haar cascade files in the python file. The video that will be captured from the real-time camera is nothing but a series of images. Therefore, they will run as an infinite while-loop of images. The algorithm will read faces using an already included haarcascade file and detectMultiscale() function where the gray image, ScaleFactor, and minNeighbors will be passed.

- ScaleFactor: Parameter specifying zoom image
- minNeighbors: Parameter specifying how many neighbors each rectangle should have to retain it.

Object Detection using Haar feature-based cascade classifiers is an effective object detection method proposed by Paul Viola and Michael Jones. It is a machine learning based approach where a cascade function is trained from a lot of positive and negative images. It is then used to detect objects in other images. Here, it will work with face detection. Initially, the algorithm needs a lot of positive images (images of faces) and negative images (images without faces) to train the classifier. Then, we need to extract features from it. For this, Haar features shown in the below image are used.

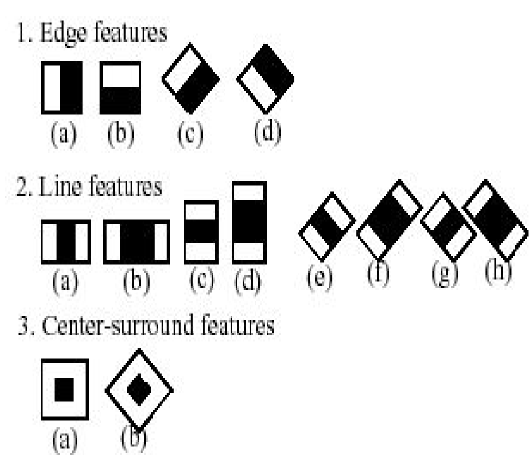


Figure 5: Haar-like features

If it detects a face, an outer boundary of the face will be drawn using rectangle() method of cv2. The method contains 5 arguments:

- image: Image on which rectangle is to be drawn.
- start_point: Starting coordinates of rectangle. The coordinates are represented as tuples of two values i.e. (X coordinate value, Y coordinate value).
- end_point: Ending coordinates of rectangle. The coordinates are represented as tuples of two values i.e. (X coordinate value, Y coordinate value).
- color: Color of border line of rectangle to be drawn.
- thickness: Thickness of the rectangle border line in px.

After the face is detected, the structure for the mouth hence a smile will be detected by using the smile haar cascade file that can be imported easily in the algorithm to detect smile in the dataset.

3.3. Auto Capture Image during Video Live Stream

The study will be using the VideoCapture() from OpenCV according to MAR values that will be set up and passed values to parameters of Haar detection function. Once the values meet the range of MAR and Haar detection function, the image will be auto captured and saved to the directory. Finally, the windows of the video streaming will be destroyed.

4. EXPERIMENT

For this project, we are comparing two existing algorithms which are by using Haar Feature Selection and Mouth Aspect Ratio (MAR) to determine which method will be implemented in the auto capture image during video live stream. We conducted a test for both approaches to analyse which methods perform better at smile detection by measuring the accuracy, precision, recall, and F1-score. The performance evaluation using confusion matrix is made in terms of achieving the best trade-off between correct detection and false detection during the testing of the dataset. After the testing is complete, live detection of smiles during a selfie will be implemented using a webcam in real-time. This is to test whether the algorithm works for both static and non-static images.

4.1. Dataset

To test the smile detector, the dataset that will be used is an existing dataset that consists of both smiling and not smiling images and video from all sets of gender including male and female. The purpose of collecting from different genders is to test the smile detector with contrasting types of smiles. Some smiles could be restrained but still considered as a smile according to the angle of the mouth, show of dental or the extent of their smiles. One of the datasets would be “SMILES” by Daniel Hromada in Github [6] that contains smiling and non-smiling grayscale images, focused on the crop of the face. The second dataset would be from the YouTube video, “Make People Smile Project," [7] that also contains people with smiling and non-smiling faces.

