Using Real Time Computer Vision Algorithms in Automatic Attendance Management Systems

Visar Shehu¹, Agni Dika²

Contemporary Sciences and Technologies - South East European University, Macedonia¹
Contemporary Sciences and Technologies - South East European University, Macedonia²
v.shehu@seeu.edu.mk¹, a.dika@seeu.edu.mk²

Abstract. This paper introduces a new approach in automatic attendance management systems, extended with computer vision algorithms. We propose using real time face detection algorithms integrated on an existing Learning Management System (LMS), which automatically detects and registers students attending on a lecture. The system represents a supplemental tool for instructors, combining algorithms used in machine learning with adaptive methods used to track facial changes during a longer period of time. This new system aims to be less time consuming than traditional methods, at the same time being nonintrusive and not interfere with the regular teaching process. The tool promises to offer accurate results and a more detailed reporting system which shows student activity and attendance in a classroom.

Keywords. computer vision, object tracking, face recognition, machine learning, teaching

1. Introduction

A key factor of improving the quality of education is having students attend classes regularly. Traditionally students are stimulated to attend classes using attendance points which at the end of a semester constitute a part of a students final grade. However, traditionally this presents additional effort from the teacher, who must make sure to correctly mark attending students, which at the same time wastes a considerable amount of time from the teaching process. Furthermore it can get much more complicated if one has to deal with large groups of students. This paper introduces a new automatic attendance management marking system, without any interference with the regular teaching process. The system can be used also during exam sessions or other teaching activities where attendance is obligatory. This system eliminates classical student identification such as calling student names, or checking respective identification cards, which can not only interfere with the teaching process, but also can be stressful for students during exam sessions.

2. Related work

Most modern learning management systems, implement some type of attendance management. Moodle [9] automates the process by using RFID or barcode scanners. Classrooms are equiped with a barcode / RFID scanner which scans and enrolls students that enter the classroom. Other LMS systems such as Angel [3] require students to login in to a web page with a special one time temporary key in order to mark their presence on class. The problem with these approaches is that they interfere with the regular teaching process.

Using face recognition in time attendance management systems in not new. There are few commercial solutions available to companies, that implement face recognition in work environments. Kawaguchi [11] proposes face recognition in attendance management systems. Related systems that use biometrics (fingerprint recognition, iris recognition etc) to identify users are time management systems used in many institutions. However, installing these systems in every classroom in a university would pose a bigger financial burden. It would also require from the university to record biometric information from all students, which would introduce further privacy concerns. These systems are also subject to physical damage from their users. Therefore they need additional maintenance costs. The idea proposed by us, removes physical access from anyone to the system.

3. General idea

Proceedings of the ITI 2010 32nd Int. Conf. on Information Technology Interfaces, June 21-24, 2010, Cavtat, Croatia
This proposed system introduces a new automatic attendance marking system, which integrates computer vision and face recognition algorithms into the process of attendance management. The system is implemented using a non-intrusive digital camera installed on a classroom, which scans the room, detects and extracts all faces from the acquired images. After faces have been extracted, they are compared with an existing database of student images and upon successful recognition a student attendance list is generated and saved on a database. This paper addresses problems such as real time face detection on environments with multiple objects, face recognition algorithms as well as social and pedagogical issues with the applied techniques.

During the practical realization of this idea, it was decided to implement machine learning algorithms, which are commonly used in the field of computer vision. The first problem was to detect face-like patterns in real-time images. In order to achieve that, we used the HAAR classifiers [10], which were additionally trained with positive images acquired from classrooms full of students [8]. However, detecting face-like patterns constitutes just a part of the problem. One must implement a face recognition algorithm which is used to positively identify a student from a database of students.

4. System architecture

The system designed is part of an in-house built learning management suite (Libri) [5]. It is constructed in many modules:
- Image capturing,
- Face Detector and
- Face recognizer.

The entire process is described in the pseudocode shown in Fig 1:

```plaintext
scan room with digital camera
for each detected object
    transfer the object in server
next object

for each face in server
    if face in students database
        record student as enrolled
        save face in students database
    else
        mark face as unrecognized
next face
```

Figure 1. System architecture pseudocode

The required infrastructure in classroom is a rotating camera positioned centrally in the front of the classroom (Fig. 2).

Using this setup, the camera is capable to capture frontal images from students such as the one in Fig 3.

Figure 3. A classroom with students

A different approach would be to use a camera at the entrance of the classroom, which would individually detect faces for everyone entering the classroom. This way, the face detector would have much less work to do, but there would be only one chance to capture a good frame. A frontal camera in the middle of the classroom can take as many pictures as necessary.

Physically the system is integrated on the existing South East European Universities infrastructure. To function, the system requires each classroom to have at least one internet connected computer. This computer communicates with the LMS server, where captured images are transferred. Fig 4 depicts the physical architecture of this system.
4.1. Image capturing

Images are captured using a module built in the LMS. The module is an application written in Windows Presentation Foundation (WPF) framework. After an image is captured, using a web services transfers the image on server for processing. Together with the image, the web service accepts the course code. Using this course code, the LMS is aware of which students are enrolled in that class and do face matching only for those students.

The camera continuously takes pictures on a given interval (by default each five minutes), until all faces detected are successfully identified or until the system is told to stop. This means that in some cases, e.g., when a face cannot be successfully identified, the camera keeps taking pictures until the class finishes.

