IC Chipz

Design Document

May 2020 - Project 40

Advisor/Client

Dr. Henry Duwe

Team

Andrew Kicklighter - Mobile Developer

Alexander Weakland - Mobile Developer

Nicholas Dykhuizen - Integration Developer

Justin Elsbernd - Integration Developer

Joshua Heiser - Embedded Developer

Paul Kiel - Embedded Developer

Contact

sdmay20-40@iastate.edu

https://sdmay20-40.sd.ece.iastate.edu

Revised: 2020-4-26/V4

Executive Summary

Engineering Standards and Design Practices

- Nvidia Jetson board
- YOLO image recognition API
- E-Con Systems camera
- Agile development
- IEEE 1012-2016 - Standard for System, Software, and Hardware Verification and Validation
- IEEE 1220-2005 - Standard for Application and Management of the Systems Engineering Process
- IEEE 829 - Documentation Standards

Summary of Requirements

- Functional mobile application for monitoring video and scoring
- Computer vision program to process scoring
- Two reliable camera locations for video capture
- Low error rate
- Product functions at 95%+ success rate with night time conditions
- Automated data collection
- Automated integration into analysis systems

Applicable Courses from Iowa State University Curriculum

- SE 319
- COM S 309
- CPRE 288
- SE 329/339
- CPRE 489

New Skills/Knowledge acquired that was not taught in courses

- Computer vision programming
- Data pipelines and integration techniques

Table of Contents

List of Definitions

Xamarin: A C#, cross platform mobile development toolkit made by Microsoft for Visual Studios

XAML: C# UI templates similar to HTML/XML

YOLO (You Only Look Once): An Object detection algorithm that detects objects in real-time

Darknet: Convolutional Neural Network for training Object Detection Computer Vision Models (such as YOLO)

Data Pipeline: Organization and implementation of the way data flows through various modules

1 Introduction

1.1 Acknowledgement

The IC Chipz team wants to acknowledge our client for providing us with the necessary equipment to complete this project. This includes: the Nvidia Jetson board, Nvidia Jetson external hardware, wifi hardware, sd card, e-con camera, and past project progress from previous groups. The team has used the past project progress in conjunction with new technology. The team also would like to acknowledge our advisor, Dr. Henry Duwe for assisting the group throughout this project.

1.2 Problem and Project Statement

1.2.1 Problem Statement

Skeet shooting should be automatically judged and scored accurately for all tournaments and events. In today's world, there is becoming a shortage of reliable skeet shooting judges. If this problem continues to be ignored, the sport of skeet shooting will eventually be out of options for judges. To solve this issue, the team will use the implementation of an automatic scoring system to fairly judge skeet shooting events.

1.2.2 Project Statement

As the sport of skeet shooting ages, the ability to find trained referees for the sport becomes more difficult. With a mobile automated scoring system the team plans to open up the sport of competitive skeet shooting to a larger audience by giving them a product that can accurately and affordably track their scores and improvement in the sport without the need for a real life referee.

As the project has already been started by a previous team of engineers, the team will pick up where the previous team left off by first developing a streamlined method for data collection. Data samples are an integral part in the development of a product like this where video data is processed and evaluated by a computer system. With a streamlined data collection platform, the team can then focus on the development of the actual product with enough data to accurately test the platform.

As the last group worked on hardware purchasing and selection, this group's main goal will be the image processing of the aforementioned data. The group will attempt to find the best video analysis algorithms to accurately score when a clay pigeon is destroyed against a solid (black of night) colored background. This group believes that by focusing on data captured at night time that the device can more accurately and easily evaluate video data for where the clay pigeon flays and is destroyed.

The team hopes that a finished project can fairly and accurately dictate realtime skeet scores for single shooters given any conditions, and given this expectation, supply avid shooters with real time feedback without the need of finding a licensed referee.

1.3 Operational Environment

IC Chipz is a product that needs to be designed to be used to judge skeet shooting. This means that it will need to be a product that is durable and can survive in harsh conditions. This product will need to survive these conditions because since skeet shooting is primarily done outside, it will need to be able to survive most outside conditions. In this project,the goal is to have IC Chipz be able to accurately score at night in various types of conditions (rain or shine and hot or cold).