4.2. Haar Feature Selection

Haar Feature Selection is the first technique we used to test the images, video and live capture. For images, we used a smile haarcascade to only detect smiles as the images are zoomed into the face. For video, eye and smile haar cascade were used to detect a smile because it had better detection than a combination of face and smile haarcascade. Lastly, for the live capture, we used a face and smile haar cascade to detect a smile before the capture. The values for the haarcascades are as below:

Smile Haarcascade:
scaleFactor = 1.5,
minNeighbors = 15
minSize = (25, 25)

Face Haarcascade:
scaleFactor = 1.3,
minNeighbors = 5
minSize = (30, 30)

4.2.1. Image

Figure 6: Example of Positive Images

Figure 7: Example of Negative Images

Haar Feature Selection is tested for images by detecting the smile area. The dataset used to test the method consists of positive images that show people smiling (Figure 6) and negative images which illustrates people who are not smiling (Figure 7). If the system detects a smile, the image is saved by their image number. Both datasets are combined based on their image number. The results are then recorded manually in a csv file for actual smile and predicted smile. ActualSmile is 1 if ImgNo belongs to Positive Images dataset while 0 if the image is from Negative Images. If Haar detects a smile from the images, the value will be 1, otherwise 0 for a non-smile in PredictedSmile. An example is shown below in Table 2.

Table 2: Haar Image Actual and Predicted Smile

ImgNo	ActualSmile	PredictedSmile
10009	1	0
10011	1	0
10012	1	0
10013	1	0
10018	1	1

*1 = smile, 0 = non-smile

The total number of images tested was 13165. 3690 of them are Positive Images dataset while 9475 are Negative Images dataset. Out of 3690 smiling images, 104 are predicted as smiles and 3586 are predicted as non-smiles. On the other hand, out of 9475 non-smiling images, 3586 are predicted as smiles and 9459 as non-smiles.

```
Predicted     0     1
Actual
0          9459    16
1          3586   104
```

Figure 8: Confusion Matrix of Haar Image Testing

Figure 8 illustrates the confusion matrix for Haar image testing. The results are listed below:

- True Negative (TN) = 9459
- False Negative (FN) = 3586
- False Positive (FP) = 16
- True Positive (TP) = 104

The results from the confusion matrix is further visualised in a form of heatmap as shown in Figure 9.

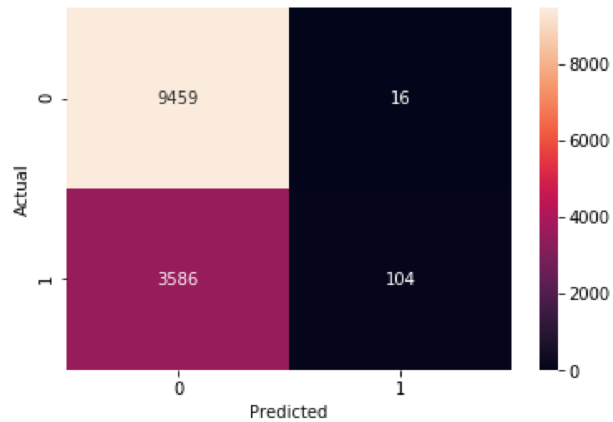


Figure 9: Heatmap for Haar Image Testing Confusion Matrix

Finally, evaluation metrics namely accuracy, precision, recall and F1-score are calculated. The formula to calculate all four evaluation metrics are shown below:

Total Images = 13165

- Accuracy = (TP + TN) / Total Images
- Precision = TP / (TP + FP)
- Recall = TP / (TP + FN)
- F1-score = 2 * (Precision * Recall) / (Precision + Recall)

Results are shown in Figure 10. The accuracy, precision, recall and F1-score for Haar image testing are 0.73, 0.87, 0.03 and 0.05 respectively.

```
Accuracy HAAR (Image): 0.7263957462969997
Precision HAAR (Image): 0.8666666666666667
Recall HAAR (Image): 0.028184281842818428
F1-Score HAAR (Image): 0.05459317585301837
```

Figure 10: Evaluation Metrics for Haar Image Testing

4.2.2. Video

Haar Feature Selection is tested for videos by detecting the eye and smile area. As the video runs, if a smile is detected, "Smile Detected" will be displayed in green to tell users that a smile has been detected. Next, "Captured" is displayed on the video to indicate that the image has been saved. Images are saved based on the video frame number.

