Automated Team Attendance

DESIGN DOCUMENT

19
Mohamed Youseff Selim
Mohamed Youseff Selim
Lance Demers – Chief Engineer Hardware
Brandon Johnson – Team Leader
Nathan Oran – Documentation Manager
Angela Schauer – Meeting Facilitator and Chief Engineer Software
Connor Sullivan – Testing Lead
sddec20-19@iastate.edu
https://sddec20-19.sd.ece.iastate.edu

Revised: 2/23/20 V1

Executive Summary

Development Standards & Practices Used

This project will be developed using test driven development and agile workflow. We will use paired programming and code reviews to make sure our software is high quality. We hope to develop a modular software system to simply future development and hopefully save ourselves time if we need to reimplement specific parts of the code.

Summary of Requirements

- Hardware
    - Project must run on a Raspberry Pi
    - Camera must be able to take clear pictures
    - Must be mounted in a way for the camera to see all of the classroom
- Software
    - Control the camera to take images of the classroom
    - Accurate detection of students from taken images
    - Determine which groups have absent members
    - Consolidate data and send a report to the professor
    - Professor must be able to create and adjust multiple seating charts for the system to work with

Applicable Courses from Iowa State University Curriculum

- CPR/S E 185/E E 285/COM S 227 - Introduction to programming
- COM S 228 - Data Structures
- COM S 309 - Application Development
- CPR E 288 - Embedded Systems

New Skills/Knowledge acquired that was not taught in courses

For several members of our team, the python programming language will be a new skill. This project also requires Machine Learning using Yolo and openCV which are new skills for all of us.

Table of Contents

List of figures/tables/symbols/definitions (This should be the similar to the project plan)

1 Introduction

1.1 ACKNOWLEDGEMENT

We acknowledge Mohamed Youseff Selim for coming up with the idea of the project and also guiding our team through it.

1.2 PROBLEM AND PROJECT STATEMENT

No doubt, Team-Based Learning (TBL) positively influenced attendance in the classroom. However, teams are still suffering from absenteeism. Although a handful of attendance tools exist, none of these tools is adapted to the team-based classes, besides, none of them records the attendance without any interaction from the instructor/student. Moreover, these tools are consuming from 2 to 3 minutes from the class time, which will sum up over the semester to 135 minutes in the worst case. The proposed tool will record attendance at zero time.

1.3 OPERATIONAL ENVIRONMENT

The Automated Team Attendance Tool will operate in a classroom environment. The final product will be mounted in classrooms used by Instructors. The product should be able to function in small and large lecture halls.

1.4 REQUIREMENTS

Data Collection:

Image collection of the classroom, over multiple instances during class

Detect Students:

Optimize the accuracy of student detection using deep learning analysis

Determine Which Groups Have Absent Members:

Compare the locations of students to a provided team-based seating chart

Send Report to Professor:

Consolidate the collected data into an email sent to the professor of the class

Adjustable Seating Arrangements:

Professor has access to create and update team seating charts

1.5 INTENDED USERS AND USES

The product is intended to be used by any instructors or teaching assistants wishing to record attendance automatically.

1.6 ASSUMPTIONS AND LIMITATIONS

Assumptions

- Instructor or TA will provide a seating chart
- Teams will be in the same spot for the whole semester
- Instructor or TA will decide when to capture the attendance
- Product will be used in a classroom environment

Limitations

- Algorithm struggles to detect objects that are densely packed
- Test to see how program performs on models of the raspberry pi that have lower performance specs

1.7 EXPECTED END PRODUCT AND DELIVERABLES

May 2020: Test product having minimum functionality. Such as determining the number of students present for each team. Being able to take a seating chart as input to be used to determine any missing students.

December 2020: A final product that will capture the number of students present for each team based on a seating chart provided by the instructor. Once the program finishes it will email the instructor the number of students present for each team.

2. Specifications and Analysis

2.1 PROPOSED APPROACH

The proposed approach to solve the problem of lengthy attendance times in TBL classrooms is using object detection to take attendance. A camera will take a picture of the classroom and verify it against a seating chart provided by the professor. An email with information about the attendance of each group will then be sent to the professor. As of now, we have been researching and testing various free, open source object detection software on pictures of lecture halls to see if they are a viable option. For this project, no specific standards need to be followed.

