	# Hacettepe University Computer Engineering Department **BBM479/480 End of Project Report**

Project Details

Title	Class Attendance With Face Recognition
Supervisor	Nazlı İkizler Cinbiş

Group Members

	Full Name	**Student ID**
1	Pelin FİLDİŞ	21827412
2	Enes Taha Elmaci	21990872
3	Abdulkadir Alaca	21827019
4		

Abstract of the Project (/ 10 Points)

Explain the whole project shortly including the introduction of the field, the problem statement, your proposed solution and the methods you applied, your results and their discussion, expected impact and possible future directions. The abstract should be between 250-500 words.

Our project aims to address the outdated and time-consuming process of manual attendance checks in educational settings by leveraging the advancements in technology. Despite numerous aspects of our lives being refreshed and updated, attendance checks have remained largely unchanged, relying on paper-based lists and individual sign-ins. This traditional approach not only disrupts the focus of both students and instructors but also consumes a significant portion of valuable class time.

As computer engineering students, we recognized the need for a more efficient and streamlined attendance management system. Our proposed solution involves the development of an automated attendance system that utilizes mobile phone cameras and artificial intelligence with face detection and recognition techniques. By capturing snapshot during class and employing face detection and recognition algorithms, our system eliminates the need for manual sign-ins, paper lists, potential errors and the associated disruptions.

We have made our data set from our friends and some celebrities. We obtain their photos individually first. After that, we obtain group photos that include our friends and celebrities that we select.

Throughout the project, we encountered various limitations and challenges. However, despite these constraints, our solution showcases promising results. By automating the attendance process, we anticipate several significant benefits. The system reduces disruptions during lectures, eliminates the need for physical signatures, and enhances overall classroom focus. Additionally, the recorded snapshots serve as tangible evidence in cases of attendance disputes or confusion.

In conclusion, this project represents an innovative and practical approach to attendance management in educational environments. By harnessing the power of artificial intelligence and computer vision, we have developed an efficient and reliable system that offers substantial advantages over traditional methods.

The potential future directions for this project include scalability, further refinement of the neural network model, and exploration of additional applications beyond attendance management. We developed this project with the intention of implementing it in real life in the future. Our aim was to assist teachers and students in making their school lives easier. Once it is put into use in real life, the plan is to obtain the necessary permissions and access BİLSİS to collect students' photos and automate the process of creating classes for the appropriate courses each semester.

Introduction, Problem Definition & Literature Review (/ 20 Points)

Introduce the field of your project, define your problem (as clearly as possible), review the literature (cite the papers) by explaining the proposed solutions to this problem together with limitations of these problems, lastly write your hypothesis (or research question) and summarize your proposed solution in a paragraph. Please use a scientific language (you may assume the style from the studies you cited in your literature review). You may borrow parts from your previous reports but update them with the information you obtained during the course of the project. This section should be between 750-1500 words.

Our project is dedicated to addressing the archaic and laborious method of manual attendance checks in educational environments by harnessing the power of technological advancements. While many aspects of our lives have undergone refreshing updates, attendance checks have unfortunately remained largely unchanged, relying on outdated practices such as paper-based lists and individual sign-ins. This conventional approach not only hampers the concentration of both students and instructors but also consumes a substantial amount of precious class time. Tracking attendance papers circulating in the classroom causes students to lose focus, and if the paper gets lost, it creates an even bigger problem. Additionally, sometimes students forget to sign the attendance sheet, resulting in them being marked as absent for the day. Furthermore, at the end of each term, the process of manually counting all the attendance papers becomes an additional burden to our teachers. This task is not only time-consuming but also prone to errors, leading to potential inaccuracies in the final attendance records. It is clear that the traditional method of attendance tracking needs a modern and efficient alternative.

Based on our research findings, we have made decisions regarding the methodology and techniques to address the attendance taking problem. We realized that utilizing face detection and recognition methods would be the key to our solution. In this problem, our initial task was to gather the faces of individuals in the classroom. To achieve this, we obtained permission from our close friends to capture their photographs from various angles. Initially, we collected approximately ten photos from each of our ten friends. We attempted to train the ResNET50[1] model using these photos. However, the results we obtained were much worse than expected. When we shared this situation with our advisor, she guided us by pointing out that our approach to the problem was incorrect. In fact, in our problem, each individual would not have multiple photos but rather a single photo obtained from BİLSİS. In this case, we couldn't use the ResNET model in its current state, which had performed poorly. We began to search previous studies about face recognition with a single photograph. During our investigation, we discovered that one of the state-of-art techniques suitable for this situation is the use of Siamese Network[10] architectures.

