Pervasive Computing Project
Gender Identifier

By: Efe Kerem Aksoy & Mert Arslan

Introduction

Computers are intertwined with our daily life and thanks to pervasive computing they make our life easier and comfortable in many ways. A very important aspect they bring to our life. That is the safety and privacy aspect. Of course there is a long going debate about how electronic devices might have privacy issues but in our project we wanted to focus on how a pervasive computing device helps us to keep our privacy. That's why we designed an easy to use gender identification device for locker rooms. Of course it is not limited only to locker rooms; it can be used in every cabin that involves separation by the genders.

Our system can be easily used in every environment since it only needs a camera to operate. Our project only deals with identifying the gender of the person on the camera, not unlocking the door. That is the reason it is very affordable to integrate to the gyms and improve privacy and safety greatly. Other improvements can be done like predicting the age of the person if the gym has a policy that people under 15 years of age are not allowed in that gym. But in our case we wanted to keep it simple to only the gender.

Also it can favor to elderly and ones that are not able to see well. If they have a hard time reading the male or female writings at the door they can simply try to go in one of the cabins and if they are not allied they can understand they are in the wrong locker room. Also you can not be 100% sure that the members of the gym have a good will. Some male members might want to sneak inside of the female locker room. We believe gyms must prevent any kind of such attempts for their female members' safety and privacy. This problem can easily be solved with our program.

Related Work

Before beginning the construction of our program, we needed a feistable design to follow. To achieve this, we researched different literature studies which related to facial recognition and gender identification. We utilized their research findings and outcomes to help guide the development of our program. We also had help from Matlab's official youtube guide[1] and their neural network program[2].

[1] MATLAB. “Deep Learning with MATLAB: Training a Neural Network from Scratch with MATLAB.” YouTube, YouTube Video, 22 June 2017, www.youtube.com/watch?v=v_FKIH-ajkg&ab_channel=MATLAB. Accessed 14 Dec. 2023.
[2] “Deep Learning with MATLAB.” *Mathworks.com*, 2023, www.mathworks.com/videos/series/deep-learning-with-MATLAB.html. Accessed 14 Dec. 2023.

First study that will be covered is titled “Gender classification using custom convolutional neural networks architecture”[3]. The study's main purpose is to develop a CNN architecture that is less complex, requires fewer parameters, and smaller input image resolutions compared to existing methods like GoogleNet and AlexNet, and still get competitive results relatively.

“This custom CNN requires smaller input image resolutions and significantly fewer trainable parameters”. This quote from the study perfectly explains what we learned from the program to bypass the limitations on our code. We first implemented the smaller input images to our code and made all the images get sized to 32x32 pixels, this allowed for reduced computational load which helped us overcome one of our biggest issues later on with our code (the fact that the program was really slow). We also implemented his technique of fewer trainable parameters and had results which led to reduced Overfitting and even lower computational cost (is talked about in the annotation part of our code).

The second study that will be covered is titled “Real Time Gender Classification using Convolutional Neural Network” and this study has generalized the process of how they were able to create a real gender classification program.

This study was really helpful for us during the design process of our program as we needed a layout of how the program was actually going to work and process data (images and capturing of a photo)[4]. Thus, we followed their layout and organized the data into two categories ('Male' and 'Female') and set up image data stores for training and testing. We also mirrored their process of testing the network with a separate dataset and using a confusion matrix for validation as shown in the paper. The limitations with this paper however was everything else almost didn't suit our program, as we had a tighter deadline compared to them and we were using a different software.

Overall leveraging insights from existing facial recognition research and Matlab's guidelines, we crafted an efficient program that effectively balances performance and project constraints.

---

[3] Zaman, Kamaru. “Gender Classification Using Custom Convolutional Neural Networks Architecture.” ResearchGate, Institute of Advanced Engineering and Science, Dec. 2020, www.researchgate.net/publication/346549382_Gender_classification_using_custom_convolutional_neural_networks_architecture. Accessed 13 Dec. 2023.

[4] Mishra, Pooja, et al. “Real Time Gender Classification Using Convolutional Neural Network.” *ResearchGate*, EDP Sciences, 9 Aug. 2021, www.researchgate.net/publication/367956651_Real_Time_Gender_Classification_using_Convolutional_Neural_Network. Accessed 14 Dec. 2023.

