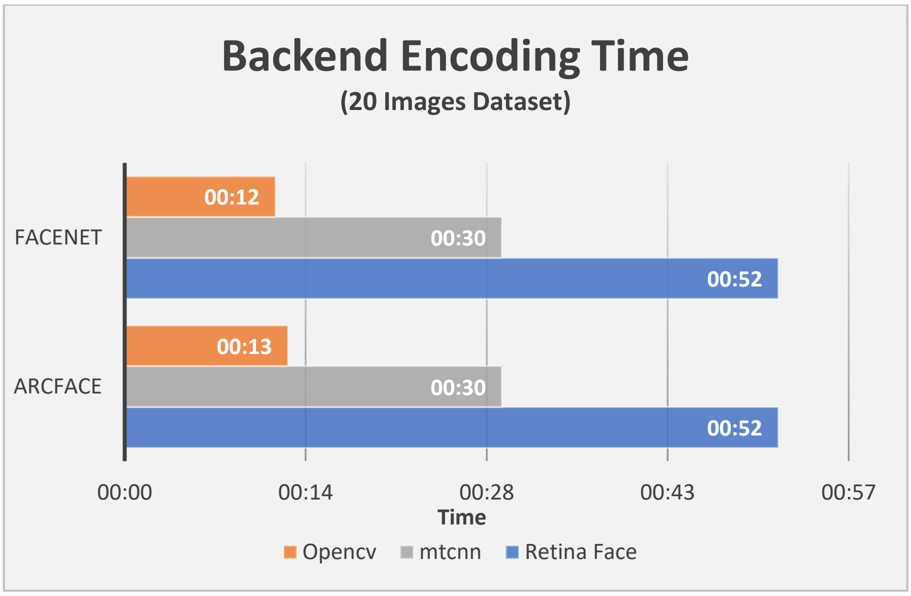
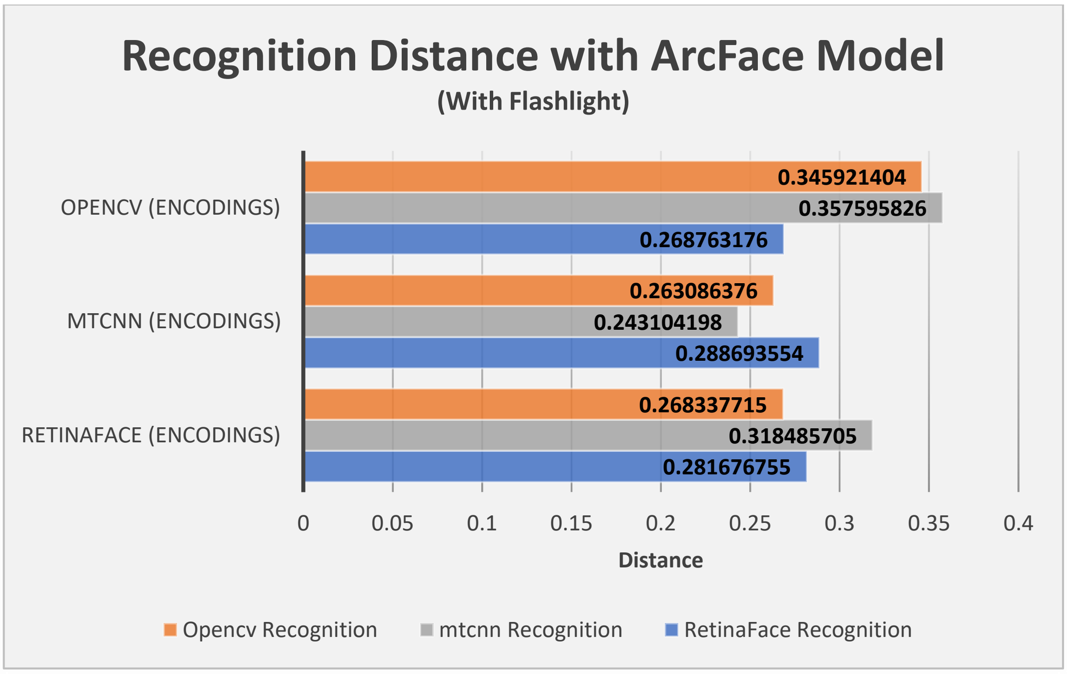
CHAPTER VI. RESULTS AND DISCUSSION

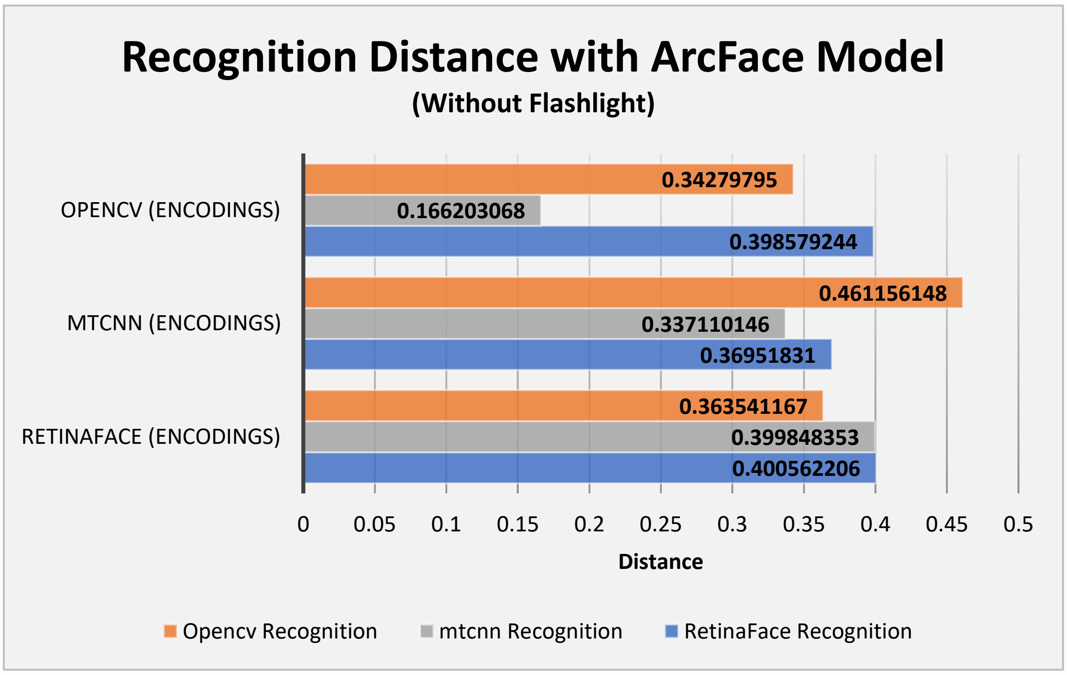
6.1 Results

Based on the results of the tests carried out in chapter V, the test begins with real-time face recognition using a photo dataset of 20 and where each person has more than one photo, the first experiment resulted in varying encoding or representation times and good accuracy.