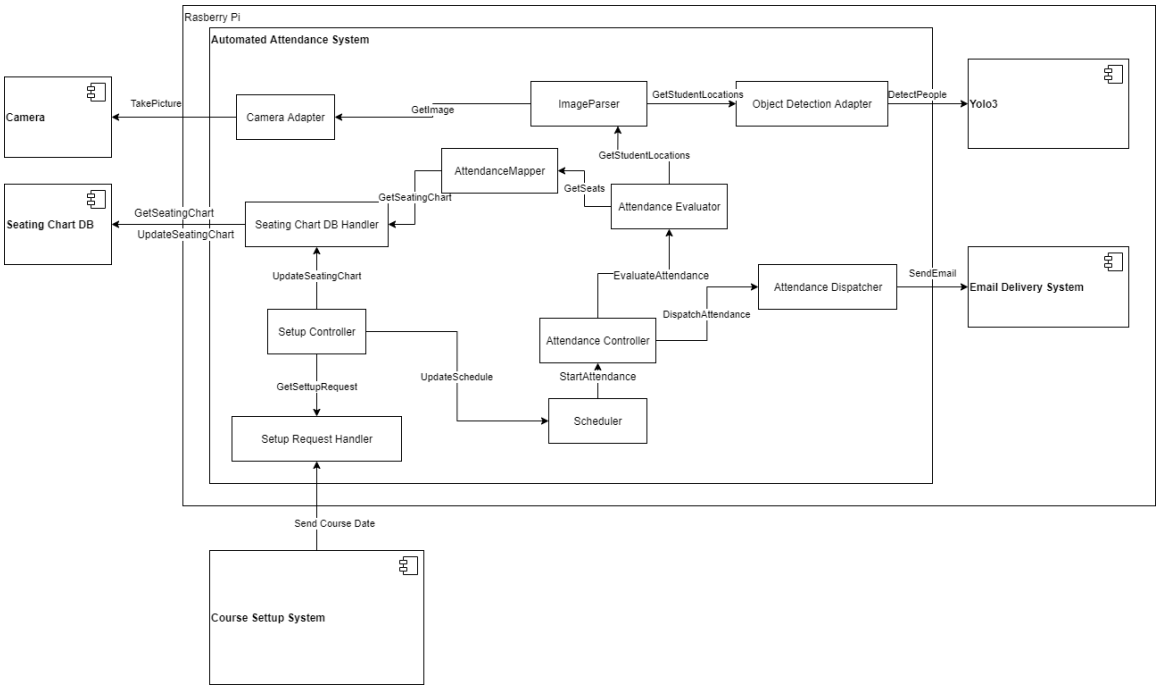
2.2 DESIGN ANALYSIS

After researching the available object detection software, we found that this proposed approach is feasible, specifically using the software YOLOv3. YOLO builds on top of OpenCV to create a system that can quickly and accurately detect and identify objects in a picture. To take the picture, we will have a camera connected to a RaspberryPi which will be connected to a server where it will run our code. We will be using Python for all the code relating to the object detection. For our project, we will take the data of the x,y coordinates of students that are detected and parse them so they are ready to be compared against the seating chart provided. To get the data of the seating chart, we will need a UI for the professor to use and analyze the data they give us.

2.3 DEVELOPMENT PROCESS

We will be using a mix of Test Driven Development and Agile processes. We want to be able to create code incrementally and includes a lot of user input, so we decided Agile was best. For each iteration, we want to use TDD to make sure all requirements we want are being met and our code does as we want.

2.4 CONCEPTUAL SKETCH


Rasberry Pi
Automated Attendance System
Camera
TakePicture
Camera Adapter
GetImage
ImageParser
GetStudentLocations
Object Detection Adapter
DetectPeople
Yolo3
AttendanceMapper
GetStudentLocations
GetSeats
Attendance Evaluator
Seating Chart DB
GetSeatingChart
UpdateSeatingChart
Seating Chart DB Handler
GetSeatingChart
UpdateSeatingChart
EvaluateAttendance
Attendance Dispatcher
SendEmail
Email Delivery System
DispatchAttendance
Setup Controller
Attendance Controller
GetSettupRequest
UpdateSchedule
StartAttendance
Setup Request Handler
Scheduler
Send Course Date
Course Settup System

3. Statement of Work

3.1 PREVIOUS WORK AND LITERATURE

Include relevant background/literature review for the project

– If similar products exist in the market, describe what has already been done

– If you are following previous work, cite that and discuss the **advantages/shortcomings**

– Note that while you are not expected to “compete” with other existing products / research groups, you should be able to differentiate your project from what is available

Detail any similar products or research done on this topic previously. Please cite your sources and include them in your references. All figures must be captioned and referenced in your text.

3.2 TECHNOLOGY CONSIDERATIONS

Highlight the strengths, weakness, and trade offs made in technology available.

Discuss possible solutions and design alternatives

3.3 TASK DECOMPOSITION

In order to solve the problem at hand, it helps to decompose it into multiple tasks and to understand interdependence among tasks.