Siamese Networks are particularly useful in tasks like face recognition, signature verification, or object tracking, where the goal is to determine whether two inputs are similar or dissimilar.

We researched which networks are commonly used as the base model in the Siamese Network architecture. Based on the results we found on platforms like Medium, Github, and HuggingFace[6-9], we identified the three most frequently used models: ResNet50, SENet50[2], and VGG16[3]. At the end of the project, for the web component, we decided to determine the most suitable network through testing.

Based on the results we obtained from our collected dataset, we decided to use SENet50 as the base model. You can find more details about this stage in the “Methodology” and “Results & Discussion” sections.

After completing the stages related to the model, we began developing the web component that would bring our project to users. We had some knowledge of web programming from the courses we took last year, but it was not sufficient for the scope of this project. To gain insights into how to proceed with this aspect, we asked help from people around us who had expertise in web development. After consulting with them, we decided that using React.js for the frontend and Flask for the backend would be sufficient for our needs. You can find the details of our decision-making process and the reasons behind them in the “Methodology” section.

During our research, we learned that we needed a model to detect and extract faces from the photos before the facial recognition stage. After conducting tests, we concluded that YOLOv8[4] is the best option for this task.

The project begins with the collection of two main inputs. Firstly, we gather photographs from our friends and some celebrities, which are utilized to train our distance metric of the artificial neural network model and to create our database of the students. Secondly, snapshots are taken during class, from which faces are detected and identified using our trained model. The output of the system includes an updated attendance list and labeled snapshots with tagged faces. It is important to state that in our case we have only one photograph of each person. It simulates the photographs that are taken from BİLSİS system for each student. The size of our dataset is limited by the voluntary participation of our friends and some celebrities that we collect on the internet.

Based on our research, we have outlined the roadmap for solving our problem. In summary, our solution involves capturing individual photos of people simulating students in classrooms and saving them. We then pass these photos, which include their faces, through a pretrained Siamese model (during the testing phase, we used three different models separately). We store the embedded layers obtained from these models in our database. Once this stage is completed, we move on to the testing phase with group photos. We select one of the group photos we have previously collected, which will simulate a photo taken in the classroom, and feed it into YOLOv8. Using YOLO, we detect the facial images and remove them from the photo. After adjusting the size of the photos, we feed these images to our identical Siamese Network models and collect the embedded layers. We compare these embeddings with those in our database and We consider the person with the lowest distance value below a certain threshold as a match. We return these results to the user through our web application, representing the list of people present in the classroom.

Methodology (/ 25 Points)

Explain the methodology you followed throughout the project in technical terms including datasets, data pre-processing and featurization (if relevant), computational models/algorithms you used or developed, system training/testing (if relevant), principles of model evaluation (not the results). Using equations, flow charts, etc. are encouraged. Use sub-headings for each topic. Please use a scientific language. You may borrow parts from your previous reports but update them with the information you obtained during the course of the project. This section should be between 1000-1500 words (add pages if necessary).

In this section, we will provide detailed explanations of the methods mentioned in the previous section “Introduction, Problem Definition & Literature Review”. We will discuss the selected methods and how they work.

After the research phase, we decided our solution path. We did all of our project of this road map. Firstly, we elaborate on the face detection and recognition methods used in our project. For face detection, we employed the **YOLOv8** model, which is known for its efficiency and accuracy in object detection tasks. YOLOv8 utilizes a single neural network to detect faces in images by dividing the image into a grid and predicting bounding boxes and class probabilities for each grid cell.