Concepts of Operations

The possible stakeholders of our project are the gym owners or any other business that has gender separated cabins. But we mainly designed this project for the locker rooms in the gym. Our system can improve the quality of life by avoiding unwanted situations in the locker rooms.

Some scenarios that show implementation of our program. -

1- A person goes to the gym. In order to change her clothes before working out she smoothly enters the locker room by showing her face to a camera next to the door of the locker room. She has a very long and intense workout. After the workout she goes to the locker room to change her clothes but she becomes light headed and not quite sure which locker to go to. She first tries to go to the male locker room but the program realizes she is a female so the door doesn't open. She realizes she is in the wrong locker room. Then she goes to the other locker room and this time the program sees she is a female and unlocks the door

2- An male intruder wants to go inside a female locker room. He is aware that he should use the male locker room but still he tries to get into the female locker room. He was surprised that there is a camera next to the female locker room. He shows his face to the camera but the program detects he is a male and does not unlock the door. He gets angry and tries to force open the door. Then he realizes it is not going to unlock the door and shows his face again to the camera thinking maybe this time program doesn't run accurately and unlock the door for him. But fortunately the program runs correctly and detects he is a male and doesn't unlock again. The program prevents an intruder from entering the female locker room.

3- An elderly person wants to do light cardio in the gym to improve his cardiovascular health. He is not able to see well the writing of male in the door. That’s why he is not certain which locker room he should use. He decides to use the new program in order to see if he is in the right locker room. He first goes to the female locker room and the program detects he is a male. He realizes he is at the wrong door so he goes to the other one. This time male locker room's camera detects he is a male and opens the door. This way elderly people can avoid getting in the wrong locker room accidentally.

Systems Requirements

Our program will be using Neural Networks (convolutional neural networks to be specific) for gender classification. It will be trained on labeled images (male, female), process images by resizing them into proper dimensions, and different layerings and architecture models will be used for the neural network training. There will also be tests to the data to calculate accuracy, visualizations for the readability of accuracy on a matrice.Finally the program will also be able to capture real time predictions through the integrated camera.

Functional

- Image Acquisition: The system shall have the ability to acquire images from the datasets ensuring a diverse and representative collection for analysis.
- Image Preprocessing: The system shall have the ability to image preprocess which includes image resizing and normalizing.
- Model Training: The system shall be able to model training which is the process of learning to identify the genders based on the image.
- Model Testing: The system shall be able to test the trained model to evaluate the performance.
- Result Display: The system shall display the results in a matrice for easy evaluation of performance.
- Real-time Classification: The system shall be able to capture a photo in real time and evaluate the person's gender in real time.

Non-Functional

- Usability: The system should be user friendly and should work at the click of one button.
- Performance: The system's performance should be faster than 30 seconds for a fast smooth experience (depends on the GPU of the device).
- Accuracy: The system's accuracy should be over 80% with no biases.
- Privacy: The system must be private, and should run offline so there is no gathering of user data.
- Ethical Considerations: The device must be able to bypass with success for genderfluid identifiers in people.

Constrains

- System deadline: The time to build the system is very constrained.
- Matlabs: The system was forced to be built in Matlab, thus harder to translate previous experience into the project.

Prioritization of requirements using the MoSCoW approach

Must Have:

- The ability for image acquisition.
- The ability for model training and testing.
- Privacy
- Adhered to the system deadline and rules.

Should Have:

- The ability for real-time Classification by photograph.
- Good performance (30 seconds or less)