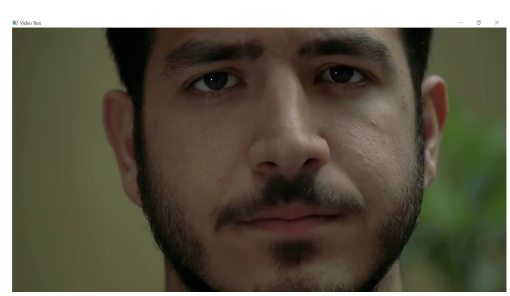


Figure 11: Haar Video Testing

The results are then recorded manually in a csv file for actual smile and predicted smile. ActualSmile is 1 if the person is smiling in the respective FrameNo while 0 if the person is not smiling. If Haar detects a smile from the video, the value will be 1, otherwise 0 for a non-smile in PredictedSmile. An example is shown below in Table 3.

Table 3: Haar Video Actual and Predicted Smile

FrameNo	ActualSmile	PredictedSmile
66	0	1
67	0	0
68	0	1
69	0	1
70	0	1

*1 = smile, 0 = non-smile

The total number of frames occurred was 3291, 1448 frames are Positive while 1843 are Negative. Out of 1448 frames of smiling images, 1273 are predicted as smiles and 175 are predicted as non-smiles. On the other hand, out of 1843 frames of non-smiling images, 1169 are predicted as smiles and 674 as non-smiles.

```
Predicted    0     1
Actual
0          674  1169
1          175  1273
```

Figure 12: Confusion Matrix of Haar Video Testing

Figure 12 illustrates the confusion matrix for Haar video testing. The results are listed below:

- True Negative (TN) = 674
- False Negative (FN) = 175
- False Positive (FP) = 1169
- True Positive (TP) = 1273

The results from the confusion matrix is further visualised in a form of heatmap as shown in Figure 13.

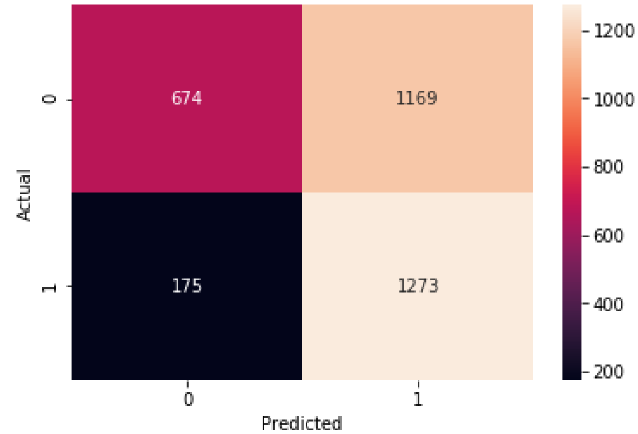


Figure 13: Heatmap for Haar Video Testing Confusion Matrix

Finally, evaluation metrics namely accuracy, precision, recall and F1-score are calculated. The formula to calculate all four evaluation metrics are shown below:

Total Frames = 3291

- Accuracy = (TP + TN) / Total Frames
- Precision = TP / (TP + FP)
- Recall = TP / (TP + FN)
- F1-score = 2 * (Precision * Recall) / (Precision + Recall)

Results are shown in Figure 14. The accuracy, precision, recall and F1-score for Haar video testing are 0.59, 0.52, 0.88 and 0.65 respectively.