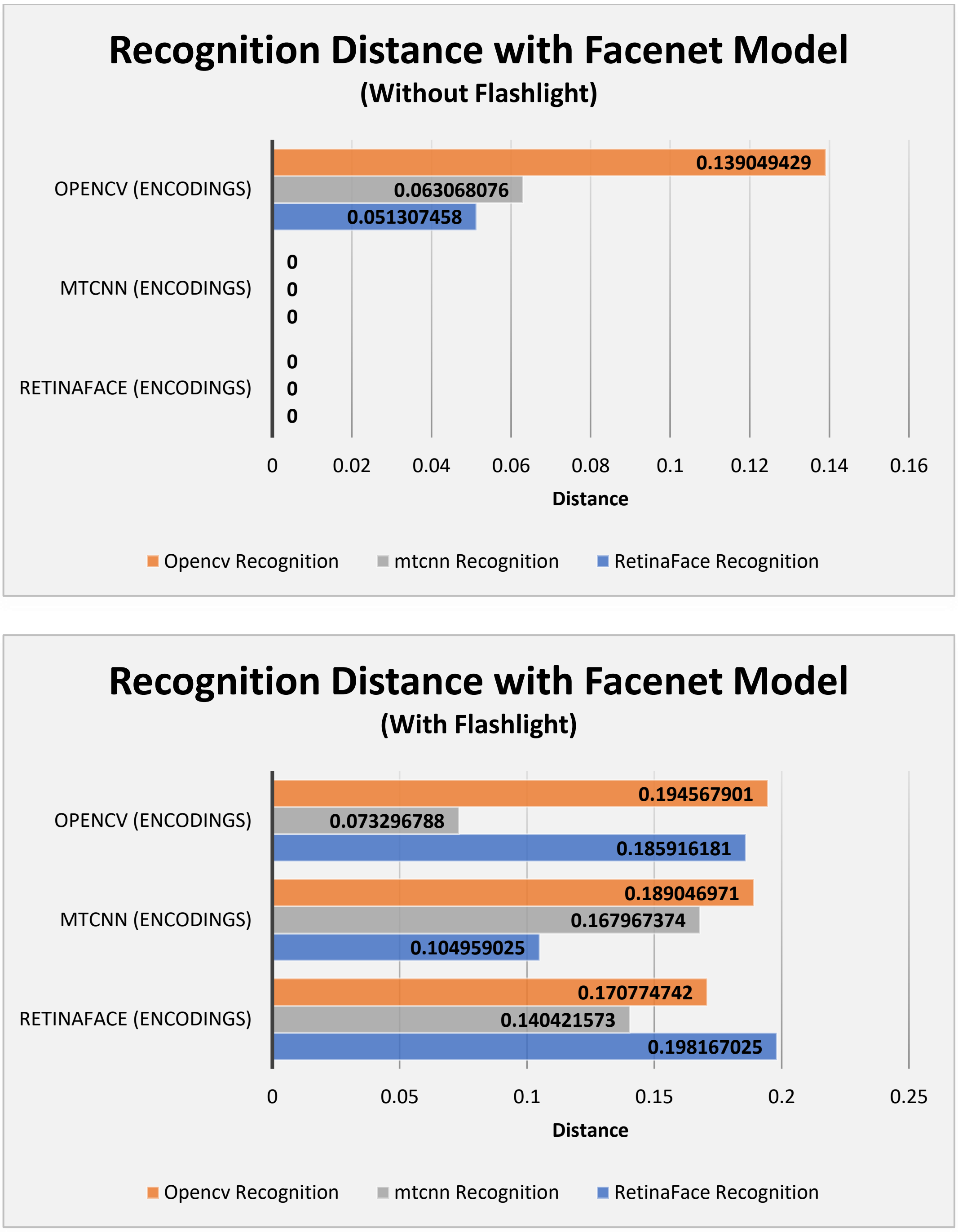


Figures 6.1 Encoding Time 20 Image Dataset

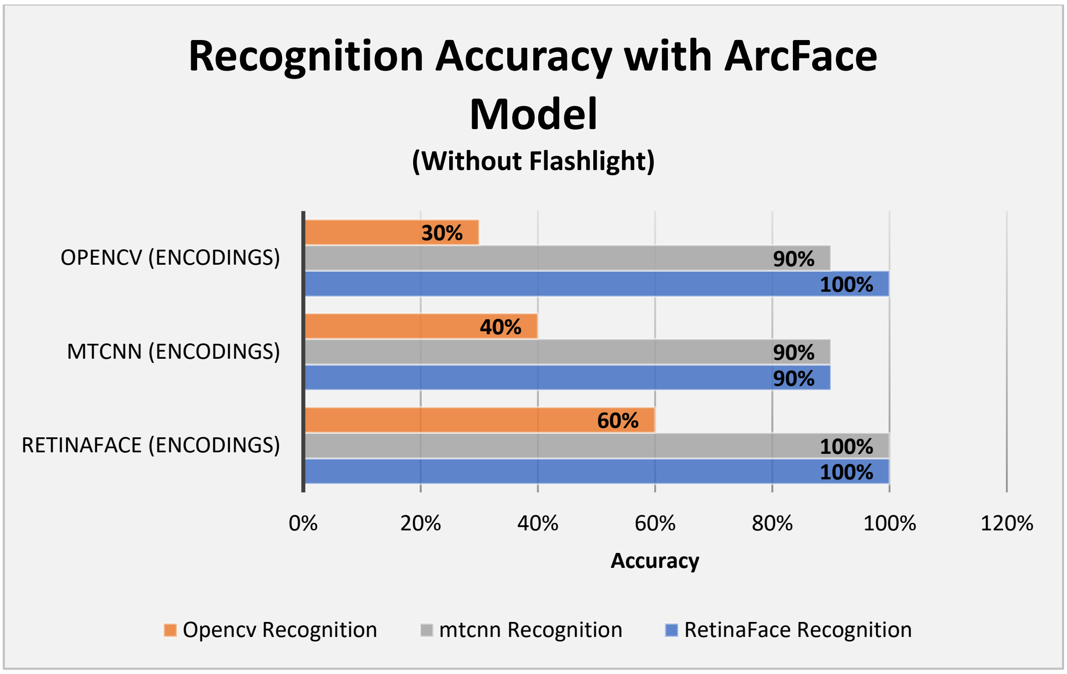


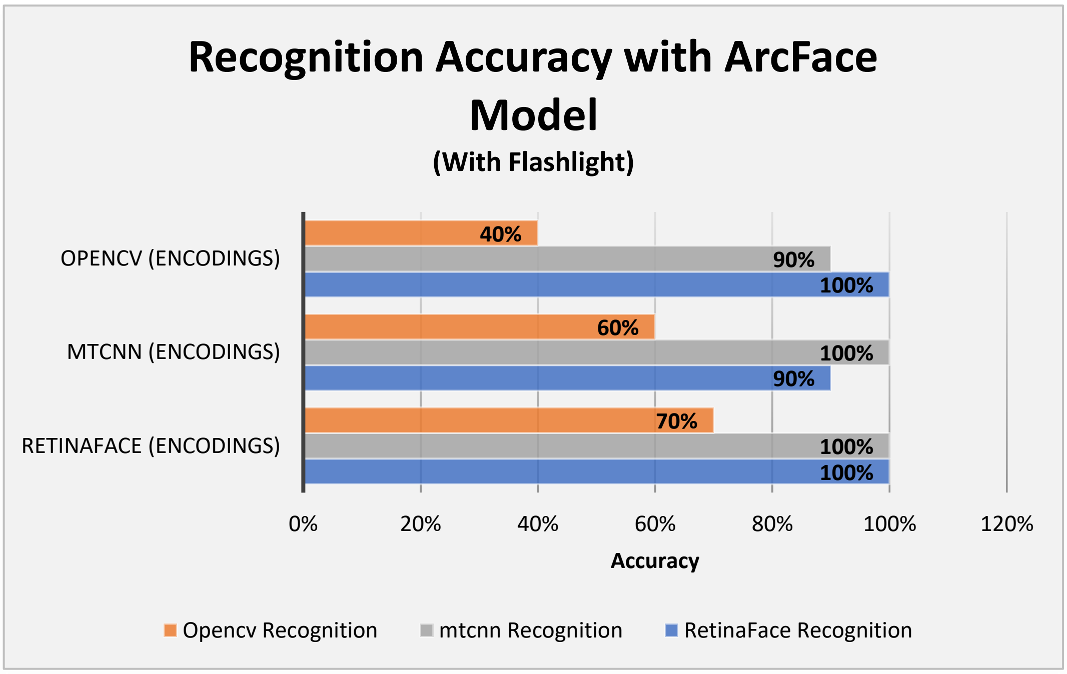