4.2. Face detection

Because of processor intensive job of the face detection algorithm, this tool is server based. Detecting a face is in essence an object detection task, where the object of interest in this case is the face. However, many factors can interfere with the face detection algorithms, factors such as face pose, scale, position, rotation, light, image colors etc. The same problems arise when one wants to identify (recognize) a face, with addition to some other obstacles which is discussed shortly.

The process of detecting faces from still pictures containing multiple faces can be separated in few steps. There are plenty face detection algorithms which can effectively detect a face (or any other specific object) in a picture. In the system presented here, most students face the camera frontally hence we chose to use the HAAR classifier for face detection. This classifier is implemented on Intel’s Open CV library.

The classifier works by training a model using positive face images and negative face images. A positive image is an image that contains the desired object to be detected, in our case this object is a face. A negative image is an image that does not contain the desired object. After the model is trained, it is able to identify face features, which is later stored on a XML file.

A problem faced during this process was the large number of false-positives: objects mistakenly detected as faces. This was not such a big issue for us, since a false-positive does not result in a positive identification during the recognition phase. Because of this, we lowered the detection threshold, so all faces could be detected.

After a face has been detected, the rectangle enclosing this face is cropped and processed later by the face recognition module. This rectangle represents a single face, and after being cropped as an image is transferred on server. Each file transferred is renamed to have a unique ID (GUID).

Fig 5, shows the face detector in action where white rectangles show detected faces. Some of the extracted faces are shown in Fig 6.

4.3. Face recognition

Recognizing a face means to identify that particular face from a list of faces on a database. Our university, upon enrollment takes pictures
from every student, and those images are stored in a database.

Same as in face detection, there are many existing algorithms used to identify a face. Our system implements a server based module, programmed in Python (Pyfaces - http://pyfaces.blogspot.com/) which takes benefit of eigenfaces [6] to identify a face. This algorithm has many drawbacks: it depends on scale, pose and the color of the compared images. However the algorithm is very fast, and can compare only to images, thus we do not need to have multiple images of a person to train our system.

Since our system is setup to capture only frontal images the pose of the face in not an issue. When a face is captured during the face detection phase, it is converted into gray scale. The same conversion is applied to faces on our student image database. We also do background subtraction on our images so other objects do not interfere during the process.

Another issue is that faces are subject of change during time (facial hair, eyeglasses etc). Whenever we successfully identify a face, a copy of that face is stored in the database of faces for that student. Together with the image we store the time and date when this image was taken.

This way even if a student gradually changes his appearance (e.g., grows a beard) the system is still capable to identify him, since it has multiple images of the same person. On each consequent scan for a student, the recognition module starts comparing images from this database, sorted by date in descending order. This approach was chosen since the latest image of a student on our database is most likely to be more similar to the current captured image.

Of course, a drastic change on a student’s look causes the system to not identify that particular student. To solve this issue, we have included a module, which lists all unidentified faces and the teacher is able to manually connect a captured face with a student from the list. This image is also stored on our database, as an updated picture of this particular student. This manual recognition process is performed only once. In a subsequent scan, this student is identified automatically by our system.

To speed up the face recognition process we only compare images captured in a classroom, with the database of students enrolled for that course only. This ensures that we process only a small subset of images available on our server.

5. Experiment

We applied our system in three courses with a total of ten groups. There were 147 students enrolled in those courses. From all of them we asked for consent and they were willing to join this experiment.

We gained the following statistics:

- From each captured image, there were approximately 70% of successfully detected faces.
- From them only about 30% were successfully identified.
- When the experiment was conducted only with first year students, the successful identification rate was much higher (about 56%).

The main reason behind these disappointing results are old images of students stored when they first enroll in the university. For first year students, these images are still fresh, thus the results are more promising. However, second year and third year student facial appearance has changed since they registered; resulting with lower recognition rate.

6. Privacy concerns

As mentioned previously, we store image data on server. This approach can always raise privacy concerns, regarding the safety of those images and access level to that server. Whenever one has to deal with this type of sensitive data, he must always offer means to protect data from unauthorized access. [4] In some countries, especially in EU, these privacy concerns are considered very seriously with regulations in place in their corresponding legislatures. [1]

Since this is system is mainly done to explore the advantages and the feasibility of using face recognition in attendance management, it still does not offer any privacy protection.

However, if the system proves to be usable one has to implement some type of secure means to ensure privacy. In password security one of the most popular approaches in this case is to encrypt data using a one way function. For us this means to encrypt images gotten from the camera and compare those images with existing (also encrypted) images on server. However, since biometric data is noisy, this approach is not feasible. Another approach can be to use one way encryption using a private / public key
approach. This way only the person in possession of the correct key is able to decrypt the data.

7. Conclusion

An automatic attendance management system is a necessary tool for any LMS. Most of the existing systems are time consuming and require for a semi manual work from the teacher or students.

Our approach aims to solve the issues by integrating face recognition in the process. Even though this system still lacks the ability to identify each student present on class, there is still much more room for improvement. Since we implement a modular approach we can improve different modules until we reach an acceptable detection and identification rate. Another issue that has to be taken in consideration in the future is a method to ensure users privacy. Whenever an image is stored on our servers, it must be impossible for a person to use that image.

8. Acknowledgement

The authors would like to thank students from the Computer Science and Business Informatics departments in South East European University for their contribution in this project.

9. References

[9] Sourish Behera, Rajesh Kumar Kushwaha. RFID Based People Management Using UHF Tags