Applied Machine Learning Intensive (AMLI), Summer 2021
Capstone Report

TEAM 1

July 29, 2021

Sam Lefforge, Department of CSCE, psleffor@uark.edu
Alejandra Barroso, Department of CSCE, ab129@uark.edu
A'Darius Lee, Department of CSCE, ajl017@uark.edu

1 Abstract

Facial recognition is a biometric technology that allows a person to be uniquely detected and identified by comparing and analyzing facial patterns and physical characteristics. With its potential for a wide range of applications, facial recognition technology has attracted a great deal of attention and has been improving over the years. The machine learning algorithms and methods that are needed for facial recognition have been constantly modified and reviewed by engineers. We have created a project that aims to detect and recognize the 21 students in the Advanced Machine Learning Intensive(ALMI) Bootcamp at the University of Arkansas, with deep neural networks. We have constructed an unique dataset that contains different illumination and pose of those 21 students. Our neural network model was trained on a dataset that we created and deployed via webcam. The dataset that we created for our model consists of frames taken from each video of the 21 students in the AMLI Bootcamp. Our model can run inference on 8 fps, thus realtime.

2 Introduction

Facial recognition was created within the 1960s by Woody Bledsoe, Helen Chan Wolf, and Charles Bisson and it was named ”man-machine”. The reason why it was named ”man-machine” was because they first needed to set up the facial highlights of a photo by getting coordinates of specific facial features before a computer could. Woodrow Wilson Bledsoe created the RAND tablet, which is a device that that input horizontal and vertical coordinates on a grid using a stylus that emitted electromagnetic pulses. The system was also very limited.

Facial recognition in static images and video sequences captured under unconstrained conditions is one of the most widely studied topics in computer vision. The facial recognition systems have seen wider applications in recent times on smartphones and in other forms of technology, such as surveillance, law enforcement, bio-metrics, marketing, etc. Traditionally, face recognition algorithms identify facial features by extracting landmarks such as the relative position, size, and/or shape of the eyes, nose, cheekbones from an image of the subject's face. These features are then used to search for other images with matching features. Our project is tasked to build a machine learning program that can detect a face given a video input. After it detected the face, it should identify that person based off data collected from our fellow classmates.

Facial recognition is the process by which a computer is able to associate an image with a unique identity. In our case, that identity is one of twenty-one students enrolled in the NACME Advanced Machine Learning Intensive Bootcamp at the University of Arkansas Campus. The main goal of our project is to build a functioning network that is capable of identifying every student in Google Machine Learning Bootcamp at University of Arkansas campus.

Like other facial recognition models our model initial suffered from bias whereas it would struggle to recognize individuals of a darker skin tone while functioning properly with individuals of lighter skin tone. There is also potential bias in the resolution of the photos. A photo with a lower resolution may have features misidentified due to lack of information. We tried to mitigate this by resizing all images to 64 by 64 before they enter the facial recognition model. Some bias may also come from pose. If our training poses are not diverse enough, then the model will likely perform poorly on images where the person takes on a different stature. The same can be said for lighting and occlusion. We tried to get a diverse mix of all three of these things in our training data by having the person spin in place while the camera follows them around, panning slightly up and down. This ensures that as they face the lights of the room from a variety of angles producing a diverse dataset for the model to train on. With that our model now functions properly with individuals of all skin tone. Our model does generalize well to the domain to which it will be applied. To test and validate the model, we will regularly test it by taking a picture of one of our classmates and see how well it can accurately predict the person in the photo.

3 Data Analysis

The following subsections present the dataset that we collected. Our dataset consists of 280 images of the student faces with different angles and illuminations.

3.1 Data Acquisition

In order to construct the facial dataset of 21 classmates, we initially collected data within our classroom to test how our model performed in a room of adequate lighting. This was accomplished by us taking a 5-second video of each classmate's faces, while making sure to get some variation in angle, expression, and illumination. To get this variation we had each student stand in an exact spot in the classroom with them starring directly at the camera. From there we had them spin around in a circle while continually starring at the camera so that we could get each students face with a light hitting their face at different angles. The filmed videos were then parsed out into their individual frames. We ran the facial detection model over the individual frames of the video. For each frame, we cropped the image to be only the detected face and then used the computer vision library to resize the image to be 64 by 64 pixels. We repeated this process 80 times for each class. We found that our model was still struggling, so we collected more data in a different location with a different camera, so as to increase the diversity of lighting conditions, occlusion, and angle. This data was collected in the J.B. Hunt main auditorium, and instead of collect 80 frames for each class, we took a longer video and collected 200.