Figures 6.2 Smallest ArcFace Cosine Distance in 20 Image Dataset

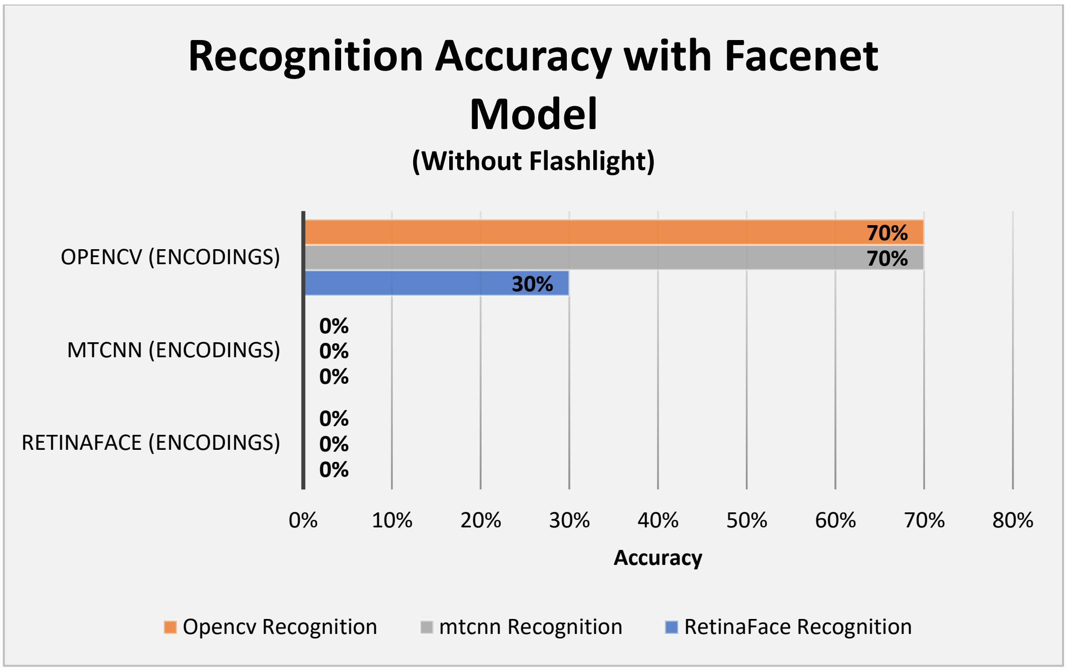


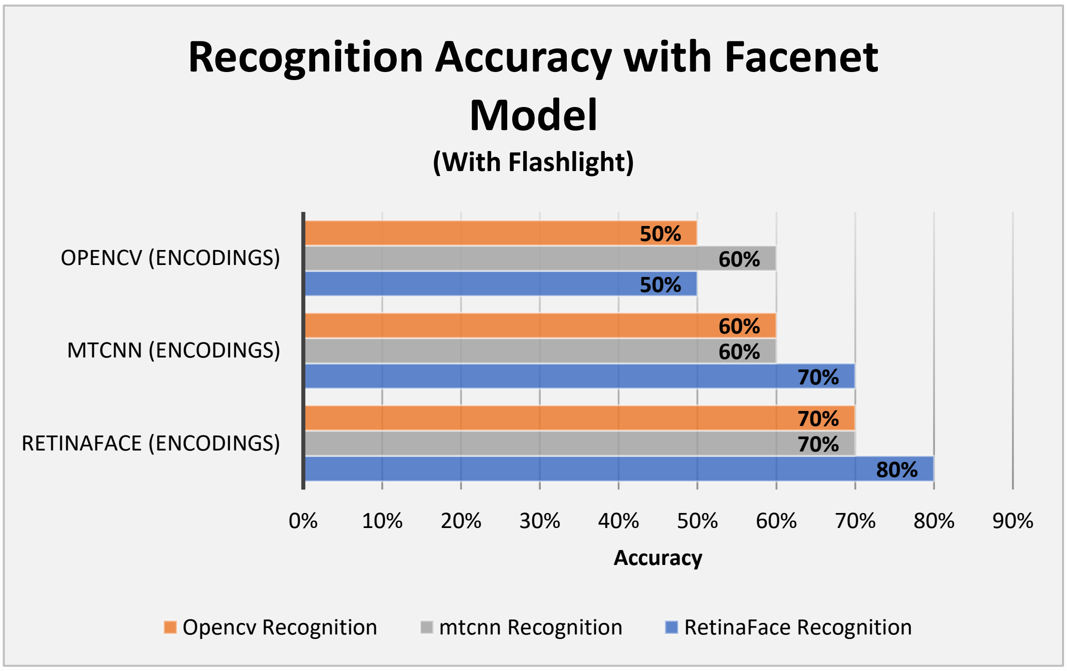
Figures 6.3 Smallest Facenet Cosine Distance in 20 Image Dataset