1.4 Requirements

1.4.1 Functional Requirements

The IC Chipz project will have the following functional requirements:

- Mobile application to monitor video, keep records, and show scoring
- Program which uses image processing to decompose video to determine scoring
- Two reliable camera locations for video capture
- Low error rate
- Product functions at a 95%+ success rate given night time conditions with dark background.
- Automated data collection and flow into analysis systems.
- Automated data labeling
- The computer vision algorithm will be You Only Look Once (YOLO)

1.4.2 Economic/Market Requirements

The IC Chipz device economic requirements are to work with the given systems that the client has provided so that no new purchases are required unless absolutely necessary. The IC Chipz device is a personal project of Dr. Duwe's, so it will have a relatively limited budget compared to a project from a business like John Deere.

1.4.3 Environmental Requirements

The IC Chipz device should work in any outdoor conditions with no change in performance. It should still be able to have the same desired accuracy for every condition on detecting live or dead clay pigeons

1.4.4 UI Requirements

The IC Chipz mobile application should have the following requirements:

- Track the score of the round
- Display whether the clay pigeon is dead or alive after a shot
- The application should output a final scorecard when the round is over
- The ability to review video to dispute a call
- The UI should be organized and easy to use
- The application should store past rounds in a database

1.5 Intended Users and Uses

In IC Chipz, there are two main intended users of this product. The first intended user will be the referees who are judging the skeet shots. They will use this product because it will help them accurately check and see if a clay pigeon has been successfully shot or not, both through the computer vision and through the video feedback through the app. This means that the referees will be able to give the players a much more accurate score. The second main user of this product is the players in Skeet shooting. They will use this product because it will help them keep track of their score without the need of always having a referee. It will also help them because they can use the video feedback in order to study their shots which can be used to help them become better players.

1.6 Assumptions and Limitations

1.6.1 Assumptions

- All skeet shots will take place at night at the request of the client
- The user has an Android/iPhone that they can use so that IC Chipz can send the video feed to that mobile application.
- The skeet range is up to official skeet standards
- The skeet match follows the official skeet match rules

1.6.2 Limitations

- Only be able to visit the test range about once a week
- Weather will be a limitation during the winter
- The Jetson board has the potential to struggle to handle the video that is being transferred to it by the E-Con camera.
- The camera is not an extremely high resolution or FPS camera, which can cause difficulties in picking up small chips flying off of the clay pigeon

1.7 Expected End Product and Deliverables

The IC Chipz final product is a portable scoring device which can be deployed before beginning a skeet round and score 95% of shots successfully. The portable device comes with the ability to pair with a mobile application which acts as a front end interface to the device. Users are able to view current round scores, challenging invalid or wrong scores and access post round information.

- Machine Vision Functional at 95% Accuracy
  - The machine vision portion of the application must meet a 95% accuracy; this means the application must detect a broken clay target 95% of the time.
- Data Collection Mechanism
  - The mobile application and portable device have a data collection mode allowing them to add additional data points to a data set or data sets.
- Auto Data Parsing and Data Labeling
  - With collected data, the portable device shall automatically splice and parse the data into buckets of hits and misses
- Skeet shot scoring [At Night]

    - Machine vision functionality of 95% initially works at night or on darker days
- Mobile Application to Review Scores
    - Mobile application has a score review process to allow invalid scores to be challenged and reversed if necessary
- Mobile Application for Post Round Information
    - Post round statistics information displayed after each round

2 Specifications and Analysis

2.1 Proposed Design

IC Chipz has broken down into three sub teams for the project: Mobile Application, Embedded System, and Data pipeline/integration development. Each team is approaching a problem in their area by testing, reviewing, and updating as necessary. The group has approached designing the project in small steps in sub-teams with the end goal being the image processing of the collected data from the skeet shooting, as the previous years team focused on hardware development.

The team has defined the problem statement with the group's client, the purpose of the project, and its value. The group has also begun the concept process of the design, The Integration/Data Team has created a top level design of the data flow which will allow the team to find a single concept that works best for the project and continue brainstorming. Approaching the planning, the team has identified the people/team, tasks, task durations, and budget needed. The design planning portion of the project will focus on translating the customer requirements and systems engineering model into a working prototype that can successfully track if a target was hit or not hit under specific conditions.