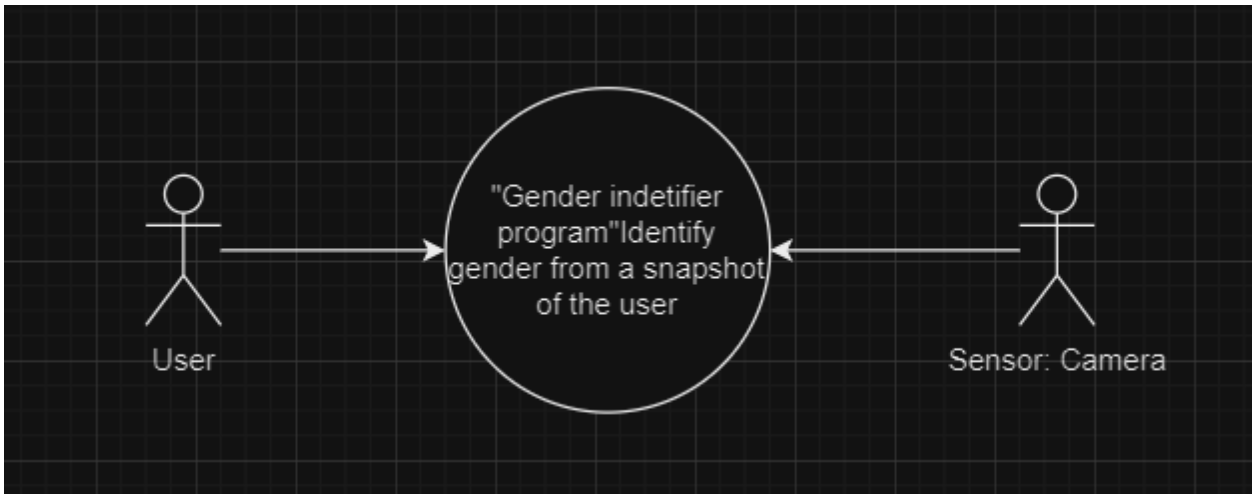
Could Have:

- Easy usability (only takes 1 click).
- Results get displayed.

Won't Have This Time:

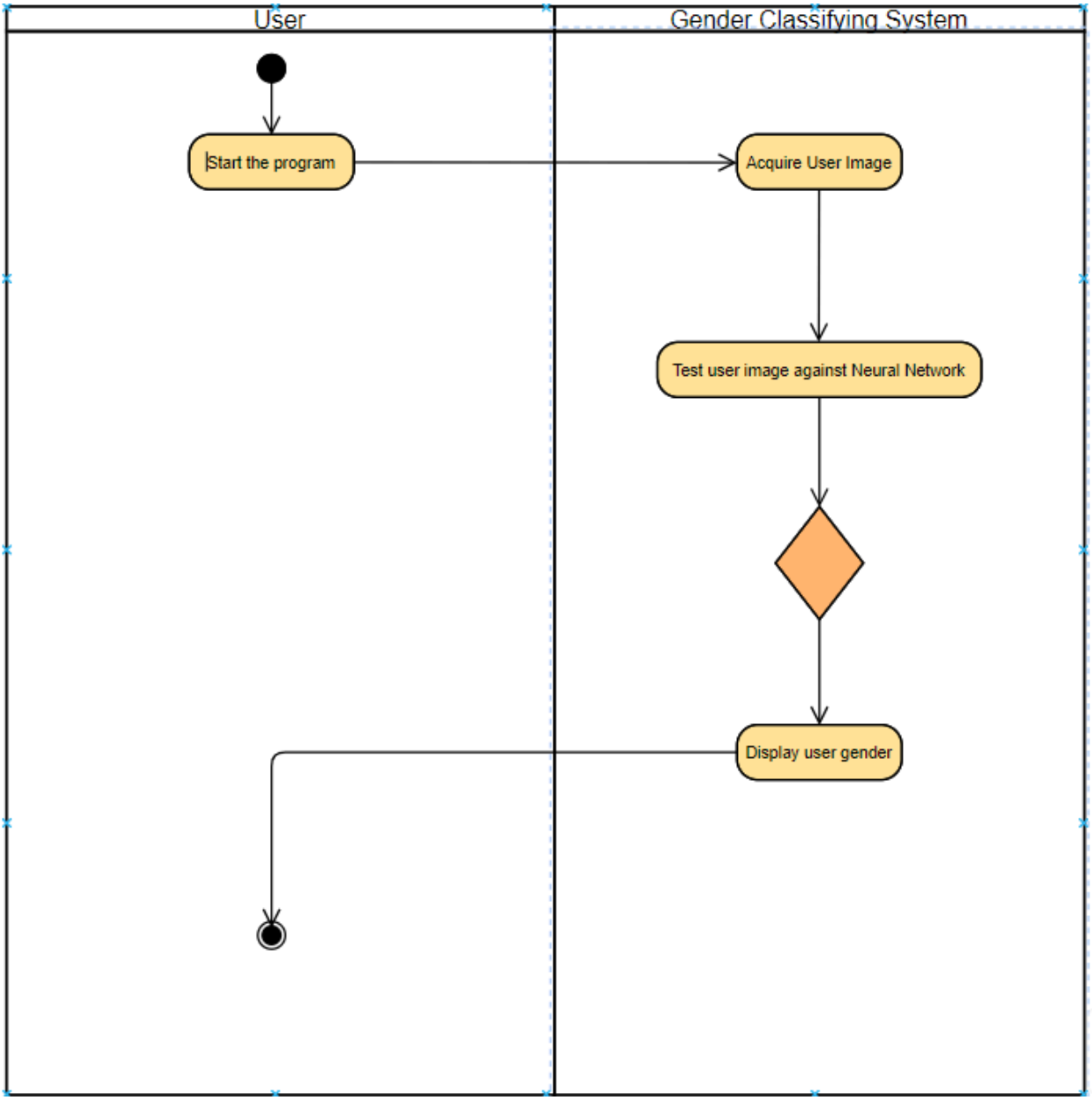
- The ability to recognise multiple people at the same time
- The ability for real-time Classification by video