Figures 6.4 Recognition Accuracy with ArcFace in 20 Image Dataset





Figures

6.5 Recognition Accuracy with Facenet in 20 Image Dataset

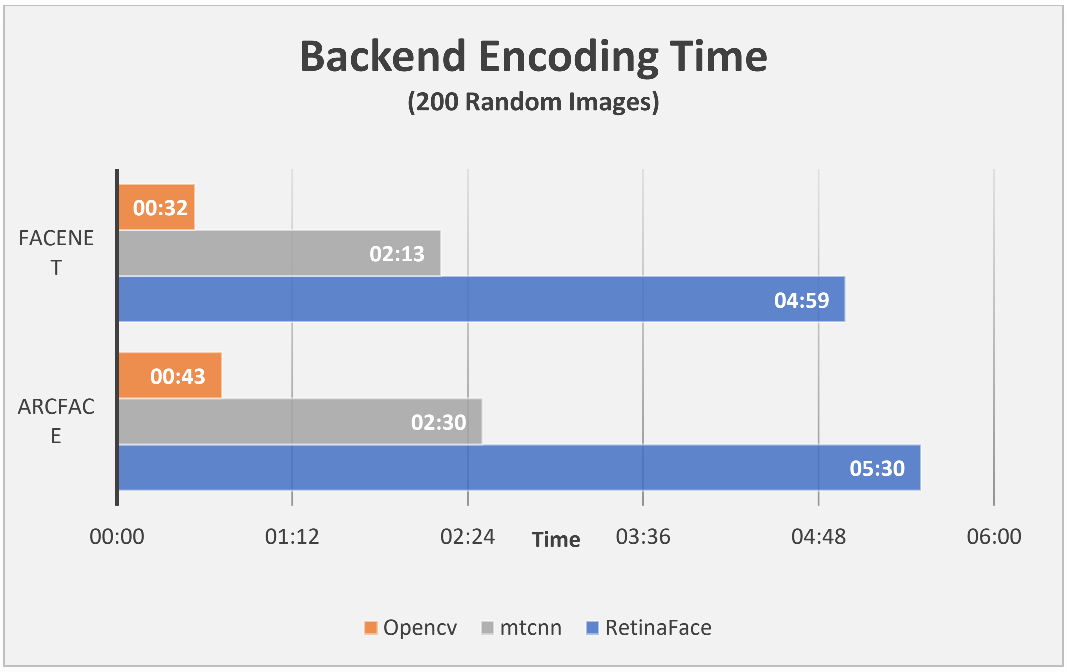
Table 6.1 Table of Average Distance from 2 Face Recognition Model in 20 Image Dataset

Information	Without Flash	With Flash
Average Distance Arcface	0,359924066	0,292851634
Average Distance Facenet	0,028158329	0,158346398

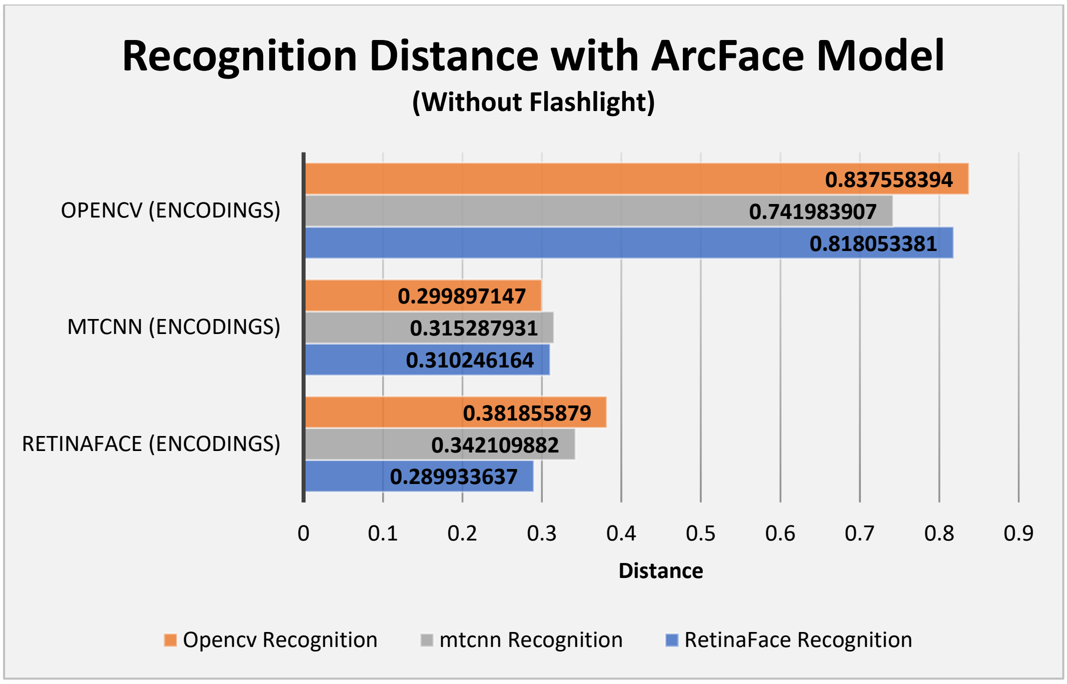
Table 6.2 Table Average Accuracy from 2 Face Recognition Model in 20 Image Dataset