The team has implemented the mobile application, the WiFi connection, and the Darknet algorithm

2.2 Design Analysis

The embedded systems team created a python script that goes through the training and test data sets to establish what is good and what needs to be updated. The team also continued to look into the "Houston" code from last year to determine what is good, and what needs to be changed/updated.

The Integration and Data Team created a top level design of the data flow from the embedded system to the mobile application which will assist in ideas for the data pipeline.

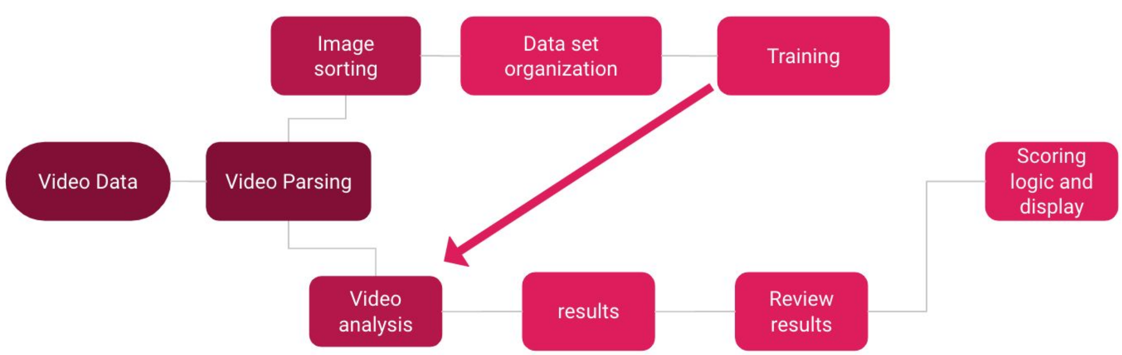
The Mobile Application Team tried and tested the existing mobile application code to see if it was sufficient enough for the team's needs and if the UI is simple enough to navigate. The code from the years previous team was not well documented and the UI was not easily navigable, as a result the mobile team decided to write the application from the ground up in Xamarin Forms which now has all the necessary functionality and a user friendly UI.

2.3 Development Process

The team's development process for this project will be Agile, which divides the functionality into small parts that can be delivered independently as the team begins to start working on them. The team chose Agile weekly sprints with weekly standups have already been apart of the project. This will help to make faster design requirements and get quicker feedback on the groups work from Dr. Duwe.

2.4 Design Plan

The team's design plan for this semester was to continue working on respective functionality for the three three sub teams for the project: Mobile Application, Embedded System, and Data pipeline/integration development. The team was working towards a prototype that can track skeet disks and determine if the shot was a hit or miss. The team will continue to develop and deliver the Functional, Economic, Environmental, and UI requirements to the team's client. Feedback from Dr. Duwe has allowed the team to see what requirements are more crucial and what each smaller team needs to focus on more. While the client wants the final design to be able to track any targets under any conditions, the group's first goal was to first have a prototype that can track just one skeet disk and determine if it was hit or not during the night time.



3 Statement of Work

3.1 Previous Work and Literature

While computer vision has been around for a long time and has been used to solve a multitude of problems, there are very few applications of computer vision used in a sense like IC Chipz, where scoring of a shooting sport is concerned.

One such all in one system that is currently on the market is the Garmin Xero® S1 Trapshooting Trainer. This small camera and display system does almost exactly what the IC Chipz system does and retails for around $1000 (Garmin).

While this product most likely performs well as it has been on the market for some time now, the system offers a sense of user customization. Because the system includes data collection all video is saved and ready for users to review, reference, and assign to data fields for future training of the systems AI.

As this project is a continuation of a previous team's work with the client, the team started with the system components already bought and were able to start integrating parts right away, this saved us what could have possibly been months of research into what hardware would work best.

One of the downsides of continuing a previous team's work is that they failed to document their code properly, so the group has been left to figure out what their code is doing which has set us back some. However, despite this setback the team still saved a lot of time, as it would take much longer to come up with the system from scratch as opposed to simply figuring out undocumented code

3.2 Technological Considerations

IC Chipz is using an Nvidia Jetson TX2 as the mind behind the system. While in reality the power this board supplies is probably much greater than needed to run the applications, it is helpful to have a system powerful enough to not just run the product, but help develop it too. It is easy to tweak the system internally and process larger data than the end product may require with a board as powerful as this. It also takes the need for external server processing out of the equation.