Figure 1. Use case diagram



Name:	Gender identifier program
Description:	Identifies and classifies people based on gender
Precondition:	The system has access to a trained model capable of gender identification and images for classification are available either from a dataset or real time capture.
Postcondition:	The system accurately identifies genders.
Situational errors:	The system has misclassified the gender.
If an error happens:	The system outputs the misclassified gender.
Actors:	User Camera
Trigger:	User clicking run.
Standard process:	1. User clicks the system button 2. The system acquires their image 3. The system preprocess their image 4. The system classifies the their gender 5. The system displays the gender

Figure 2. Activity Diagram



System design

With a picture of someone a computer can detect a lot of things such as a person's age, race, gender, mood etc. We decided to choose only one topic since making all of it is too challenging for us. We agreed on a person's gender. After that we discussed a useful real world application of our gender identification by a photo that might be useful.

Hardware

- A computer that has internet connection and MATLAB installed
- A webcam

Software

- Matlab
- Training set size 2000
- Clasiffer; Artificial neural network

Project plan

Project schedule

Location	Date	Time
Lab class	27.12.2023	15.30-17.15
Lab class	30.12.2023	9.00-10.45
Mert’s house	1.12.2023	11.00-14.00
Lab class	4.12.2023	15.30-17.15
Kerem’s house	6.12.2003	17.00-20.00
Lab class	7.12.2023	9.00-10.45
Mert’s house	9.12.2023	14.00-17.30
Kerem’s house	10.12.20023	20.00-22.00
Lab class	11.12.2023	15.30-17.15
Kerem’s house	13.12.2023	15.00-18.00

Project planning and risk analysis

Date	Project planning/comments	Risk description
27.12.2023	Find a project idea	We might not able to find a good idea
30.12.2023	Find a past study similar to our project	A similar study does not exist on the internet or there is completely same study we would like to do
1.12.2023	Do distribution of work and decide on which day we work on what part of the project.	Not agreed on the same work distribution
4.12.2023	Mert does the sampling and Kerem works on the testing	The gathered data might not be diverse, leading to a biased system.
6.12.2003	Mert does the sampling and Kerem works on the testing	The testing data might not be diverse, leading to a biased system
7.12.2023	Mert does the sampling and Kerem works on the testing	Both the trained and the tested data might not be diverse leading to a biased system.
9.12.2023	Do a final testing of the code	Code might not work correctly
10.12.2023	Write the slides	Can not find appropriate images for the slides
11.12.2023	Finish the slides and work on documentation Rehearse the presentation	Rehearsals are not good as we planned
13.12.2023	Finish the documentation	Can not finish the documentation on time

Week 1 2 Tasks

	Task	Group member	Timeline
☐	Research	Mert/Kerem	27 Nov - 3 Dec
☐	Code	Kerem	27 Nov - 3 Dec

Week 2

	Task	Group member	Timeline
☐	Sampling	Mert	4 - 10 Dec
☐	testing	Kerem	4 - 10 Dec

Week 3

	Task	Group member	Timeline
☐	Documents/slides	Mert/Kerem	11 - 14 Dec

Testing

Our neural network was trained with 2000 samples for both datasets and our testing was conducted with 300, thus we had a 85% to 15% ratio. The training set had surprising results as per the small category of training data it was able to crush our expected accuracy of 80% (the actual accuracy is discussed below) and is able to detect all races of people with different features without bias.