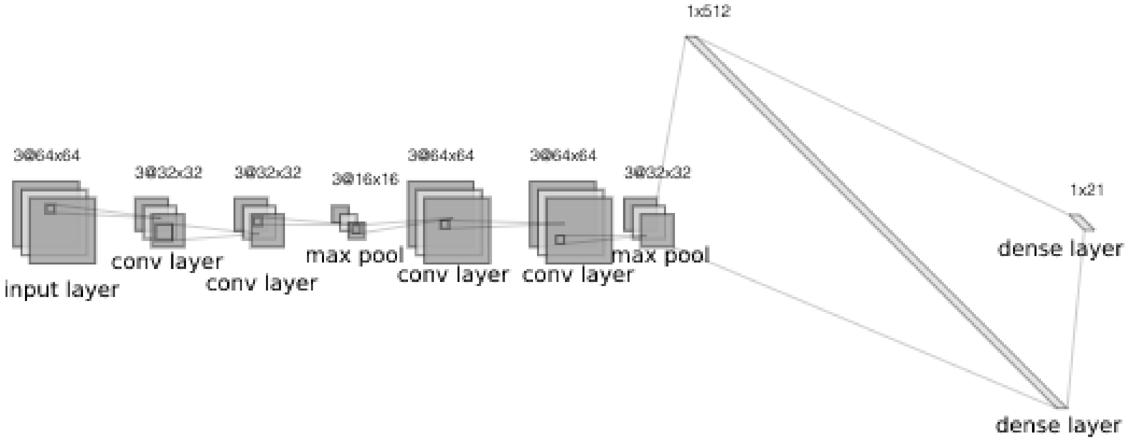


Figure 1: The recognition structure of our model.

3.2 Data Processing

The resolution of the original video was full HD (1080x1920 pixels). From the original video, we would use the facial recognition library to crop to just the face. After that, we resized the image to be 64 by 64 pixels so that the model is able to accept it. Then, we standardized each pixel of the image using the standardization formula in equation X.

After, we got the frames of the shortest video and used that to determine how many frames of each video will be used for training data and for testing data. We decided that we should use 80 frames from the classrooms and 200 frames from the J.B. Hunt auditorium.

4 Project Implementation

Our model operates by first cropping the frames from the videos to only display the face, and then puts the images into an array. Additionally, we looked at the video resolution and we edited the frames of the videos in order for them to be as consistent as possible. For the frames, we resized them, flipped the color channels, and rotated the frames.

The first step we took to approach our project was to create the 'draw_boxes_on_faces' function that successfully recognized faces in an image. The next step was getting the model to identify people by encoding a picture of each classmate and testing it by feeding it a picture of another classmate. The output of the function are booleans and will therefore return True or False depending on whether the face in the unknown video matches a face

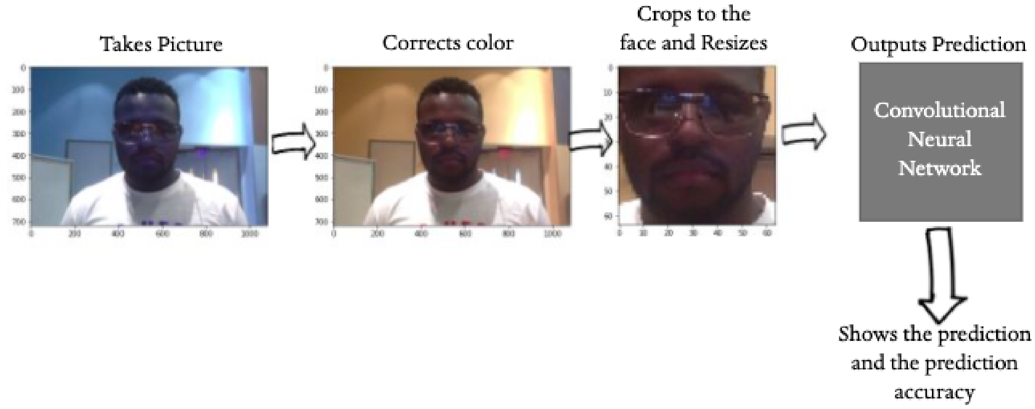

Takes Picture
Corrects color
Crops to the face and Resizes
Outputs Prediction
Convolutional Neural Network
Shows the prediction and the prediction accuracy

Figure 2: The overall pipeline of our model.

of a person in the array.