In addition to that, the camera purchased can get video quality up to 4k which is extremely helpful when testing. While 4k video may be overkill for the given implementations, the fact that the functionality is there if needed is extremely helpful.

3.3 Task Decomposition

The main tasks for the full development of the system are...

1. System integration (hardware, application, and mobile all connected)
    a. Create embedded application
    b. Create mobile application
    c. Set up video collection
    d. Connect mobile app to embedded app
2. Automated Data labeling
    a. Create script to label data from video footage
    b. Successfully label data
    c. Run the test data through the machine vision algorithm to further test labeling
3. Algorithm Training
    a. Gather night data
    b. Train algorithm based on night data
    c. Gather day data
    d. Train algorithm based on day data
    e. Gather poor weather condition data
    f. Train algorithm based on poor weather condition data
4. Algorithm testing
    a. Take night footage results from algorithm and check accuracy
    b. Take day footage results from algorithm and check accuracy
    c. Take poor weather footage from algorithm and check accuracy
5. Data/algorithm tweaking
    a. Refine algorithm for night conditions based on testing
    b. Refine algorithm for day conditions based on testing
    c. Refine algorithm for poor weather conditions based on testing

3.4 Possible Risks and Risk Management

The team is concerned that the weather may slow down the implementation of the project, as the team needs to gather data outside at the shotgun range. If it is snowing/raining this can hinder the progress as the team can't gather the data to train the model.

Another potential risk is general safety hazards that comes with shooting shotguns, as gathering data requires the use of shooting shotguns. To manage these risks, the team will only be at the shotgun range with direct supervision and will follow all rules set forth at the range.

3.5 Project Proposed Milestones and Evaluation Criteria

A key milestone the team had looked forward to this semester is data collection automation. With that, the team could start to gather data to train the computer vision with. After the model is trained then the team can integrate the trained model into the data collection schema, and thus evaluate the accuracy of the algorithms reading on data as it is labeled by a user. A success rate of 95% is what the client and the team is looking for given that a success is when the users input of the data matches the algorithm's guess on if a clay pigeon was broken or not.

The specific milestones are as follows:

1. Full System Integration
2. Data labeling completed and fully automated
3. Algorithm is fully trained
4. Algorithm is fully tested
5. Data/Algorithm tweaked as needed
6. Loop back through the milestones as needed

3.6 Project Tracking Procedures

The group will utilize GitLab milestones in order to track progress throughout the course. The group will also utilize Google Drive and a schedule/progress document to help track progress.

3.7 Expected Results and Validation

The end goal is to have a scoring application that reads destruction of clay pigeons at a 95%+ success rate. Once the application can denote whether pigeons are destroyed or not, implementing the scoring system based on vision data should be fairly simple. To test the success of the system, the team will simply have to relate human visualised input to the data and input provided by the application. A key to streamlining this is to make sure the Computer vision works flawlessly and is tested extensively before integrating into the final application.