Figure 3. Confusion Matrix

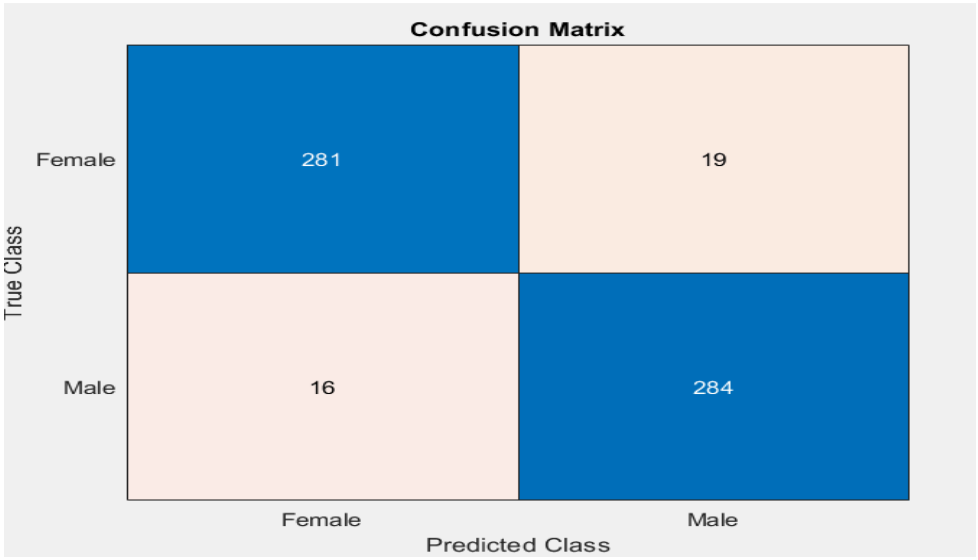


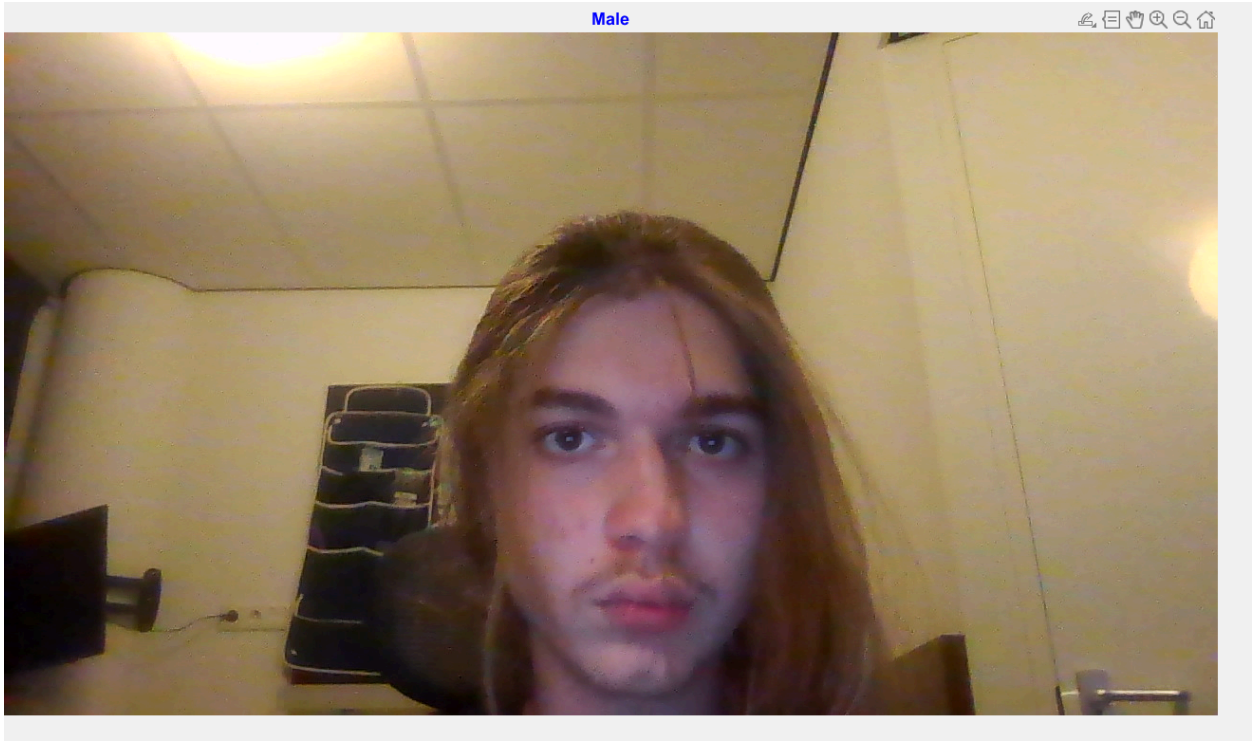
Figure 3 here shows the confusion matrix of our code, here we can see that 281 out of the 300 female samples were predicted correctly, and 284 out of 300 male samples were also predicted correctly. Hence our program can be labeled as non biased, which was a part of the non-functional requirements we had which we completed with success.