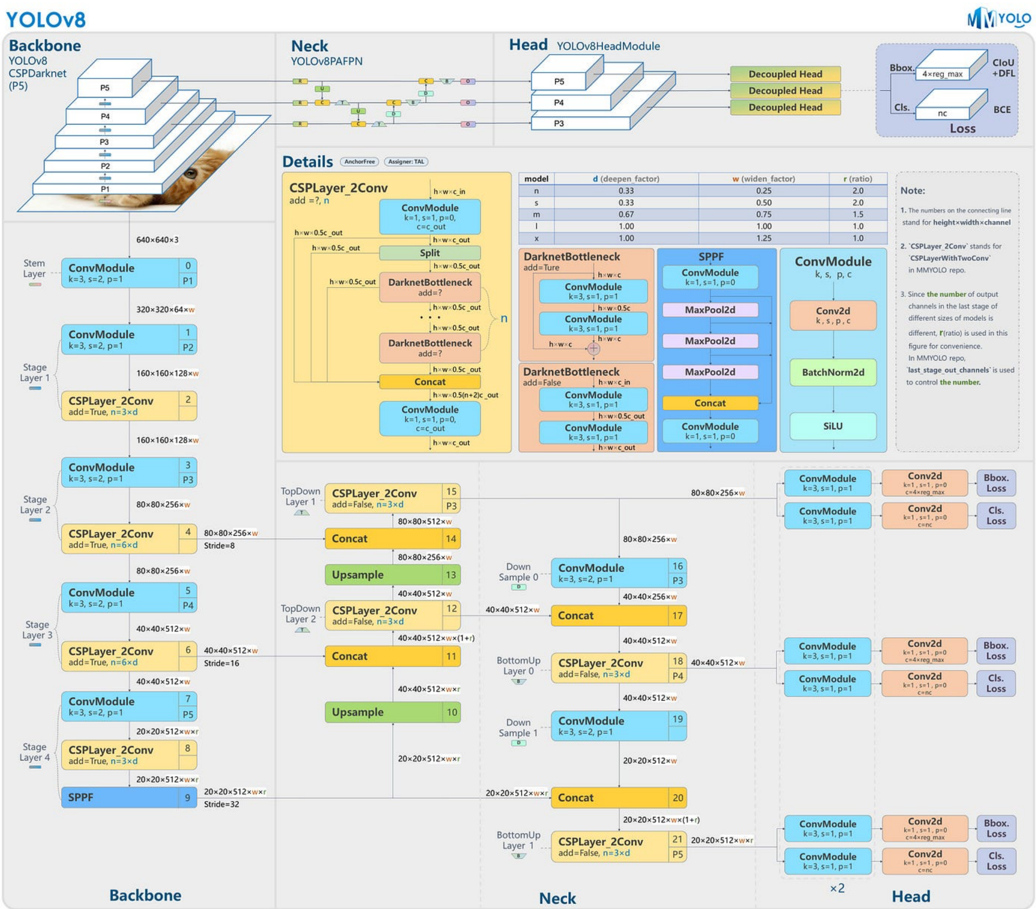


Figure: YOLOv8 Architecture

Using YOLOv8, we extracted faces of the persons in the photos. We used face detector part in person photos for making our database and the group photos to extract faces. Using the face detector, we extracted the faces from the individual photos. We then fed these extracted faces into the designated Siamese Networks and obtained the results. This ensured that the model's outputs were solely based on the features of the individuals' faces and were not influenced by the background of the photograph. By saving the embedded layers corresponding to each person who take the course, we created our database. Further details about the models will be provided later.

To create the dataset, we initially took photos of 10 of our friends. However, realizing that this number would not be sufficient, we followed our advisor's suggestion and expanded our dataset by including photos of celebrities and additional friends. This significantly increased the amount of data we had for testing our project. In total, we worked with and tested our system on the photos of nearly 60 different individuals. For the face recognition part, we conducted tests using group photos with various resolutions, lighting conditions, distances, and similar-looking faces.

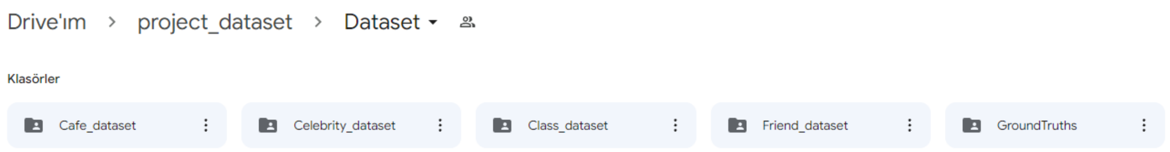


Figure: Dataset

In the case of group photos, we used the face detector to detect the faces. Using the coordinates of the detected faces in the photos, we separated them from the rest of the image. These face images were then passed through the pre-defined Siamese models to extract their embedded layers. Finally, we compared the embedded layers obtained from the group photos with the embedded layers in our database using a defined distance metric, allowing us to identify the individuals.