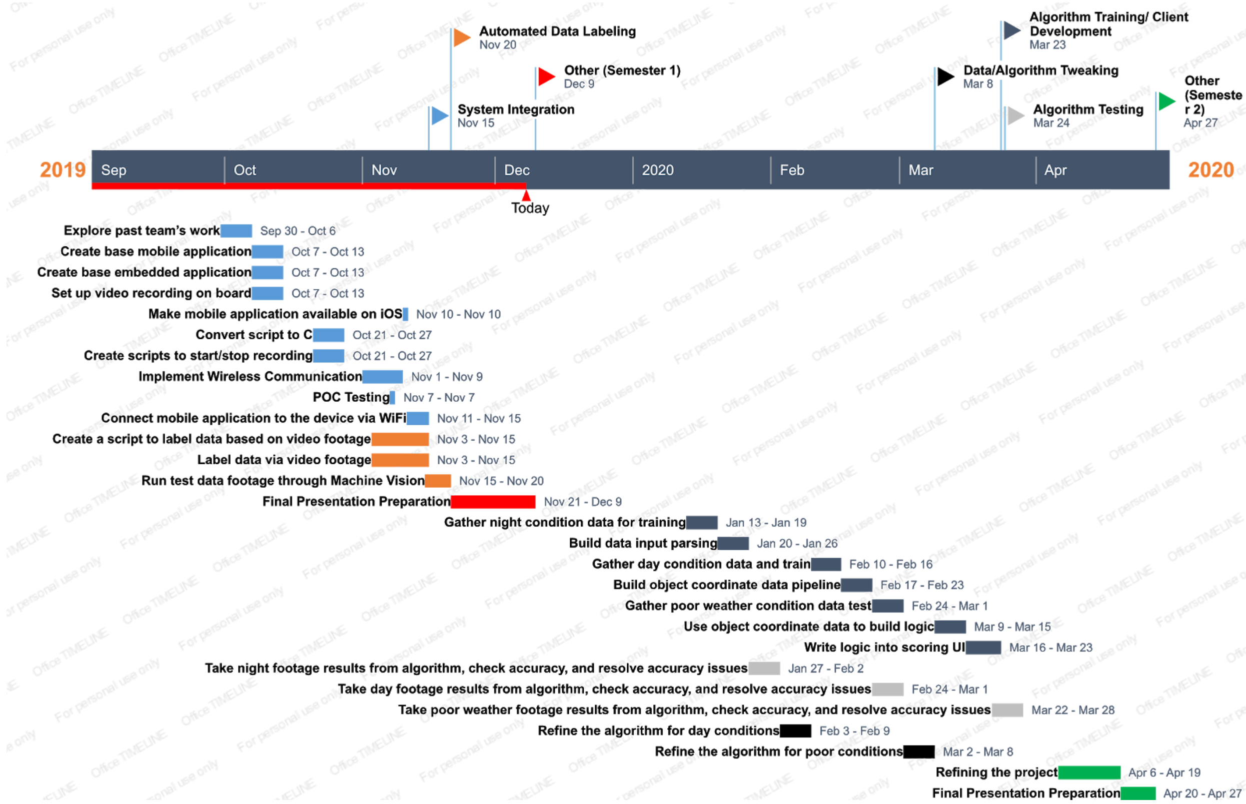
4 Project Timeline, Estimated Resources, and Challenges

4.1 Project Timeline

Task Name	Start Date	End Date
Semester 1		
System Integration		
Explore past team's work	September 30, 2019	October 6, 2019
Create base mobile application	October 7, 2019	October 13, 2019
Create base embedded application	October 7, 2019	October 13, 2019
Set up video recording on board	October 7, 2019	October 13, 2019
Make mobile application available on iOS	October 21, 2019	November 10, 2019
Convert script to C	October 21, 2019	October 27, 2019
Create scripts to start/stop recording	October 21, 2019	October 27, 2019
Implement Wireless Communication	November 1, 2019	November 9, 2019
POC Testing	November 7, 2019	November 7, 2019
Connect mobile application to the device via WiFi	November 11, 2019	November 15, 2019
Automated Data Labeling		
Create a script to label data based on video footage	November 3, 2019	November 15, 2019
Label data via video footage	November 3, 2019	November 15, 2019
Run test data footage through Machine Vision	November 15, 2019	November 20, 2019
Other		
Final Presentation Preparation	November 21, 2019	December 9, 2019

Semester 2		
Algorithm Training/ Client Development		
Gather night condition data for training	January 13, 2020	January 19, 2020
Build data input parsing	January 20, 2020	January 26, 2020
Gather day condition data and train	February 10, 2020	February 16, 2020
Build object coordinate data pipeline	February 17, 2020	February 23, 2020
Gather poor weather condition data test	February 24, 2020	March 1, 2020
Use object coordinate data to build logic	March 9, 2020	March 15, 2020
Write logic into scoring UI	March 16, 2020	March 23, 2020
Algorithm Testing		
Take night footage results from algorithm, check accuracy, and resolve accuracy issues	January 27, 2020	February 2, 2020
Take day footage results from algorithm, check accuracy, and resolve accuracy issues	February 24, 2020	March 1, 2020
Take poor weather footage results from algorithm, check accuracy, and resolve accuracy issues	March 22, 2020	March 28, 2020
Data/Algorithm Tweaking		
Refine the algorithm for day conditions	February 3, 2020	February 9, 2020
Refine the algorithm for poor conditions	March 2, 2020	March 8, 2020
Other		
Refining the project	April 6, 2019	April 19, 2019
Final Presentation Preparation	April 20, 2019	April 27, 2019

Gantt Chart



System integration is one of the longest tasks the team has to do, and the plan is to get that finished by the end of the semester, along with automatically labeling data from a video. The team is giving a week for each task, as there is the weekly standup with the group's client/advisor every Monday and this makes the project compatible with Agile (weekly sprints).