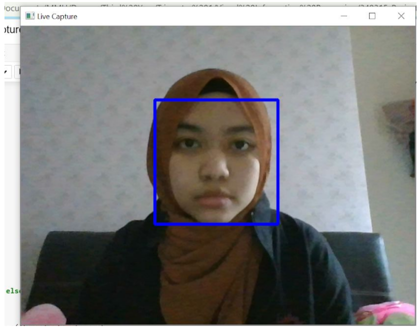
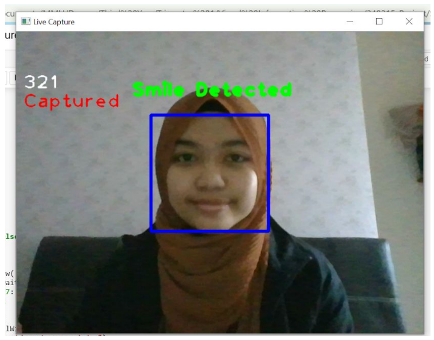
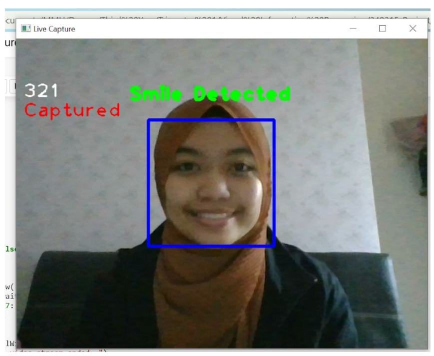
```
Accuracy HAAR (Video): 0.5916134913400183
Precision HAAR (Video): 0.5212940212940212
Recall HAAR (Video): 0.8791436464088398
F1-Score HAAR (Video): 0.6544987146529563
```

Figure 14: Evaluation Metrics for Haar Image Testing

4.2.3. Webcam

Haar Feature Selection is tested for selfie capture through a webcam. As it is a realtime test, the accuracy can only be indicated by how well Haar can detect a smile for three situations. The first situation is if the person is not smiling or showing a neutral face. The second situation is if the person is smiling. The third situation is a wide smile. If a smile is detected, the system will show “Smile Detected” while showing the area of the face. A countdown will begin and display “3 2 1” before capturing a selfie. When the selfie is captured, the system will display “Captured” and the webcam is stopped for 2 seconds. The image is saved. Below are examples of capturing selfies with smile detection using Haar.

Table 4: Tests for Selfie Capture with Smile Detection using Haar

Test	Webcam
No Smile / Neutral Face	
Smiling	
Wide Smile	

4.3. Mouth Aspect Ratio (MAR)

Mouth Aspect Ratio (MAR) is the second technique we used for testing the images by detecting the smile area. Based on its theory as explained earlier in Section 3.3, this study will set the threshold values for smiling with mouth closed as 0.25 and smiling with mouth opened as 0.38. The detection of a person smiling will be detected when MAR is less than 0.25 (smile wider) or greater than 0.38 (smile bigger with teeth).

Table 5: Threshold values set for mouth aspect ratio

MAR < 0.25	*0.25 <= MAR <= 0.38*	MAR > 0.38
Smile wider (detect smile and capture image)	Neutral face	Smile bigger with teeth (detect smile and capture image)

4.3.1. Image

The same dataset as Haar Feature Selection (Section 4.2.1) consisting of positive images that show people smiling (Figure 6) and negative images which illustrate people who are not smiling (Figure 7) is used to test the method. If the system detects a smile, the image is saved by their image number. Both datasets are combined based on their image number. The results are then recorded manually in a csv file for actual smile and predicted smile. ActualSmile is 1 if ImgNo belongs to the Positive Images dataset while 0 if the image is from Negative Images dataset. If MAR is either less than 0.25 or greater than 0.38, it detects a smile from the images, hence, the value will be 1, otherwise 0 for a non-smile in PredictedSmile. An example is shown below in Table 6.