To implement the webcam, we created a 'take_photo' method that resizes the output to fit the video element and waits for capture to be clicked to take the picture. After it takes the picture, it names it "photo.jpg". It then goes into the 'face_identification' method and results in an array of True/False that shows if the unknown face matched anyone in the known_faces array. To get the results to display the names of our classmates instead of True/False, we created an array with our classmates names. We also created a new method that crops the picture to only show the face.

The model we created does all of the above and it predicts who the person is. The model displays its top three predictions and confidence score. The predictions are the student names and the confidence score is a decimal number that shows how confident it is in each prediction. We tweaked the model compared it to our old model and it slightly worked better.

An obstacle that we came across in our model was that it was having a difficult time recognizing darker skinned people. This bias comes from the data pre-processing phase. When we crop each frame to be only the face of the individual, we rely on the facial recognition library to detect a face. Sometimes, the library fails to recognize people with darker skin, resulting in less data for our model to train on.

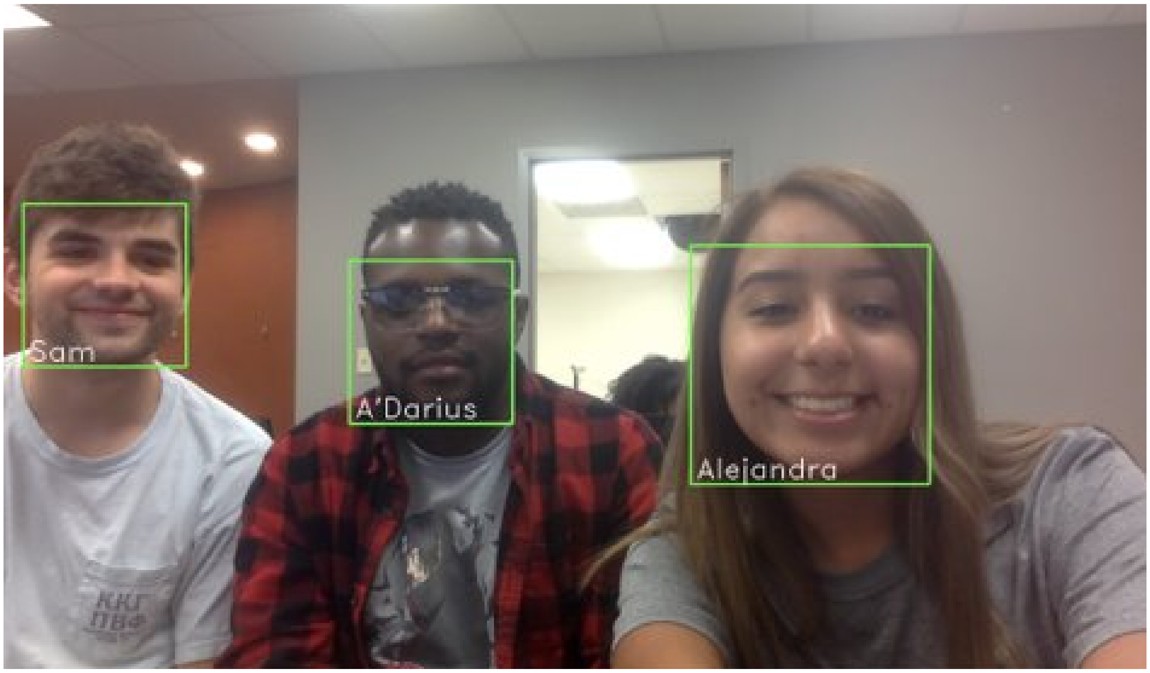


Figure 3: Deployed model in action. Note that the model correctly detects three faces which are in the field of view of the web camera.