The team is giving the entire final semester for the algorithm training, testing, and refining because it takes a full week for the computer vision algorithm to be trained on a computer, as the team does not have access to a super computer. The team also anticipates various weather delays in collecting data, so the schedule is taking that into account as well. The team's client wants the device to work in all conditions, so there is time set aside for the three major conditions that will affect the device (a day background, a night background, and a poor weather background), as each condition has a different background that the clay pigeon will have to stand out against. There is also a week scheduled for the team to refine the algorithm that will determine the shot result since the algorithm that the team will design for automating the result will also need to adjust depending on the conditions present.

The schedule above is split into 5 major tasks, with 2 being done in the first semester and 3 being done in the second semester. The first semester has more planning to it than the second semester, hence the less tasks. There is also a section for other tasks that need to be done that don't fall into the major tasks, this mainly consists of presentation preparation.

4.2 Feasibility Assessment

The team's realistic projection of this project is that there will be a completed mobile app that sends data to the embedded board when a button is pressed. The embedded board will then begin recording video of a clay pigeon. This video will then be sent to the machine vision algorithm, where it will determine if the clay pigeon was shot in the video or not. If it has, then it will score points and return the result to the user. If the machine vision determined that the user missed a shot, then the user will be notified that they missed the clay pigeon and their score will not be updated. The group would like to have this project work in all conditions, but realistically, the goal is to get it working with high accuracy in night conditions.

Some of the challenges that may make these objectives difficult to complete is the weather, especially with winter weather in Iowa. There may be a week or two where the team can not gather data due to a blizzard or it being too cold. Another challenge is the training of the algorithm. The group has planned on one week for training the algorithm, and then one week for fixing any bugs.

Depending on the condition that is being tested and the accuracy desired, it may take longer for the algorithm to be trained and debugged than the two weeks that the group has allotted. This could be an issue if this deadline is missed as it would put the group behind schedule. Lastly, another challenge that will make these objectives difficult is the debugging of the current machine vision algorithm.

The move to online instruction due to COVID-19 has also affected some of the objectives and made it so the team cannot collect any data as previously planned. The team has been working with the algorithm with previously collected data from the previous team.

4.3 Personnel Effort Requirements

https://sdmay20-10.sd.ece.iastate.edu/docs/Design_Document_V2.docx.pdf

Task	Personnel Effort
System Integration	All groups members will conduct research into the previous teams work. Group members will also communicate with Client to gain an understanding of the objectives. The group will then be split into smaller teams that will focus on one of the pieces of the application. Each member of a team will work on setting up their part so that video recording of skeet shots can begin.
Automated Data Labeling	All team members will focus on sending video data into a data pipeline where the data can be labeled. All members will also run test samples through the machine vision algorithm to determine accuracy, and what work needs to be done in Algorithm Training.
Algorithm Training	All team members will work on fixing the issues that were found in Automated Data Labeling. Team members will also collect video samples in various conditions in order to train this algorithm. The samples being used in Algorithm Testing to help the machine vision learn.
Algorithm Testing	All team members will check the results that were obtained by the machine vision algorithm and make the necessary tweaks to fix them. The team members may also collect new video recordings to test the Algorithm's accuracy, and make changes if needed. Any small bugs that are present will be recorded to be fixed

	in Data/Algorithm Tweaking.
Data/Algorithm Tweaking	All team members will work on increasing the accuracy of the algorithm by fixing the small bugs that were not fixed in Algorithm Testing. The team will refine all conditions in order to increase the accuracy of the machine vision algorithm. The team should have a workable product that obtains the correct outcome 95 percent of the time.