Table 6: MAR Image Actual and Predicted Smile

ImgNo	ActualSmile	PredictedSmile
10030	1	0
10033	1	1
10035	1	0
10045	1	0
10046	1	1

*1 = smile, 0 = non-smile

The total number of images tested was 13165, 3690 are Positive Images dataset while 9475 are Negative Images dataset. Out of 3690 smiling images, 1154 are predicted as smiles and 2536 are predicted as non-smiles. On the other hand, out of 9475 non-smiling images, 4442 are predicted as smiles and 5033 as non-smiles.

```
Predicted     0     1
Actual
0          5033  4442
1          2536  1154
```

Figure 15: Confusion Matrix of MAR Image Testing

Figure 15 illustrates the confusion matrix for MAR image testing. The results are listed below:

- True Negative (TN) = 5033
- False Negative (FN) = 2536

- False Positive (FP) = 4442
- True Positive (TP) = 1154

The results from the confusion matrix is further visualised in a form of heatmap as shown in Figure 16.

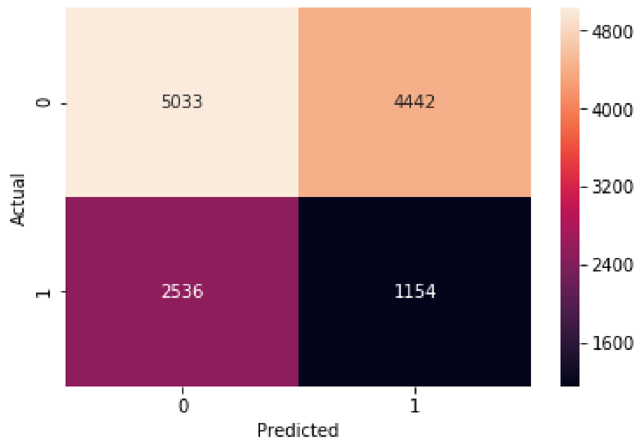


Figure 16: Heatmap for MAR Image Testing Confusion Matrix

Finally, evaluation metrics namely accuracy, precision, recall and F1-score are calculated. The formula to calculate all four evaluation metrics are shown below:

Total Images = 13165

- Accuracy = (TP + TN) / Total Images
- Precision = TP / (TP + FP)
- Recall = TP / (TP + FN)
- F1-score = 2 * (Precision * Recall) / (Precision + Recall)

Results are shown in Figure 17. The accuracy, precision, recall and F1-score for MAR image testing are 0.47, 0.21, 0.31 and 0.25 respectively.

```
Accuracy MAR : 0.4699582225598177
Precision MAR : 0.20621872766261615
Recall MAR : 0.31273712737127374
F1-Score MAR : 0.24854619857850527
```

Figure 17: Evaluation Metrics for MAR Image Testing

4.3.2. Video

Mouth Aspect Ratio (MAR) is also tested for videos. As the video runs, MAR value is displayed on the top-left window to indicate the mouth aspect ratio by each of the people in the video. If a smile is detected, "Smile Detected" will be displayed in green to tell users that a smile has been detected. Next, "Captured" is displayed on the video to indicate that the image has been saved. Images are saved based on the video frame number. Figure 18 shows the example of an image captured during video testing.

Figure 18: MAR Video Testing

The results are then recorded manually in a csv file for actual smile and predicted smile. ActualSmile is 1 if the person is smiling in the respective FrameNo while 0 if the person is not smiling. If MAR is either less than 0.25 or greater than 0.38, it detects a smile from the images, hence, the value will be 1, otherwise 0 for a non-smile in PredictedSmile. An example is shown below in Table 7.