5 Experimental Results

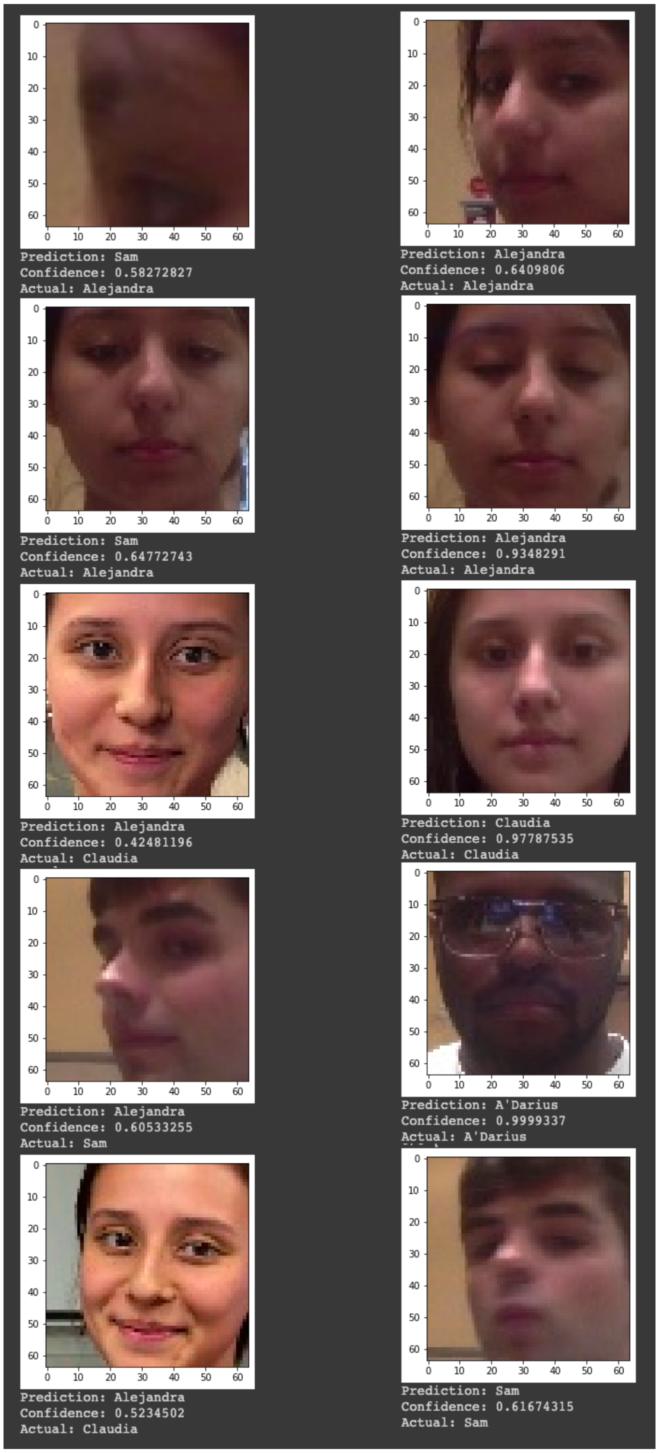
We used 80 percent of our data for training and 20 percent for testing. We evaluated our models quantitatively using accuracy, recall, precision, and F1 score. We also evaluated our model visually with a confusion matrix.

5.1 Qualitative Analysis

As you can see in figure 2, our model is able to correctly detect and identify our team. This was taken in lighting conditions different to what the model was trained on, and yet it was still reliably identifying each of us correctly.

Figure 3 sheds some insight into why the model failed at different points. The very first picture in the left column shows an example of a failure in the facial detection portion. The photo shows just a sliver of a person's neck, but the facial detection model thought the face was located there. As such, it makes sense that the identification portion would also fail.

In the figures below that, you can see some more puzzling failures of the model. Machine learning is a black box, so in cases like these it's hard to identify exactly what is going wrong, but we can take guesses. For example, in the fourth image down in the left column, the angle is a little weird, so it's possible that the model happened to not train on that



```
Prediction: Sam
Confidence: 0.58272827
Actual: Alejandra
```

```
Prediction: Sam
Confidence: 0.64772743
Actual: Alejandra
```

```
Prediction: Alejandra
Confidence: 0.42481196
Actual: Claudia
```

```
Prediction: Alejandra
Confidence: 0.60533255
Actual: Sam
```

```
Prediction: Alejandra
Confidence: 0.5234502
Actual: Claudia
```

```
Prediction: Alejandra
Confidence: 0.6409806
Actual: Alejandra
```

```
Prediction: Alejandra
Confidence: 0.9348291
Actual: Alejandra
```

```
Prediction: Claudia
Confidence: 0.97787535
Actual: Claudia
```

```
Prediction: A'Darius
Confidence: 0.9999337
Actual: A'Darius
```

```
Prediction: Sam
Confidence: 0.61674315
Actual: Sam
```

Figure 4: Some example output from the model. On the left column are five incorrect outputs and on the right are five correct outputs.

many images from that specific angle. Additionally, in the bottom image on the left column, the person's head is tilted slightly. Perhaps there was not enough diversity in the training data in terms of rotation on this axis, resulting in an incorrect conclusion from our model.