Once the faces are detected, we move on to the face recognition stage. For this, we adopted the Siamese Network architecture. Siamese Networks are neural networks that use a pair of identical subnetworks to process two inputs and learn to compare and measure their similarity. In our case, the inputs are the reference face images and the detected faces from the classroom photos. The Siamese Network models learn to generate embedded representations (embedded layers) for each input face, capturing the unique features and characteristics of individuals.

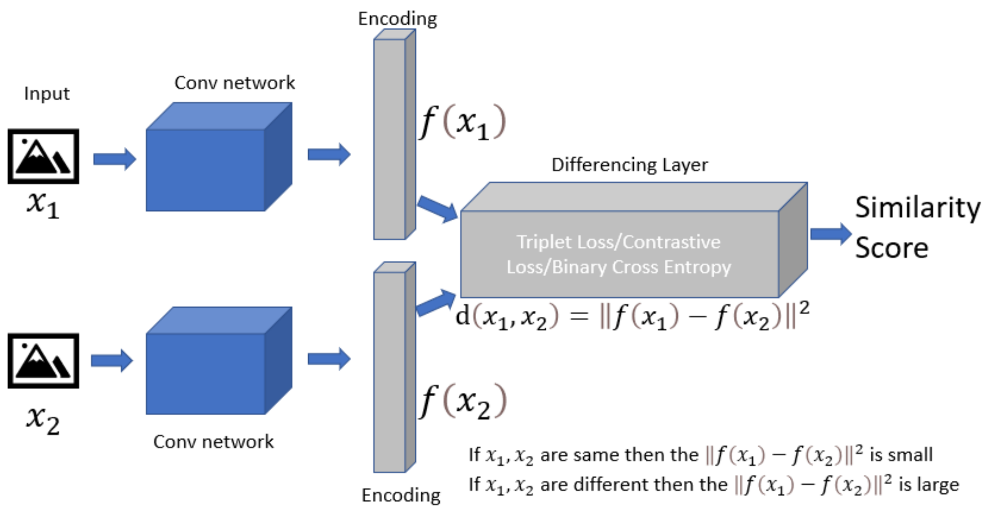


Figure: Siamese Network Architecture

To test the Siamese Network models, The collected images were used to test the network and learn the facial features specific to each person. During the testing phase, we evaluated the performance of three different Siamese Network models, namely ResNet50, SENet50, and VGG16, to determine which one yielded the best results. We found these models for our research phase and we did not choose one of them in the first place. So, we decided which network we will use in the final web application after the tests.

Although we initially attempted to train the base models of the Siamese network using our collected dataset, we obtained results well below our expected performance. During the training phase, our accuracy was around 10%, which was not suitable for our project. After explaining this situation to our advisor, they advised us to explore different models and informed us that we could find pre-trained model weights from model zoos[11] specifically trained for extracting facial features. We were able to find pre-trained model weights for **ResNet50, SeNet50,** and **VGG16** models, and continued our project using these models. For these weights, we use the VGGFace[5] library. It utilizes Keras and TensorFlow libraries and has weights of especially these three model face features.

```
from keras_vggface.vggface import VGGFace

# Based on VGG16 architecture -> old paper(2015)
vggface = VGGFace(model='vgg16') # or VGGFace() as default

# Based on RESNET50 architecture -> new paper(2017)
vggface = VGGFace(model='resnet50')

# Based on SENET50 architecture -> new paper(2017)
vggface = VGGFace(model='senet50')
```

Figure: VGGFace Library

Once the face detection and recognition components were ready, the next step was to select the appropriate model for our problem. To do this, we processed the photos of individuals in our dataset one by one using YOLO to extract only the facial regions. We then fed these facial images to each of the three models in sequence to extract their embedded layers, which were saved for later use.