Table 7: MAR Video Actual and Predicted Smile

FrameNo	ActualSmile	PredictedSmile
1354	1	1
1355	1	1
1356	1	1
1357	1	1
1358	1	1

*1 = smile, 0 = non-smile

The total number of frames occurred was 3291, 1448 frames are Positive while 1843 frames are Negative. Out of 1448 frames of smiling images, 883 are predicted as smiles and 565 are predicted as non-smiles. On the other hand, out of 1843 frames of non-smiling images, 246 are predicted as smiles and 1597 as non-smiles.

```
Predicted      0     1
Actual
0           1597   246
1            565   883
```

Figure 19: Confusion Matrix of MAR Video Testing

Figure 19 illustrates the confusion matrix for MAR video testing. The results are listed below:

- True Negative (TN) = 1597
- False Negative (FN) = 565
- False Positive (FP) = 246
- True Positive (TP) = 883

The results from the confusion matrix is further visualised in a form of heatmap as shown in Figure 20.

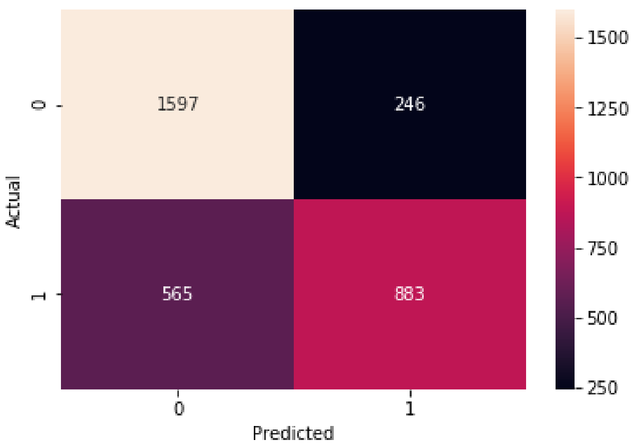


Figure 20: Heatmap for MAR Video Testing Confusion Matrix

Finally, evaluation metrics namely accuracy, precision, recall and F1-score are calculated. The formula to calculate all four evaluation metrics are shown below.

Total Frames = 3291

- Accuracy = (TP + TN) / Total Frames
- Precision = TP / (TP + FP)
- Recall = TP / (TP + FN)
- F1-score = 2 * (Precision * Recall) / (Precision + Recall)

Results are shown in Figure 21. The accuracy, precision, recall and F1-score for MAR video testing are 0.75, 0.78, 0.61 and 0.69 respectively.

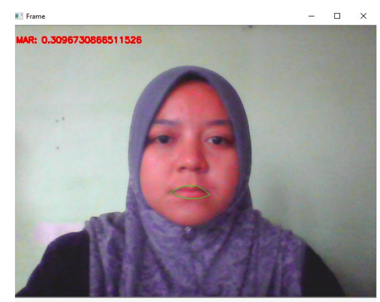
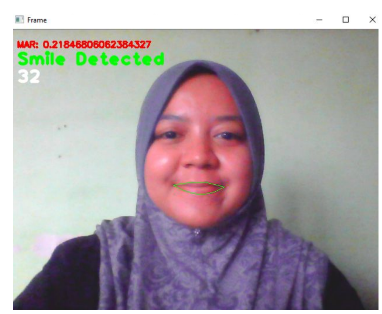
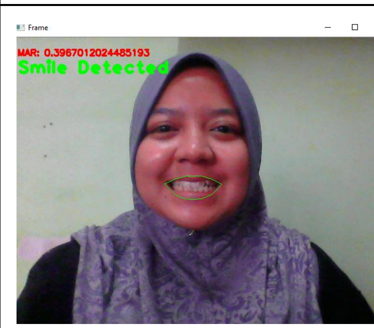
```
Accuracy MAR (Video): 0.7535703433606806
Precision MAR (Video): 0.7821080602302923
Recall MAR (Video): 0.6098066298342542
F1-Score MAR (Video): 0.6852929763290648
```