One unsurprising trend in our model's incorrect prediction is low confidence. It intuitively makes sense that the model would be less sure of a prediction that does not match the ground truth. If you look at the right column which showcases correct predictions, you can see generally higher confidence scores.

The first image on the right column is at a bit of an off angle, so the confidence score is not as high as some of the other images, but the model still ultimately makes the correct prediction. In the second image, you can see that the person has their eyes closed, but again the model is able to predict correctly. As we were recording the data for the model to train on, it's natural that there would be a good amount of frames where the person is blinking, so it makes sense that the model is able to correctly identify people even with their eyes closed.

In the final image on the right column, the angle is not quite head on, and the image is a bit blurry, but still the model is able to correctly identify the person. Again, the nature of the way we collected our training data means that some blurry photos are present in our training data, so that explains how the model is able to perform well under those conditions.

5.2 Quantitative Analysis

We evaluated our model performance with following metrics: Accuracy, Precision, Recall and F1 Score. Those metrics are based on the confusion matrix,, which is a table that is often used to describe the performance of a classification model on a set of test data for which the true values are known. Within a confusion matrix, true positive (TP) and true negatives (TN) are the observations that are correctly predicted, while false positives (FP) and false negatives (FN) are the observations that are wrongly predicted. The true positive and true negatives are the diagonal component of the confusion matrix. The four metrics are represented as follows:

$$Accuracy = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN} + \mathrm{TN}} \tag{1}$$

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{2}$$

$$Recall = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{3}$$

$$F_1 = 2\frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

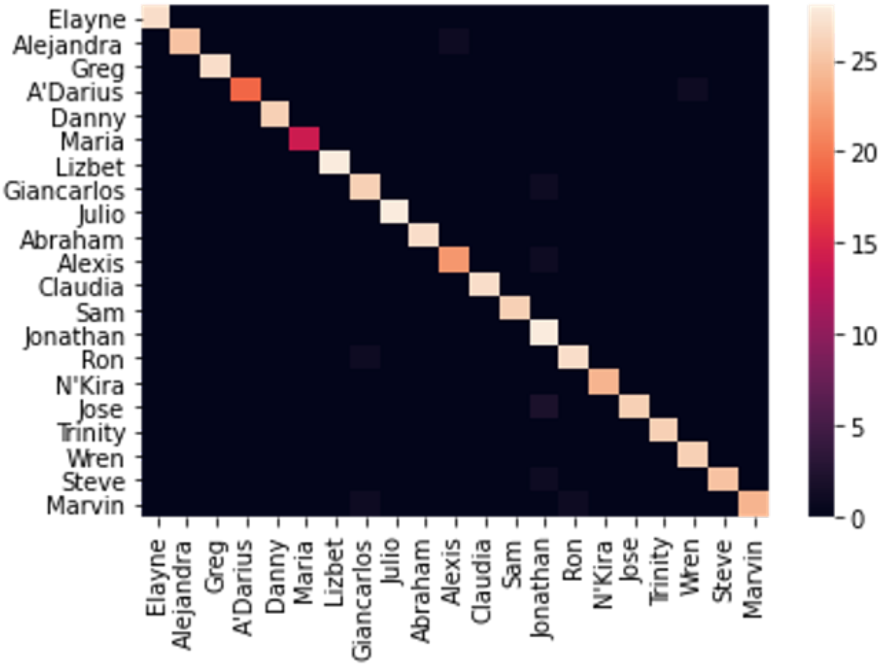


Figure 5: The confusion matrix of our own model

Figure 5 shows the confusion matrix of our own model and Figure 6 shows the confusion matrix of the model based off [1]. We suspect that there may be some kind of over fitting going on, because the amazing performance does not carry over to our webcam demo.