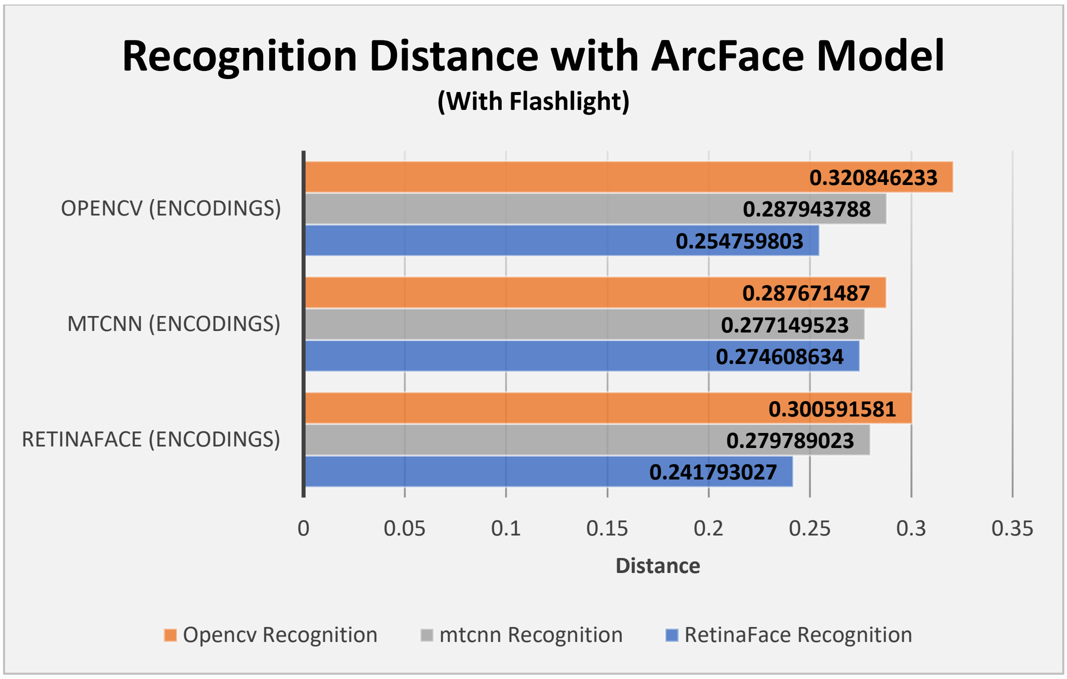
Information	Without Flash	With Flash
Average Accuracy ArcFace	78%	83%
Average Accuracy Facenet	19%	63%

Then followed by a second experiment by performing real-time facial recognition using a photo dataset of 200 photos and where each person only has one photo dataset, the experiment resulted in a longer and varied encoding time while the results for some had poor accuracy, and some had same or higher accuracy than before. From the trial in scenario two, regarding the combination of the face recognition model and the backend detector, it has been concluded that the ArcFace face recognition model is more robust than the facenet face recognition model, and for the retina face backend detector it helps the recognition to be more accurate compared to mtcnn and opencv, so the combination which will be used is ArcFace with retinaface as the advanced method and ArcFace with mtcnn as the basic method.

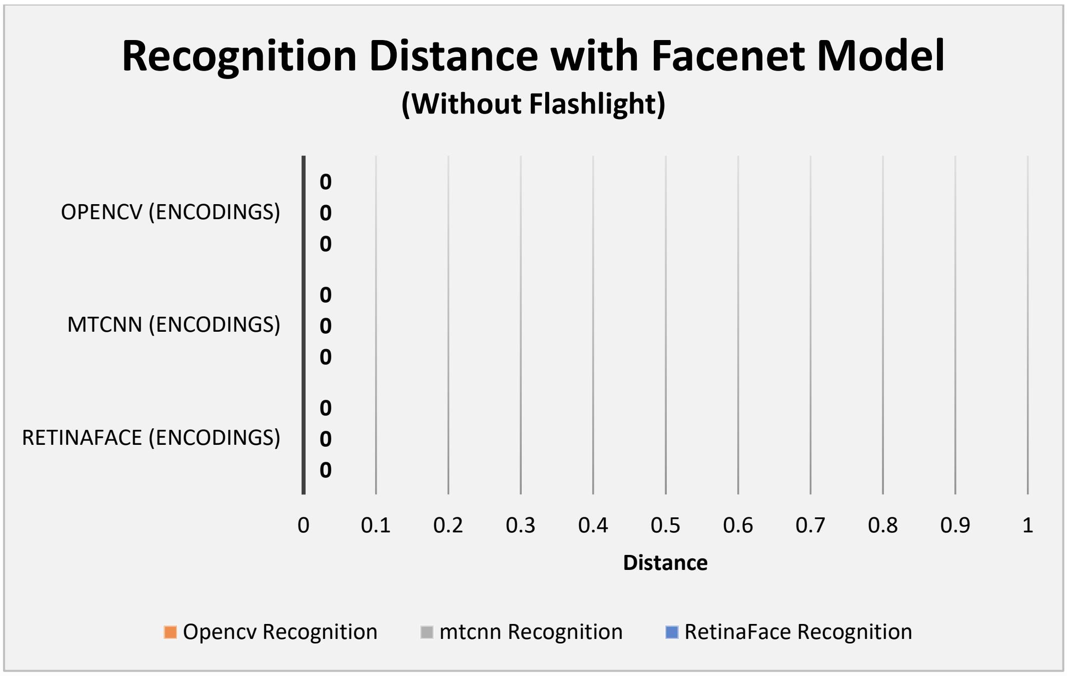


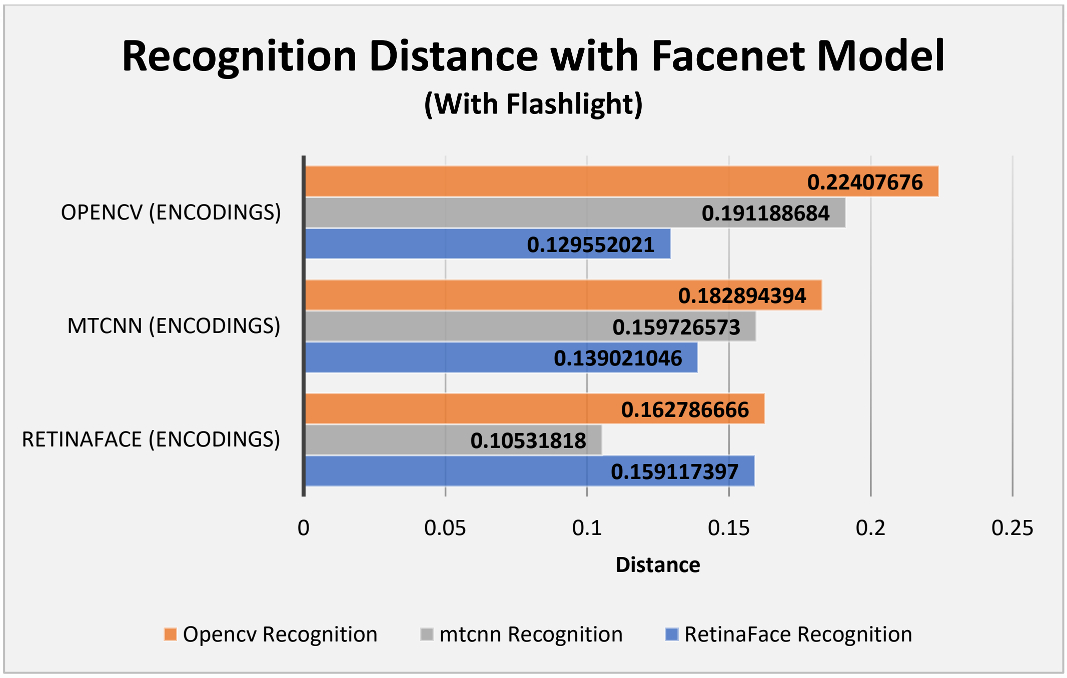
Figures 6.6 Encoding Time in 200 Image Dataset