3.4 POSSIBLE RISKS AND RISK MANAGEMENT

Include any concerns or details that may slow or hinder your plan as it is now. These may include anything to do with costs, materials, equipment, knowledge of area, accuracy issues, etc.

3.5 PROJECT PROPOSED MILESTONES AND EVALUATION CRITERIA

What are some key milestones in your proposed project? Consider developing task-wise milestones. What tests will your group perform to confirm it works?

3.6 PROJECT TRACKING PROCEDURES

What will your group use to track progress throughout the course of this and next semester?

3.7 EXPECTED RESULTS AND VALIDATION

What is the desired outcome?

How will you confirm that your solutions work at a **High level**?

4. Project Timeline, Estimated Resources, and Challenges

4.1 PROJECT TIMELINE

• A realistic, well-planned schedule is an essential component of every well-planned project

• Most scheduling errors occur as the result of either not properly identifying all of the necessary activities (tasks and/or subtasks) or not properly estimating the amount of effort required to correctly complete the activity

• A detailed schedule is needed as a part of the plan:

– Start with a Gantt chart showing the tasks (that you developed in 3.3) and associated subtasks versus the proposed project calendar. The Gantt chart shall be referenced and summarized in the text.

– Annotate the Gantt chart with when each project deliverable will be delivered

• Completely compatible with an Agile development cycle if that's your thing

How would you plan for the project to be completed in two semesters? Represent with appropriate charts and tables or other means.

Make sure to include at least a couple paragraphs discussing the timeline and why it is being proposed. Include details that distinguish between design details for present project version and later stages of project.

4.2 FEASIBILITY ASSESSMENT

Realistic projection of what the project will be. State foreseen challenges of the project.

4.3 PERSONNEL EFFORT REQUIREMENTS

Include a detailed estimate in the form of a table accompanied by a textual reference and explanation. This estimate shall be done on a task-by-task basis and should be based on the projected effort required to perform the task correctly and not just "X" hours per week for the number of weeks that the task is active

4.4 OTHER RESOURCE REQUIREMENTS

Identify the other resources aside from financial, such as parts and materials that are required to conduct the project.

4.5 FINANCIAL REQUIREMENTS

If relevant, include the total financial resources required to conduct the project.

5. Testing and Implementation

Testing is an **extremely** important component of most projects, whether it involves a circuit, a process, or a software library

Although the tooling is usually significantly different, the testing process is typically quite similar regardless of CprE, EE, or SE themed project:

1. Define the needed types of tests (unit testing for modules, integrity testing for interfaces, user-study for functional and non-functional requirements)
2. Define the individual items to be tested
3. Define, design, and develop the actual test cases
4. Determine the anticipated test results for each test case 5. Perform the actual tests
6. Evaluate the actual test results
7. Make the necessary changes to the product being tested 8. Perform any necessary retesting
9. Document the entire testing process and its results

Include Functional and Non-Functional Testing, Modeling and Simulations, challenges you've determined.

5.1 INTERFACE SPECIFICATIONS

– Discuss any hardware/software interfacing that you are working on for testing your project

5.2 HARDWARE AND SOFTWARE

– Indicate any hardware and/or software used in the testing phase

– Provide brief, simple introductions for each to explain the usefulness of each

5.3 FUNCTIONAL TESTING

Examples include unit, integration, system, acceptance testing

5.4 NON-FUNCTIONAL TESTING

Testing for performance, security, usability, compatibility

5.5 PROCESS

– Explain how each method indicated in Section 2 was tested

– Flow diagram of the process if applicable (should be for most projects)

5.6 RESULTS

– List and explain any and all results obtained so far during the testing phase

- – Include failures and successes
- – Explain what you learned and how you are planning to change it as you progress with your project
- – If you are including figures, please include captions and cite it in the text

  • This part will likely need to be refined in your 492 semester where the majority of the implementation and testing work will take place

-**Modeling and Simulation**: This could be logic analyzation, waveform outputs, block testing. 3D model renders, modeling graphs.

-List the **implementation Issues and Challenges**.

6. Closing Material

6.1 CONCLUSION

Summarize the work you have done so far. Briefly re-iterate your goals. Then, re-iterate the best plan of action (or solution) to achieving your goals and indicate why this surpasses all other possible solutions tested.

6.2 REFERENCES

This will likely be different than in project plan, since these will be technical references versus related work / market survey references. Do professional citation style(ex. IEEE).

6.3 APPENDICES

Any additional information that would be helpful to the evaluation of your design document.

If you have any large graphs, tables, or similar that does not directly pertain to the problem but helps support it, include that here. This would also be a good area to include hardware/software manuals used. May include CAD files, circuit schematics, layout etc. PCB testing issues etc. Software bugs etc.