Next, we applied YOLOv8 to the group photos to extract the faces of individuals. These extracted face images were then passed through each of the three models for testing purposes. For each embedded layer produced, we compared it with the embedded layers in our dataset using **cosine distance** as the chosen distance metric.

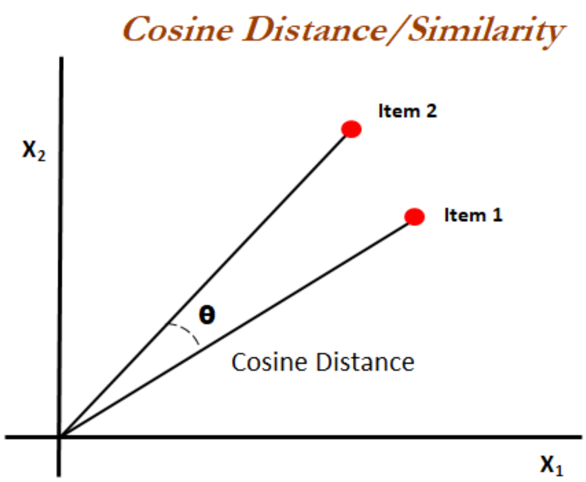


Figure: Cosine Distance

Cosine distance is commonly used in similarity comparison tasks, such as face recognition, because it measures the angle between two vectors rather than their magnitudes. This property makes it robust to variations in feature scales. Cosine distance is particularly useful when dealing with high-dimensional feature spaces as it provides a more meaningful measure of similarity. Another advantage of cosine distance is that it is computationally efficient, making it suitable for real-time applications where fast comparisons are required. Overall, by employing cosine distance, we can effectively compare embedded layers and identify similarities between faces, leading to accurate matching and recognition results.

After conducting our tests, we concluded that **SENet50** provided the best results for our problem. You can review our findings in the next section. Following this decision, we began working on the web application.

When we started the project, we initially planned to use React for the frontend, along with Spring and MySQL for the backend and database, respectively. We chose these languages and frameworks because we already had experience with them. We proceeded with our initial decision for the frontend, but made some changes for the backend and database. For the backend, we used Flask, as it is easier to use and optimized to work with our model written in Python. With Flask, we were able to execute our model function by function, avoiding the need to run the entire model from scratch for each operation and saving time. Additionally, we initially considered using a microservices architecture for the backend, but we realized that we could continue the project with simpler systems and that microservices were not necessary for our specific needs. In the future, if the project is used on a larger scale, such infrastructure changes may be required, but they are not essential within the scope of our graduation project.

Figure: Flask

Figure: React.js

During the project process, we also realized that we didn't need to use a database. As a result, we decided to store student, course, and attendance information in JSON files, which helped simplify our workflow. This approach reduced the complexity of managing a database system and made the project lighter in terms of resource usage.

Results & Discussion (/ 30 Points)

Explain your results in detail including system/model train/validation/optimization analysis, performance evaluation and comparison with the state-of-the-art (if relevant), ablation study (if relevant), a use-case analysis or the demo of the product (if relevant), and additional points related to your project. Also include the discussion of each piece of result (i.e., what would be the reason behind obtaining this outcome, what is the meaning of this result, etc.). Include figures and tables to summarize quantitative results. Use sub-headings for each topic. This section should be between 1000-2000 words (add pages if necessary).

First of all, we need to focus on the face detection part, and we decided to use yoloV5. We trained yoloV5 with the https://universe.roboflow.com/mohamed-traore-2ekkp/face-detection-mik1i dataset. But the results are not as good as we expected. We tested it especially in pictures with a large number of people and because of the dataset or like low image quality our model was able to detect 25 out of 125 people and the found ones' bounding boxes are not only covering the people's faces.

For the image above, accuracy is 0.20, precision is 1.00, recall is 0.20 and F1 score is 0.33.