Figure 21: Evaluation Metrics for MAR Video Testing

4.3.3. Webcam

Mouth Aspect Ratio (MAR) is also tested for selfie capture through a webcam. As it is a realtime test, the accuracy can only be indicated by how well MAR can detect a smile for three different situations. The first situation is if the person is not smiling or showing a neutral face which MAR is set to value in between 0.25 and 0.38. The second situation is if the person is smiling with MAR value less than 0.25, and the third situation is if the person is smiling with MAR value greater than 0.38. If a smile is detected, the system will show “Smile Detected” while showing the area of the face. A countdown will begin and display “3 2 1” before capturing a selfie. When the selfie is captured, the system will display “Captured” and the webcam is stopped for 2 seconds. The image is saved. Table 8 below are examples of capturing selfies with smile detection using MAR.

Table 8: Tests for Selfie Capture with Smile Detection using MAR

<table>
<tr><th>Test</th><th>Webcam</th></tr>
<tr><td>No Smile / Neutral Face (0.25 <= MAR <= 0.38)</td><td>
</td></tr>
<tr><td>Smiling (MAR: < 0.25)</td><td>
</td></tr>
<tr><td>Smiling (MAR: > 0.38)</td><td>
</td></tr>
</table>

4.4. Comparison of Haar and MAR

4.4.1 Haar VS MAR (Image Testing)

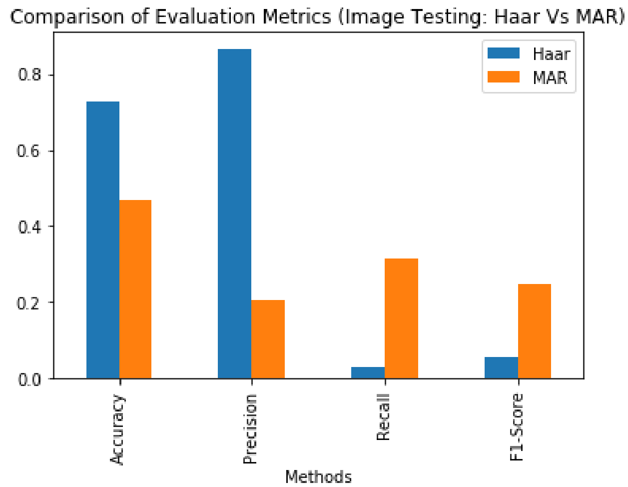


Figure 22: Comparison of Evaluation Metrics between Haar and MAR (Image)

Figure 22 shows the comparison of evaluation metrics between Haar and MAR for image testing. As seen in the graph, Haar has higher accuracy and precision than MAR which is very good as they are more than 0.7. However, recall and F1-score for Haar is very low compared to MAR.

Accuracy is a great measure but only when you have symmetric datasets where values of false positives and false negatives are almost the same. Otherwise, we have to look at other parameters to evaluate the performance of these algorithms.

High precision relates to the low false positive rate. We have got more than 0.8 precision for Haar which is pretty good.

F1-score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. F1 is usually more useful than accuracy because of the uneven class distribution of the datasets.

To conclude, although Haar has a higher accuracy than MAR, F1-score for both techniques are very low, which are less than 0.5. This shows that Haar Feature Selection and Mouth Aspect Ratio did not perform very well for static images. Some improvements could be made to improve the performance for both methods, such as the features used and haarcascades values for Haar, as well as the threshold values set for MAR.

4.4.2 Haar VS MAR (Video Testing)

Figure 23 shows the comparison of evaluation metrics between Haar and MAR for video testing. As seen in the graph, MAR beats Haar for accuracy, precision and F1-score but lower recall compared to Haar.

Values for false positives and false negatives are quite different for both techniques used, hence we still have to look at other parameters to evaluate the performance of these algorithms.