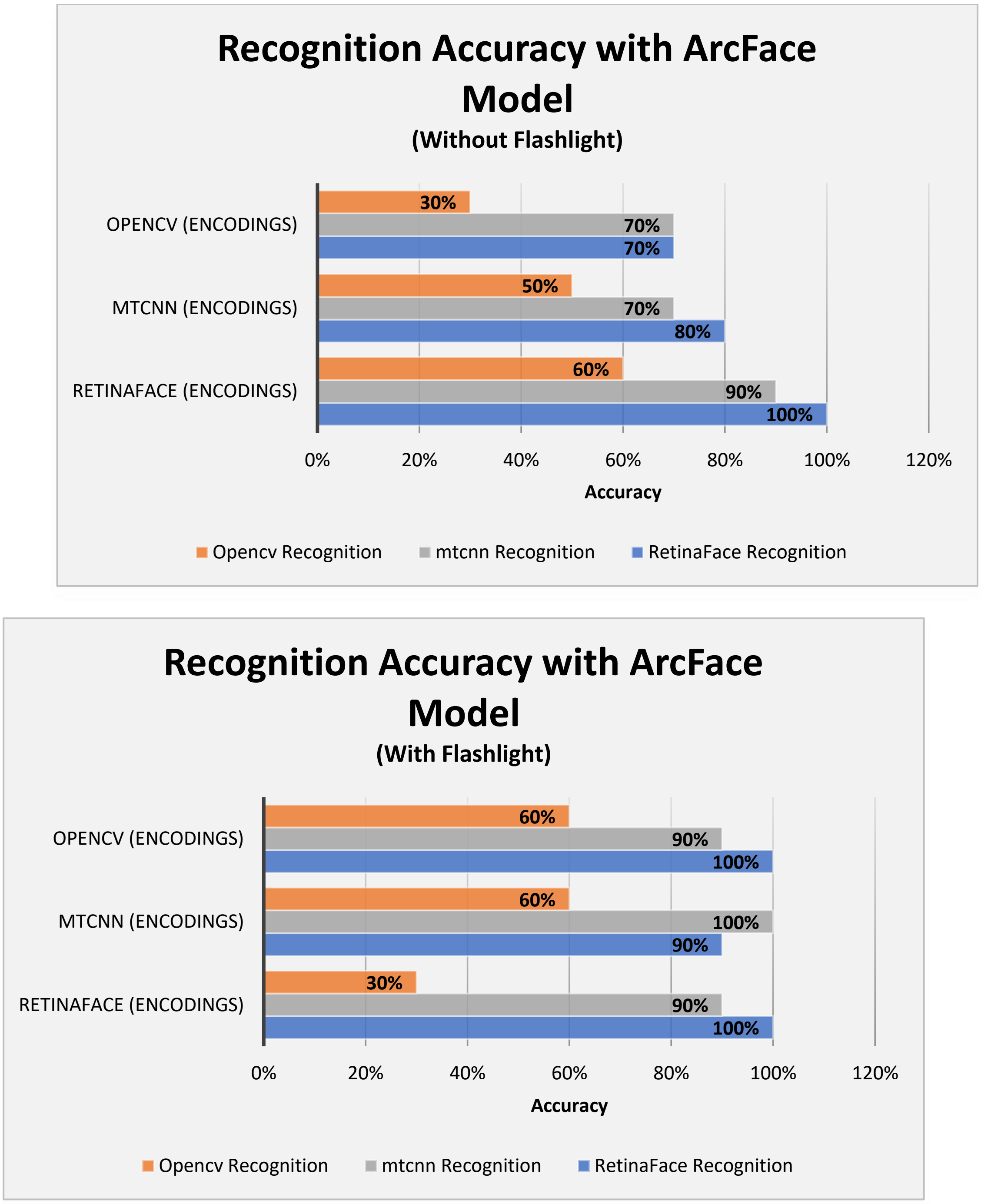


Figures 6.7 Smallest ArcFace Cosine Distance in 200 Image Dataset

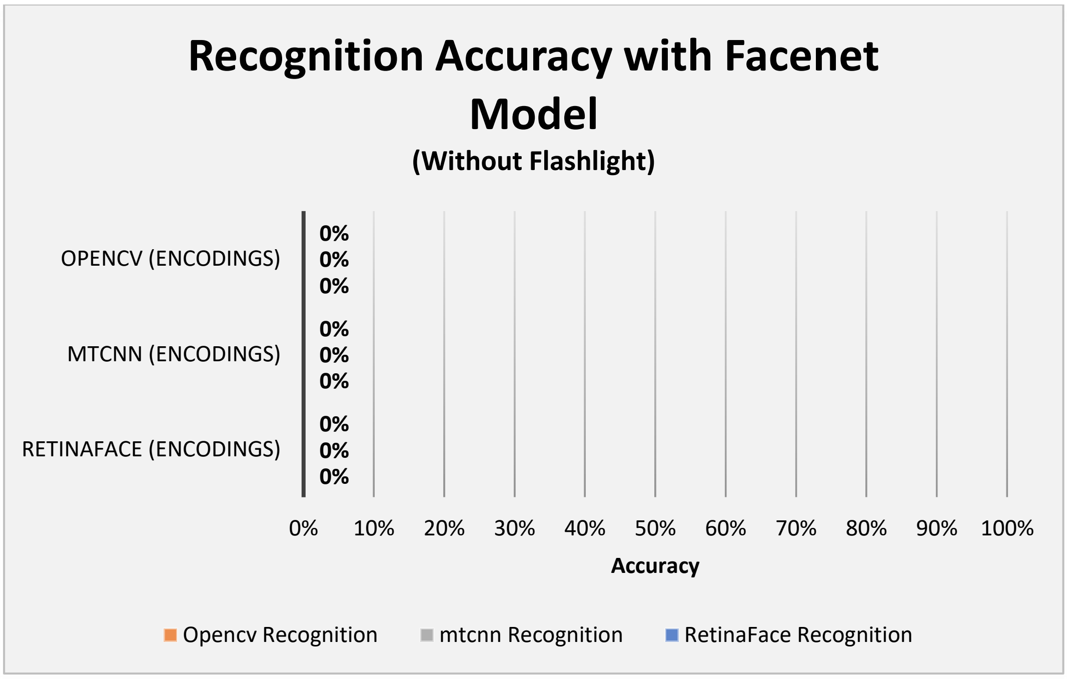


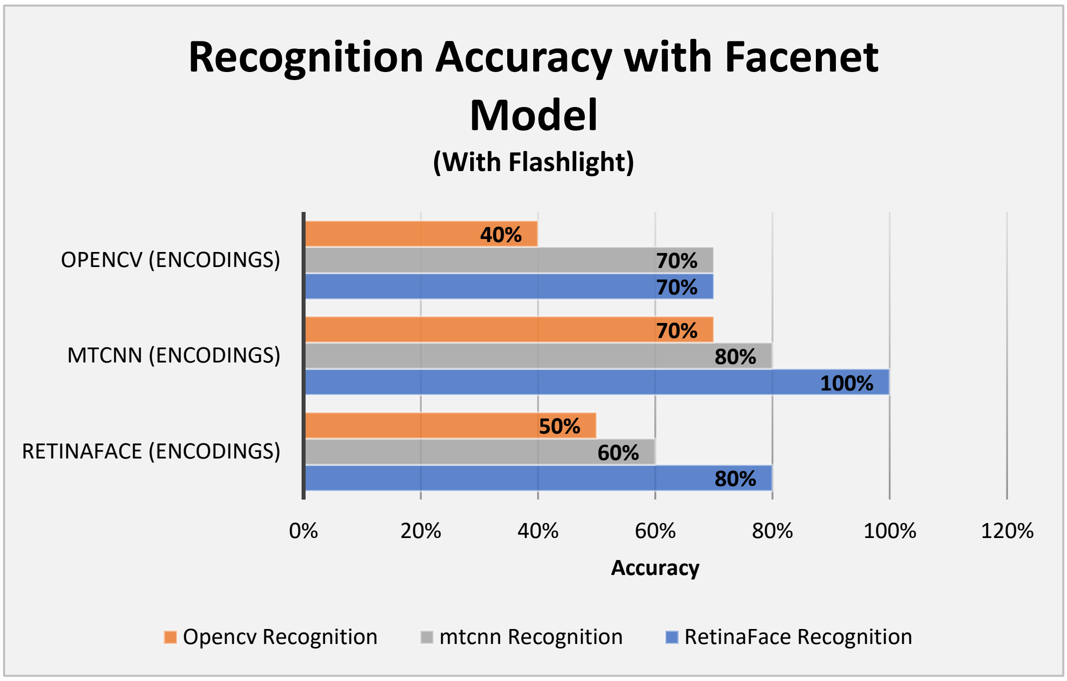


Figures 6.8 Smallest Facenet Cosine Distance in 200 Image Dataset



Figures 6.9 Recognition Accuracy with ArcFace in 200 Image Dataset