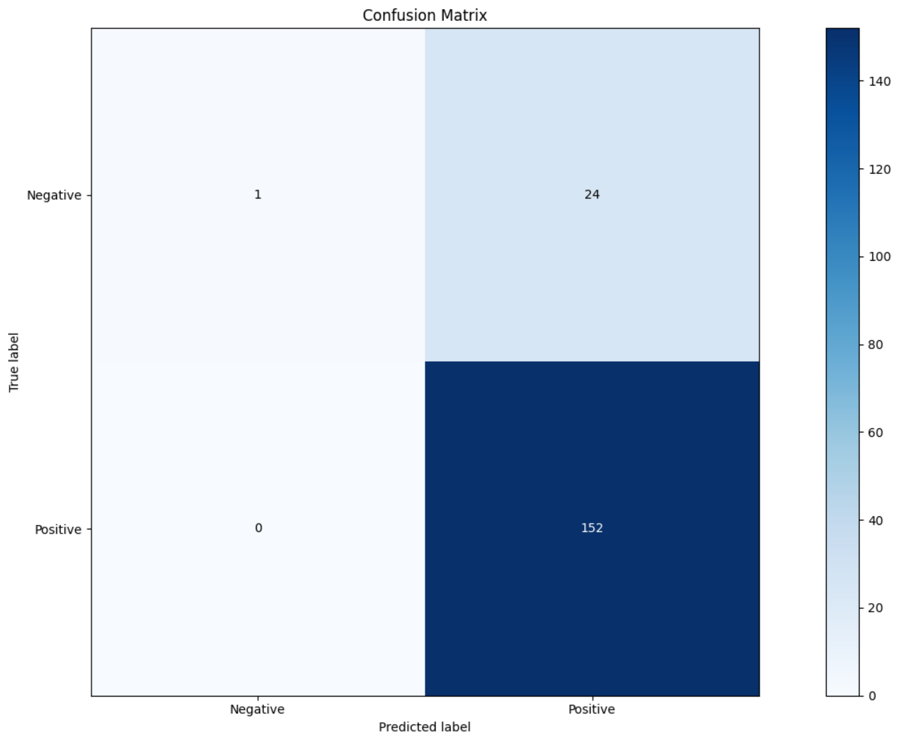
The reasons why the accuracy is low and the model does not work well may be the poor quality of the photo and the people too far away. Also, brightness is changing in the image and the dataset that we trained does not contain images like that or the dataset may not contain photos that can detect faces that are far away.

Also, we tried the model on some class photos. Our YOLOV5 model worked well but their confidentiality was not good .Model may find faces with .25 confidentiality and also it can detect even if there is no face. As a result of that, we can say that our YOLOV5 model is not working well.

After some research we found the last version of yolo which is YOLOV8 and a new dataset to train which is WIDER Face(http://shuoyang1213.me/WIDERFACE/). We tested the same photos with the YOLOV8 model. The results were much better than the YOLOV5. The V8 model was able to detect 93 out of 125 people and the bounding boxes were only capturing face.

As a result, we can say that the dataset contains faces that far away. But there are some false negatives (24). As I mentioned before the brightness change affects the model accuracy. There are people hiding from the camera in the photos and some people were wearing masks. These reasons affect the model accuracy and make it harder to find faces.

Here is the confusion matrix of YOLOV8.



For the identification part, we used pretrained models(SENet50, RESNet50 and VGG16) and extracted the embeddings of faces which are coming from YOLOV8. Then by calculating and finding the cosine distance deciding if there is a match or not. In our dataset for identification, there are 79 people (43 celebrities, 36 our friends). We saved the embedding of the person and their facial extractions in a folder. After that we started to upload photos that include some of the people in our dataset.

As we mentioned before, there are three models for extracting embeddings of faces. Here is some results:

SENet50

ResNet50

VGG16

Ground Truths

While we are calculating cosine distance and deciding if there is a match or not, we need the set threshold. Because the dataset does not contain all the people in this photo. We set the cosine similarity threshold to 0.35 and if the cosine similarity distance is lower than the .35 we model count as a match and select the lowest cosine distance of the all cosine distance for a person. We were expecting to find all the people in the dataset.

In these three models, SENet50 is the most accurate one. Resnet50 cannot find three of the people who are in our dataset. SENet50 cannot find one person in the dataset and find one person that is not in our dataset and who is not actually Samet but it is tagged as Samet. VGG16 is the worst model of these three and five people are tagged with the wrong name and it cannot find two people that are in our dataset.