4.4 Other Resource Requirements

Other resources for the project are as follows:

- Xamarin Forms
- Nvidia Jetson TX2 board
- A new camera lens for the TX2 board
- Shotgun ammunition for gathering data at the skeet range
- Clay pigeons

4.5 Financial Requirements

N/A

5 Testing and Implementation

With the project being spread across multiple systems, the majority of the tests will consist of integration and individual modular testing of each system. There are multiple items for each system that have been identified which need to be tested. These items include:

1. eCon's TX130 camera integration
2. Wifi/Bluetooth communication between camera system and mobile device
3. Mobile device integration with user interface
4. Machine vision neural network accuracy testing

5.1 Interface Specifications

There are two different hardware devices, an "embedded" Nvidia Jetson board with a mobile device (iOS or Android). These two devices are connected via the SkeetRecord program over either a bluetooth or Wifi wireless system, depending on the mobile device's preference and available hardware.

With this cross-platform system, many interfaces open up with regards to testing. Specifically, the team is going to target:

1. Commands must be successfully transferred and interpreted across a wireless medium.
2. Video footage can be taken and recorded on the Jetson Board
3. Mobile device can interpret user input and translate it to an underlying action
4. Nvidia Jetson board can transfer files, usually large, over a wireless medium
5. Transferred files can be successfully viewed and results can be challenged

5.2 Hardware and Software

For the Embedded part of the system, there are a couple pieces of hardware/software that need (or have) been tested in the testing phase. The main piece of hardware that the embedded team tested was the Econ-systems camera. This camera (and the software associated with it) was tested because the camera is being used to record the clay pigeons being shot out of the air. If the camera on the board was not working, then the group would not be able to use the software to record footage, and thus the automated scoring cannot be done. Since the rest of the hardware was implemented, there was no other hardware to test. In terms of software, there is a couple pieces of software to test. The first piece of software that has already been tested is the scripts that were created to start/stop recording video footage. This needed to be tested because if these scripts did not work, then the video would need to be manually started/stopped for every shot, which would be inefficient. The second piece of software that needed to be tested is the program that starts the previous scripts when a user on the mobile devices specifies it to do so. This needed to be tested because this would meet the requirement of the mobile application keeping records and scoring.

To handle the machine vision aspect of the system the team decided to use a library called Darknet. This library builds out a neural network based on a training dataset. The team will be continuously testing the underlying neural network using a parallel testing dataset. This verifies the accuracy of the neural network and helps achieve the 95% accuracy requirement.

The mobile application utilizes a framework called Xamarin Forms, which allows for cross platform development in both Android and iOS, and acts as a middle man between the user and the underlying commands the Jetson board can handle. This portion of the project will house typical unit tests.

5.3 Functional Testing

Testing the mobile app will require unit testing to make sure the methods for connection with the Jetson board and tracking the players score during a round are accurate and have the results needed. This will make sure the commands sent by the mobile app are correctly receiving the right data from the board when it takes a video.

More unit testing will be required to test other important functionalities and commands as they are added or deemed necessary. System testing will also be required to make sure the system complies with specified requirements from the client and that the Jetson board is successfully transferring files and that these files are accessible via the mobile application.

5.4 Non-Functional Testing

Non-Functional Testing has focused on the video and file transfer speed, security of the connection between mobile and board, and the compatibility of the mobile application. The mobile app is intended to work on both Android and iOS, which required testing to make sure both applications run the same and outputs the same results. To do this, the mobile team has utilized Apple's Test Flight, which allows for the team members and client to be "Beta Testers" and use the application on their own device. If the application crashes in use, the user can send a crash/bug report to the mobile team. This ensures constant user experiences and results between the two apps. Testing of the video and file transfer speed is important as it will determine the speed of the results to get to the user from the clay pigeon being hit or not. Security of the connection and mobile app is vital, as it will make sure a third party is not able to change or modify the results.

5.5 Process

The Jetson board was tested via system testing. Darknet will be tested using a method called confusion matrices. The non-functional testing covered video and file transfer speed, security of the connection between mobile and board, and the compatibility of mobile application.

Unit testing was used to test the mobile application, along with checking the physical Wifi/Bluetooth connection manually. Along with unit testing on the mobile application, the mobile team used Apple's Test Flight to allow for each team member and the client to download the application on their personal iOS device and send bug and crash reports back to the mobile team.