Figures 6.10 Recognition Accuracy with Facenet in 200 Image Dataset

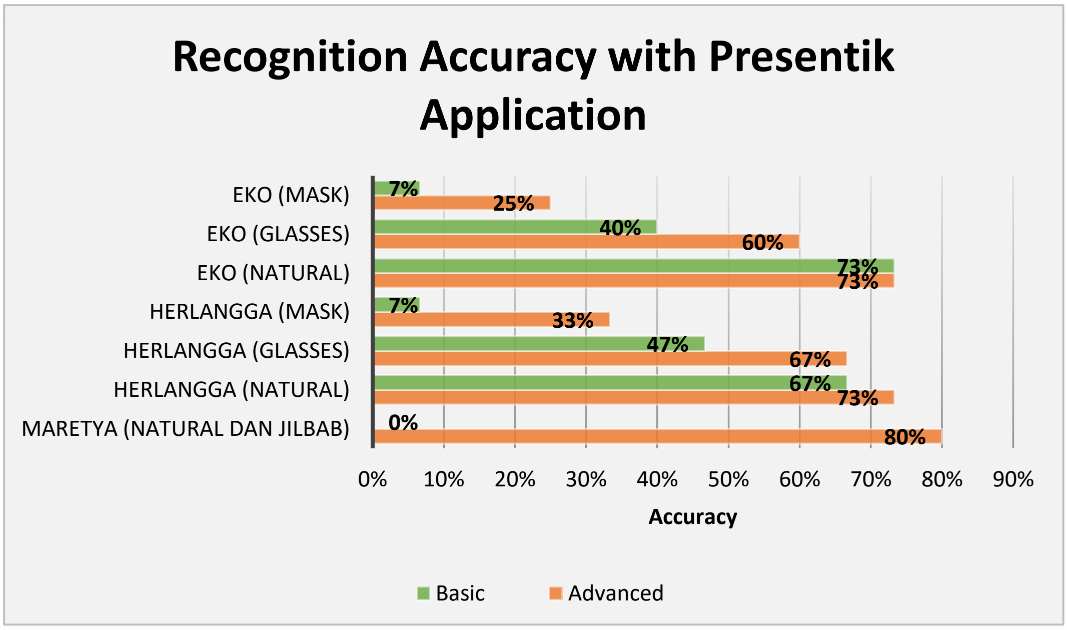
Table 6.3 Table of Average Distance from 2 Face Recognition Model in 200 Image Dataset

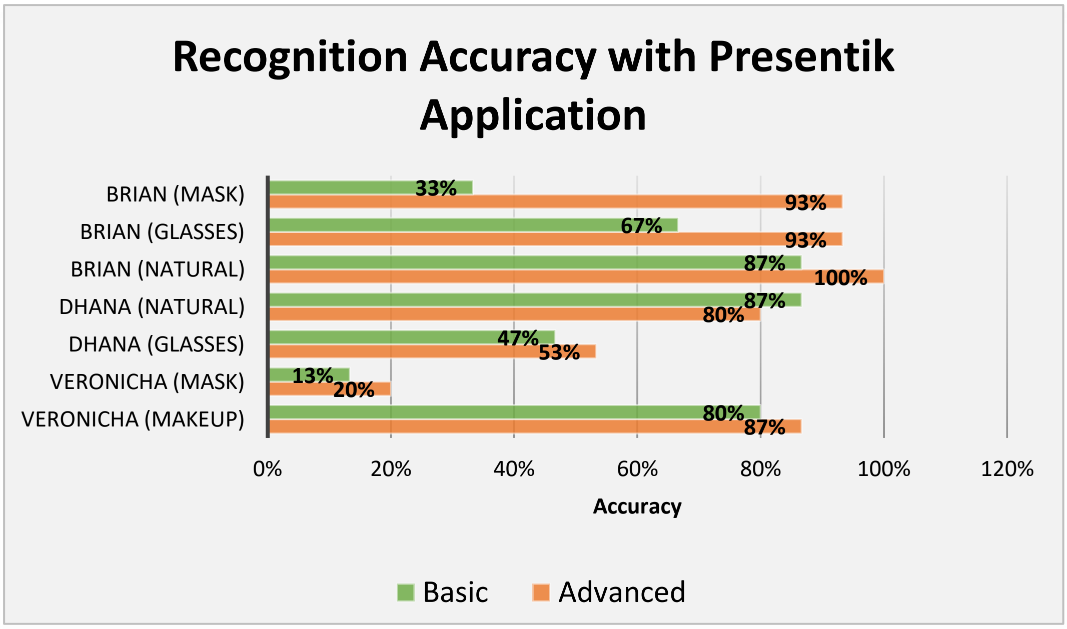
Information	Without Flash	With Flash
Average Distance Arcface	0,481880703	0,280572567
Average Distance Facenet	0	0,161520191