For the mismatch of “Can Yavuztekin” the source photo may not be biometric and his face not clearly detected. For Senet50’s true negative(TN) who is tagged as “Samet” but he is not in our dataset. The reason for this TN is the person is not looking at the camera and his face is not clearly visible and because of that model predicted wrongly. ResNet50 may predict correctly but they cannot pass the threshold value not tagged correctly. VGG16 is not working correctly and finds many uncorrect faces and counts as a match. It is not appropriate for our problem.

After testing all the photos in the dataset that include our friends.

SENet50 has 20 true positives, 16 true negatives , 2 false positives and 1 false negative.

ResNet50 has 16 true positives, 16 true negatives , 3 false positives and 0 false negatives.

We tried our models on some celebrity images. Results are given below:

SENet50 **ResNet50**

VGG16

All of the people in that photo are also in our dataset but none of the models find all of them. But still as we can see the SENet50 is the most accurate one because tagged ones are true positives. But the ResNet50 and VGG16 tagged with wrong names/ids and in our attendance system we do not want to make any mistake while taking attendance that may cause problems.

The reason for the false negative is the test image quality is low and also people's facial image quality is low. May the people be changed and because of the getting older their face changed a bit and the model could not identify them. Also, the model may identify them but they could not pass the threshold value and because of them not tagged.

To sum up, by comparing all models for the face detection and also identification part we can say that the YOLOV8 is the most accurate model for face detection even in extreme conditions and the SENet50 is the most accurate model for the face identification part.

The Impact and Future Directions (/ 15 Points)

Explain the potential (or current if exist) impacts of your outcome in terms of how the methods and results will be used in real life, how it will change an existing process, or where it will be published, etc. Also, explain what would be the next step if the project is continued in the future, what kind of qualitative and/or quantitative updates can be made, shorty, where this project can go from here? This section should be between 250-500 words.

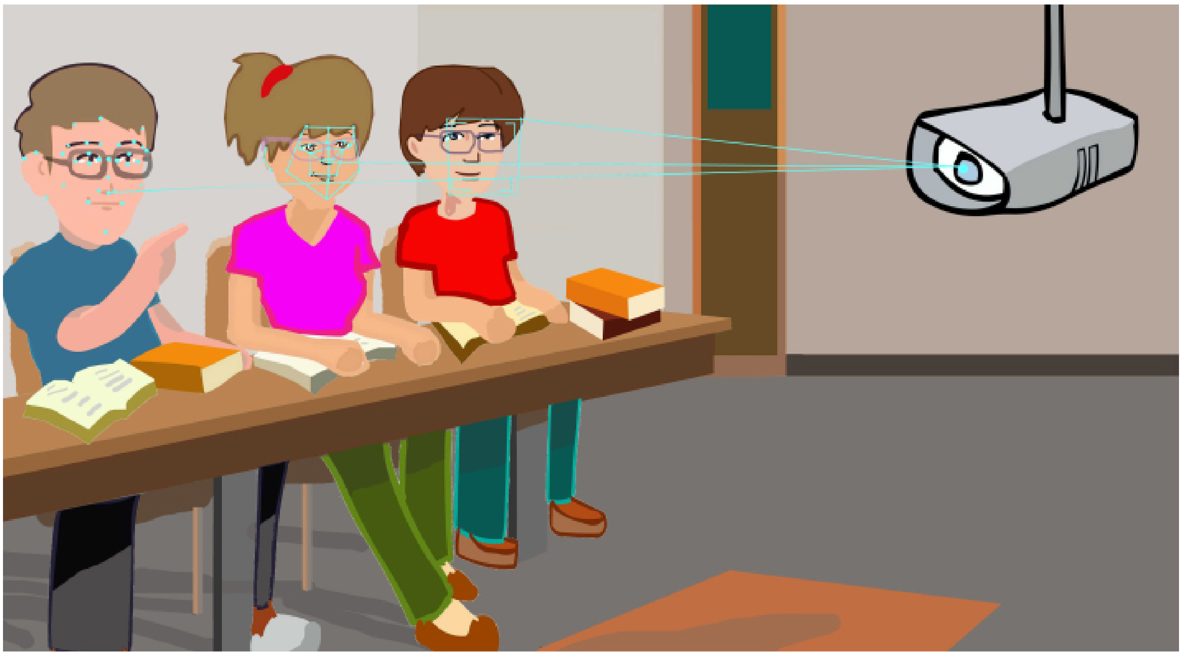
The outcome of our graduation project, an AI-based attendance system using face recognition, holds great potential for real-life applications and has the ability to transform the existing attendance-taking process in educational institutions. By automating this process, our system brings forth numerous benefits. Firstly, it saves time and improves efficiency by eliminating the need for manual attendance sheets and reducing the effort required by teachers to count and record attendance. This allows instructors to devote more time to teaching. Secondly, our system enhances accuracy by leveraging artificial intelligence and machine learning algorithms for precise face detection and recognition, minimizing errors and mitigating fraudulent practices like proxy attendance. Thirdly, it offers convenience to students, as they no longer need to physically sign attendance sheets, resulting in fewer disruptions during lectures. The attendance can be captured discreetly and seamlessly, enhancing the overall learning experience. Lastly, the system improves data management by storing attendance records in a centralized database, simplifying administrative tasks such as generating reports and calculating attendance percentages.