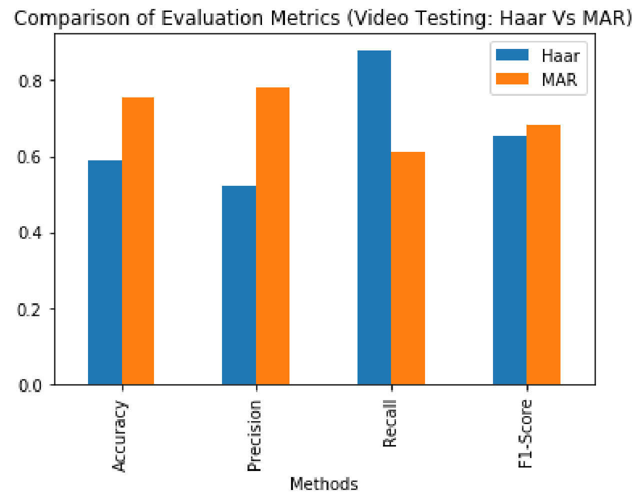


Figure 23: Comparison of Evaluation Metrics between Haar and MAR (Video)

As high precision relates to the low false positive rate, precision for Haar is more than 0.5 which is pretty good, while MAR predicted at its best with value near to 0.8.

Due to the uneven class distribution of the datasets, F1-score is also the important metric to be considered to evaluate their performances. F1-score for both techniques are good which are more than 0.6.

To conclude, MAR performed better on video testing. However, all evaluation metrics for Haar and MAR are considered high although one is higher than the other for certain values. This shows that Haar Feature Selection and Mouth Aspect Ratio perform well for videos or non-static images.

5. CONCLUSION

As a conclusion, this project was to capture a selfie with smile detection. Two methods were used which are Haar Feature Selection and Mouth Aspect Ratio (MAR). With good feature selections and threshold values for Haar and MAR respectively, both methods showed good results on capturing selfie using the real-time webcam. Meanwhile, these two methods were also compared to identify which method is more superior. From the tests conducted, MAR is generally better at detecting a smile. This is because it focuses on detecting the mouth figure on a person's face, while Haar focuses on finding features using the right haarcascade values. However, Haar also proved to be a good method to detect smiles depending on the features selected, and MAR could be worse if the threshold value is not set accurately. Furthermore, from the tests conducted, Haar and MAR works better for non-static images when detecting a smile. In future works, Haar and MAR could be combined to get a higher performance in smile detection.

6. REFERENCES

[1] J. Whitehill, G. Littlewort, I. Fasel, M. Bartlett and J. Movellan, "Toward Practical Smile Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 11, p. 2106, November 2009.

[2] C.-S. F. Yu-Hao Huang, "FACE DETECTION AND SMILE DETECTION," 2020.

[3] Winal Zikril Zulkifli, Syamimi Shamsuddin, Fairul Azni Jafar, Rabiah Ahmad, Azizah Abdul Manaf, Alaa Abdulsalam Alarood, Lim Thiam Hwee, "Smile Detection Tool using OpenCV-Python to Measure Response in Human-Robot Interaction with Animal Robot PARO," (IJACSA) International Journal of Advanced Computer Science and Applications, vol. 9, no. 11, 2018.

[4] DataFlair Team, "Python Project – Auto-capture Selfie by Detecting Smile," 18 August 2020. [Online]. Available: https://data-flair.training/blogs/python-project-capture-selfie-by-detecting-smile/.

[5] R. Agarwal, "Smilefie: how you can auto-capture selfies by detecting a smile," 7 August 2018. [Online]. Available: https://www.freecodecamp.org/news/smilfie-auto-capture-selfies-by-detecting-a-smile-using-opencv-and-python-8c5cfb6ec197/.

[6] Hromada, D, "SMILEsmileD," GitHub, 28 October 2010. [Online]. Available: https://github.com/hromi/SMILEsmileD.

[7] Huzeyfe Kurt, "Make People Smile Project," 15 May 2016. [Online]. Available: https://www.youtube.com/watch?v=8YuBxP4CKZc.