Table 6.4 Table of Average Distance from 2 Face Recognition Model in 200 Image Dataset

Information	Without Flash	With Flash
Average Accuracy ArcFace	69%	79%
Average Accuracy Facenet	0%	69%

Then the third test is carried out when the results have been obtained in determining the best combination of face recognition model and backend detector, the combination will be implemented into the application and then tested according to the experiments that have been carried out in chapter V. From the conclusion of the experiment, it is evident that the combination of face recognition model ArcFace with the Retina Face detector backend, it is far superior to the combination of the facenet face recognition model with the MTCNN backend detector. This experiment was conducted using 11 datasets from volunteers who were kind enough to provide facial data for research, and the following is a chart of the results of several volunteers who are willing to do a direct trial using this presentik application.

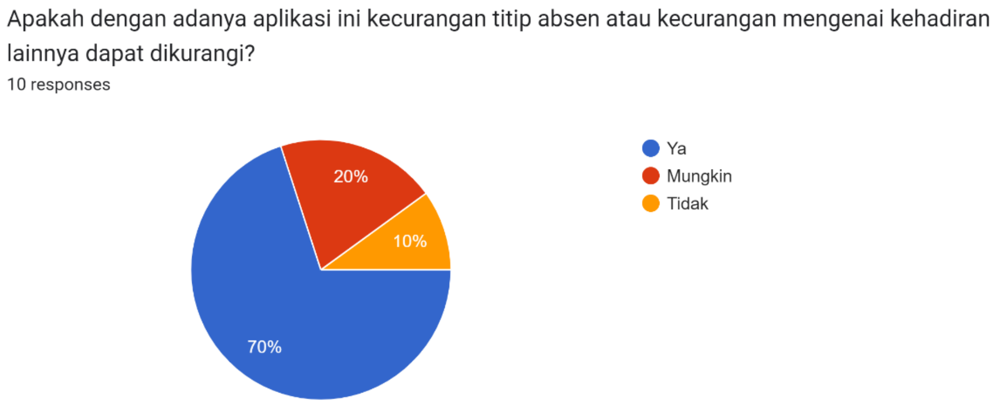




Figures 6.11 Recognition Accuracy with Presentik Application

In the chart above, it can be proven that almost all of the advanced algorithms are superior to the basic algorithms, and the greatest accuracy is obtained at 100% with natural face classification (without makeup or additional decoration) using the advanced algorithm.

Based on form functionality testing distributed based on the link: https://forms.gle/Kq8HW8RPpwebJbuA7, resulted that around 70% (out of 10 respondents) answered agreeing that this application can reduce fraud such as "titip absen" and 20% (out of 10 respondents) answered maybe and 10% (out of 10 respondents) answered no.

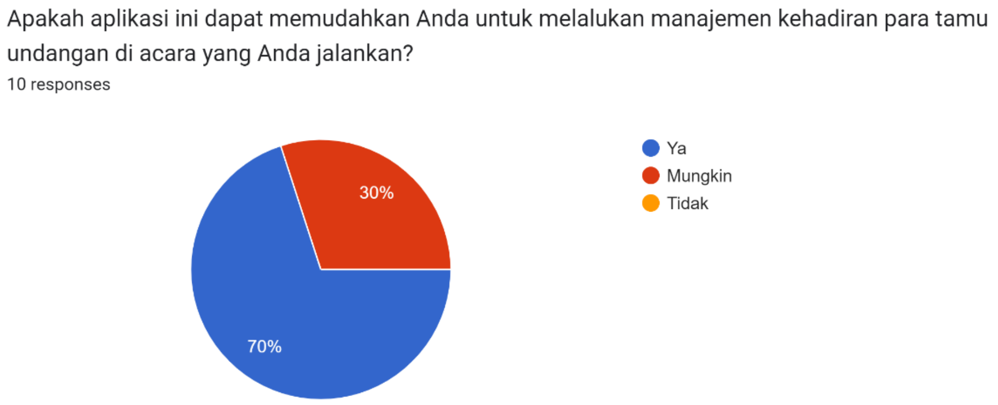


Apakah dengan adanya aplikasi ini kecurangan titip absen atau kecurangan mengenai kehadiran lainnya dapat dikurangi?

10 responses

Figures 6.12 Respondent result for “Titip Absent” question

Then about 70% (out of 10 respondents) answered agreeing that this application makes it easier to do attendance management and 30% (out of 10 respondents) answering maybe.



Apakah aplikasi ini dapat memudahkan Anda untuk melalukan manajemen kehadiran para tamu undangan di acara yang Anda jalankan?

10 responses

Figures 6.13 Respondent result for manage attendance question

6.2 Discussion

Based on the results of testing research data that has been carried out by researchers, the following are the results of the discussion obtained:

1. The CNN method is used to perform face recognition.
2. In testing the CNN method, the data is taken through the kaggle dataset to determine the combination of the face recognition model and the backend detector at the beginning and the data is retrieved through google forms filled in by the volunteer as guest data in the presentik application.
3. The use of the CNN method is used to perform feature embeddings or facial representations that will be stored in the database and will later be called for face recognition or validation.
4. For fraud such as "titip absen" can be reduced by adding an attendance feature at the event venue with a distance of 100m and checking manually in the attendance log provided.
5. Application development using Python as backend programming language and Dart as frontend programming language
6. In application development, the CNN method is used using the deepface library to facilitate the development of existing applications.
7. The selection of the face recognition model and backend detector is very influential in the accuracy generated in performing face recognition.
8. The amount of facial data of each person is certainly very influential with the accuracy of the resulting facial recognition because the more data each person has, the more chances that person's face will be recognized.
9. The use of the deepface library helps in facilitating research and system development, and also helps with data normalization or accurate face reading or detection.
10. The use of Google Cloud Platform as a backend server using Cloud Run and Docker as its facilities can help the use of endpoints with very small server latency.
11. Docker used is python:3.10-slim with running using gunicorn which has 1 worker and 8 threads
12. The cloud run used has a CPU specification of 4 with 8 GB of RAM, with a request timeout of 3600 seconds. for instances a minimum of 2 and a maximum of 10