Moving forward, there are several potential qualitative and quantitative updates that can be considered for the project. Expanding the dataset is one such avenue, as collecting a larger and more diverse set of student photos would enhance the system's accuracy and robustness. This could include images captured in various lighting conditions, angles, and with different appearances, such as students wearing masks. Additionally, integrating the attendance system with existing student management systems, such as BİLSİS, would streamline the process further, allowing for automatic extraction of student photos or direct import of attendance records. Continuous model training with new data would contribute to adapting the system to changes in student appearance over time, ensuring reliable performance. Implementing real-time attendance tracking during lectures would provide instant feedback to instructors, enabling them to address any attendance-related issues promptly. Furthermore, the development of a mobile application for the attendance system would add flexibility and convenience, enabling users to access attendance records, receive notifications, and upload class photos directly from their mobile devices.

In terms of publication, the project's results and findings could be shared through academic conferences, journals, or technical reports. Disseminating this information would contribute to the academic community and facilitate knowledge exchange in the field of AI-based attendance systems.

The project has the potential to be used in all departments and universities, as well as by companies. The facial recognition and attendance tracking system can play a significant role not only in educational institutions but also in businesses. In the corporate world, monitoring employee attendance, implementing security protocols, and optimizing time management are crucial. This system can automate and record attendance for meetings, seminars, training sessions, and other events in the workplace. It can be a valuable tool for personnel management and performance evaluations.

Furthermore, the project has the potential for implementation in all universities. Attendance and class participation are important factors in universities, and manual tracking methods can be time-consuming and prone to errors. A facial recognition-based attendance system can be utilized in all university courses to accurately record student attendance. It provides instructors with better attendance analysis and allows for monitoring student performance.



Moreover, the project can be applied in the private sector as well. For instance, in large-scale events, conferences, or seminars, tracking and recording attendance is essential. Facial recognition technology facilitates attendance management in such events, enabling organizations to obtain accurate attendance data. Additionally, it can assist companies in meeting their security requirements, such as providing restricted access to specific areas or detecting unauthorized entries.

In conclusion, our graduation project has the potential to significantly impact the attendance-taking process in educational institutions. By harnessing AI technologies, we have developed a system that improves accuracy, efficiency, and convenience. The project's future trajectory includes expanding the dataset, integrating with existing systems, continuous model training, real-time tracking, and mobile application development. Sharing the project's results through various publication channels would further contribute to the advancement of attendance tracking systems in educational settings.

Reference:

[1] https://paperswithcode.com/method/resnet
[2] https://paperswithcode.com/method/senet
[3] https://paperswithcode.com/method/vgg-16
[4] https://docs.ultralytics.com/
[5] https://github.com/rcmalli/keras-vggface
[6] https://www.thalesgroup.com/en/markets/digital-identity-and-security/government/biometrics/facial-recognition
[7] https://www.eff.org/tr/pages/face-recognition
[8] https://paperswithcode.com/task/face-detection
[9] https://www.techtarget.com/searchenterpriseai/definition/face-detection
[10] https://paperswithcode.com/method/siamese-network
[11] https://modelzoo.co/