Here are our performance scores for our first model:

Accuracy	.9814
Recall	.98
Precision	.98
F1	.98

And here are our scores for the model based on the paper [1]:

Accuracy	.9907
Recall	.99
Precision	.99
F1	.99

As you can see, our second model performed slightly better than the first. The second model is more complex and is thus able to account for the many features of a face much better than the simple first model. There was no real difference in recall in precision, which is an

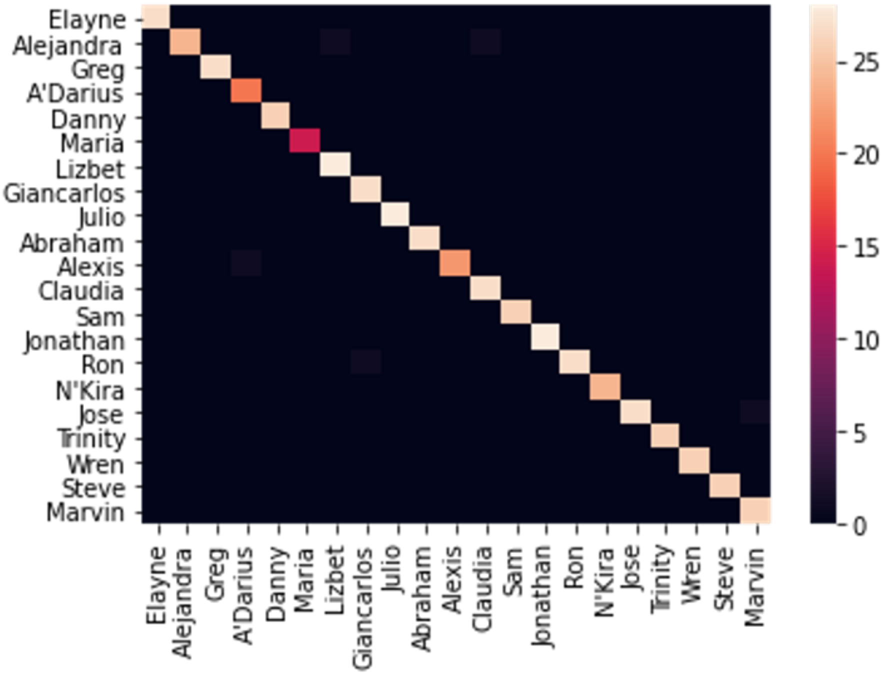


Figure 6: The confusion matrix of the model based off [1]

indicator that the model does not produce false positives any more than it produces false negatives, and likewise it does not produces false negatives any more than false positives.

6 Conclusion

Our work showed that a convolutional neural network can be trained to accurately detect 21 people. We did this using a facial detection library that provides the location of a face in an image, and our own recorded videos we took in the classroom.

This work is important because facial recognition as a technology can be applied to many important problems. Facial recognition is already being used on many smartphones as a form of identification. Other applications include accessibility features for the blind, and searching for missing people's digital footprint on social media.

One major improvement that could be made is run time. Our live facial recognition runs pretty slowly, usually only being able to process a few frames per second. Additionally, it isn't very accurate when tested in different lighting conditions and angles.

Some future work could be to collect more data and try training again on a convolutional neural network. We were training on only a couple hundred images per class, which is far below the standard for really robust facial recognition.

7 Acknowledgment

We thank our advisor, Dr. Le, our TA, Kashu, for the guidance and I appreciate the financial support from my family and sponsors.

Best,

Sam Lefforge
Alejandra Barroso
A'Darius Lee

References

[1] X. Wang, S. Wang, S. Zhang, T. Fu, H. Shi, and T. Mei, "Support Vector Guided Softmax Loss for Face Recognition," *arXiv e-prints*, p. arXiv:1812.11317, Dec. 2018.

[1] Marciniak, T., Chmielewska, A., Weychan, R. et al. Influence of low resolution of images on reliability of face detection and recognition. Multimed Tools Appl 74, 4329â4349 (2015). https://doi.org/10.1007/s11042-013-1568-8

[2] AbdELminaam, Diaa Salama, et al. âA Deep Facial Recognition System Using Computational Intelligent Algorithms.â PLOS ONE, Public Library of Science, journals.plos.org/plosone/article?id=10.1371

[3] Kaspersky. âWhat Is Biometrics Security.â Www.kaspersky.com, 13 Jan. 2021, www.kaspersky.com/resource-center/definitions/biometrics.

[4] Xiaob o Wang, Shuo Wang, Shifeng Zhang, Tianyu Fu, Hailin Shi, Tao Mei1. "Support Vector Guided Softmax Loss for Face Recognition." https://arxiv.org/pdf/1812.11317.pdf.

[5] âFacial Recognition System.â Wikipedia, Wikimedia Foundation, 13 July 2021, en.wikipedia.org/wiki/Facial$_{r}ecognition_{s}ystem.$

[6] âHistory of Face Recognition amp; Facial Recognition Software.â FaceFirst Face Recognition Software, 9 May 2019, www.facefirst.com/blog/brief-history-of-face-recognition-software/.