Figure 4. Final Results

```
>> genderClassifier
Training on single GPU.
Initializing input data normalization.
|=========================================================================================|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Mini-batch  |  Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |     Loss     |      Rate       |
|=========================================================================================|
|       1 |           1 |       00:00:01 |       37.50% |       1.1412 |          0.0010 |
|       1 |          50 |       00:00:03 |       92.19% |       0.2566 |          0.0010 |
|       2 |         100 |       00:00:04 |       90.62% |       0.3362 |          0.0010 |
|       3 |         150 |       00:00:06 |       96.88% |       0.1528 |          0.0010 |
|       4 |         200 |       00:00:07 |       98.44% |       0.1107 |          0.0010 |
|       5 |         250 |       00:00:09 |       98.44% |       0.0899 |          0.0010 |
|       5 |         300 |       00:00:10 |       93.75% |       0.1508 |          0.0010 |
|       6 |         350 |       00:00:12 |       95.31% |       0.1096 |          0.0010 |
|       7 |         400 |       00:00:13 |       93.75% |       0.1926 |          0.0010 |
|       8 |         450 |       00:00:15 |       98.44% |       0.0877 |          0.0010 |
|       9 |         500 |       00:00:16 |       98.44% |       0.0661 |          0.0010 |
|       9 |         550 |       00:00:18 |       96.88% |       0.0989 |          0.0010 |
|      10 |         600 |       00:00:19 |       98.44% |       0.0792 |          0.0010 |
|      10 |         620 |       00:00:20 |      100.00% |       0.0584 |          0.0010 |
|=========================================================================================|
Training finished: Max epochs completed.
Accuracy:
    0.9417
```

From figure 4 it can be understood that our functional and non functional goals were achieved. This can be seen as we were able to crush our minimum accuracy rate of 80% and were able to obtain a 94% accuracy from our testing data. On top of that, the program only takes 20 seconds to run which obliterated our expectations as well. Our hypothesis on why our accuracy and the program run time was very fast is due to the fact that we translated our findings from the study papers we read onto our code.

Figure 5. Proof of results



In figure 5 we can see the snapshot feature of our identifier in effect, and we can observe the person in the picture being identified as a male successfully even though he has long hair. This shows that “gender fluid” identifiers such as makeup on men or short hair on women still work with our program.

Reflection

It was the first time we worked as a team. Evendough it was our first group project I think we finished smoothly. We know that not every day or hour we will both be free for both of us but we managed to create that time every week. Also we believed that our project is realistic and applicable in the real world. With few upgrades it can be a legit working program in the gyms.

Some improvements we could make to further improve our program would be to increase our training and testing data. This would stabilize our program further and make the readings even more reliable. Another improvement we could have made if we had more time would have been to add real time gender identification through video cameras live, this would allow our program to not only be used in lockers but also in shops for other purposes such as finding out what percentage of their customers are male or female.

The biggest challenge we faced was creating the Convolutional Neural Network, however with the help of research papers and from our lab classes we were able to create an efficient program, which was able to exceed our goals that were stated above.