5.6 Results

Semester 1

The team has been able to successfully implement the eCon's camera integration and Wifi/Bluetooth communication between the mobile app and the Nvidia Jetson board. The ability to send commands through the mobile app using Wifi/Bluetooth to activate the camera commands to record video footage and transfer a video file to the mobile app was completed. In addition, the team is able to mark the video as a hit or miss for testing purposes.

When testing, the team ran into an issue of initially implementing the Wifi/Bluetooth connectivity. At first, the team tried using a USB Wifi adapter to create a peer-to-peer Wifi connection. Through this, the team determined that this would not be possible due to the limitations of the eCon camera drivers. Therefore, the team decided to implement Wifi by using the on-board Wifi connection as a hotspot.

Semester 2

The mobile team has tested everything currently written on the application using Apple's Test Flight program or unit testing. As mentioned earlier, this program allows for the developer to send out invites to Beta Test the program. The mobile team has gotten a lot of crash reports and feedback by utilizing this program and it has been extremely helpful in ironing out bugs in the program. Overall, for the mobile team this semester has been much smoother in the testing side of things due to the discovery of Test Flight.

The embedded team successfully created a Darknet dataset and created model evaluations that were automated for accuracy logs. These allow future developers to view dataset discrepancies in order to Identify problems within the training set and modify the set accordingly to account for those errors. Because of the world's current health situation, we were limited in data collection, which limited us from delivering too far into dataset correction. With that being said, the tools we created to automate the process should help assist future developers working on this project so that they can easily get set up and working to find any issues still present. In addition to the automation of Darknet model evaluations, the embedded team also created an algorithm for automated scoring inside of the overall application. This was done by adding the Darknet C++ API to the overall application. From here, the algorithm would then pull frames from a video feed (either from an already existing file or a current stream), readjust it to Darknet's liking, and place it into Darknet. After this was done, Darknet would then make a prediction on that frame and output it. From here, this output would then be used to see if the status of the clay pigeon had changed, been hit, or left the frame when it had previously existed (indicating a miss). At the end of the recording, the result found by Darknet would then be used to transfer the file to the directory that corresponds with its status. Additionally, depending on the mode, the result would be sent back to the mobile application. Overall, the embedded team made good progress in getting a working algorithm to output the result of a shot to a user, but was unable to complete its goals on improving Darknet, due to the lack of an ability to gather and train new data.

The integration team has successfully converted the entire application into C++, including the scripts used by the eCon camera. The team also implemented openCV into the application to better integrate the eCon camera and darknet library with the overall system. In addition with the changes listed above, the team also worked on more ways to auto label the data collected. Once these changes were made, the board had to have openCV installed in order to compile the new application. Integration testing involved stress testing the connection between the mobile application and the embedded system as well as testing the connection from the system to the darknet library. Overall testing of the integration pipeline shows that connections will remain stable for full extended use of the system.

6 Closing Material

6.1 Conclusion

A significant amount of work has been completed on this project throughout the last two semesters. It began as a slow and steady progression, as a lot of the areas that this project covered this team had never dealt with before. Each individual team has been satisfied with their successes this year, especially due to the interruption that COVID-19 had on the project. Due to the pandemic, the team was not able to get out to the range and gather data to train the algorithm. The next team that takes on this project from Dr. Duwe will have pretty much the entire project laid out for them, with the only thing to do is to gather more data, train the algorithm, and iron out bugs and kinks in things such as the mobile application. Without the pandemic, the team feels like the project could have been successfully completed and all of the requirements met.

Overall the project has been extremely successful and the team feels like it has left behind a lot of documentation (unlike the previous team) and working code for the next team to work with. The team has full faith that the next team to take over the IC Chipz project can reach the 95% success rate laid out by Dr. Duwe.

6.2 References

1. J. Redmon and A. Farhadi, *YOLOv3: An Incremental Improvement*, 2018, https://pjreddie.com/darknet/yolo/.
2. E-con Systems. *"ecam_tk1_guvcview Build & Installation Guide."* e-Con Systems, 23 July 2018
3. Garmin. (n.d.). Garmin Xero® S1 Trapshooting Trainer. Retrieved November 3, 2019, from https://buy.garmin.com/en-US/US/p/612942.

6.3 Appendices

N/A