Smart Cat Door

Authors: Irene Lin, Philip Petrakian, Jing Wu: Electrical and Computer Engineering, Carnegie Mellon University

***Abstract*—This system is capable of recognizing cats in a video feed and determining if the breed matches an authorized breed. Computer vision detects and tracks movement. Machine learning determines if a cat is present, and if the cat is the correct breed. The system is bidirectional to allow freedom, quiet to not scare the cat, accurate so cats are not locked out and raccoons are not let in, and fast so the cat is not left waiting for the door to open. A mobile app allows the owner to remote lock and unlock the door and authorize pets.**

***Index Terms*—cat breed, cat door, computer vision, machine learning**

I. INTRODUCTION

The purpose of this system is to keep unwanted animals out of your home while allowing a pet owners cat to have the freedom of entering and exiting. Current smart cat doors use RFID technology and have poor reviews online [4][5][9]. Recurring themes in the product reviews were about the desire to keep raccoons out, have consistent and reliable door opening and locking over extended period of time, and to have longer battery life. Other complaints included the sound of the bolt scaring the cat and the owner having to retrain the cat to push through a heavier door.

In our design, a camera mounted on top of the cat door will send footage to the processor. Computer vision is used to detect and track movement and a machine learning algorithm is used to decide if the animal is your cat. A panel covering the cat door is lifted by a servo and closes after the cat has passed through. Through a mobile app, the pet owner can remote lock and unlock the door, set a curfew time, view logs, and authorize multiple pets.

II. DESIGN REQUIREMENTS

Assumptions:

- The pet will walk towards the camera headfirst
- The pet is not covered in anything that significantly changes its appearance

Requirements:

1. Raccoons get in 5% of the time:
    - Reasoning: With current smart and non-smart cat doors, raccoons get in 100% of the time because raccoons are clever enough to lift latches and strong enough to brute force their way in. Any number less than 100% is already an improvement. Facial recognition for humans have around a 5% miss rate, so we decided to challenge ourselves to match this.
    - Unit Test: 15% of the data set will be withheld for testing.
2. Owners cat is stuck outside 5% of the time:
    - Reasoning: If a cat enters and exits four times a day, then across five days, the owner gets one notification. This is reasonable.
    - Unit Test: 15% of the data set will be withheld for testing.
3. 1.09 s for door to open when the cat starts at 1m away from the outside:
    - Reasoning: Cats walk 3.3 km/h (2 mph) on average. When a cat has moved to within 1 meter to the door, we interpret this as the cat having the intention to enter the house.

    $$1.09s = \frac{3600seconds}{1hour} \times \frac{1hour}{3.3km} \times \frac{1km}{1000m} \times 1m$$

    - Unit Test: Have an authorized cat start from 1 meter away and walk towards the door. Time with a stopwatch how long it takes the door to open.
4. Detect when the cat is all the way through and close the door after:
    - Reasoning: The door should not close while the cat is still in the middle of the doorway, as to not injure the cat.
    - Unit Test: Human walks to and from the door. The purpose of this test is to measure the accuracy of the PIR sensors. A live cat is not needed because to the PIR sensor, a live cat and a human both look the same because both emit heat and radiation.
5. Set curfew, remote locking and unlocking, authorize breeds:
    - Reasoning: There are times when an owner does not let their cat outside (e.g. when it is raining outside). The owner needs to be able to set a curfew for when the cat is not allowed outside. The owner also needs to be able to initialize the system to work with his cat's breed type.
    - Unit Test: This will be tested by setting parameters through the mobile app and testing if the door opens when presented with video footage of an authorized cat and does not open for video footage of an unauthorized raccoon.
6. Consistent lighting for nighttime:

- Reasoning: Computer vision requires consistent lighting and the camera needs to be able to obtain clear footage of the cat even at night.
- Unit Test: The LED turns on and the door opens when the lights are off and the door is presented with a live cat.

7. Sturdy enough to withstand 13.6kg of resistance:

   - Reasoning: A research study [7] conducting strength testing on raccoons indicated that adult raccoons can lift 13.6kg of resistance.
   - Unit Test: Strength testing will be performed throughout the construction period of the door.

8. Operate under 70dB:

   - Reasoning: This number comes from the sound intensity of a vacuum cleaner. Cats are often afraid of fireworks, thunderstorms, and vacuum cleaners, with vacuums being the most quiet of the three [3].
   - Unit Test: A decibel reading app will be used to measure the sound intensity of the door while it is in operation.

Full System Testing: The product will be taken off campus and tested using a real cat, courtesy of our friends. We will film the working system and that it meets all the requirements. Budget permitting, a taxidermy raccoon will be used to show that the system does not let a raccoon in. Otherwise, videos[1][11] of raccoons will be fed into the computer vision script.

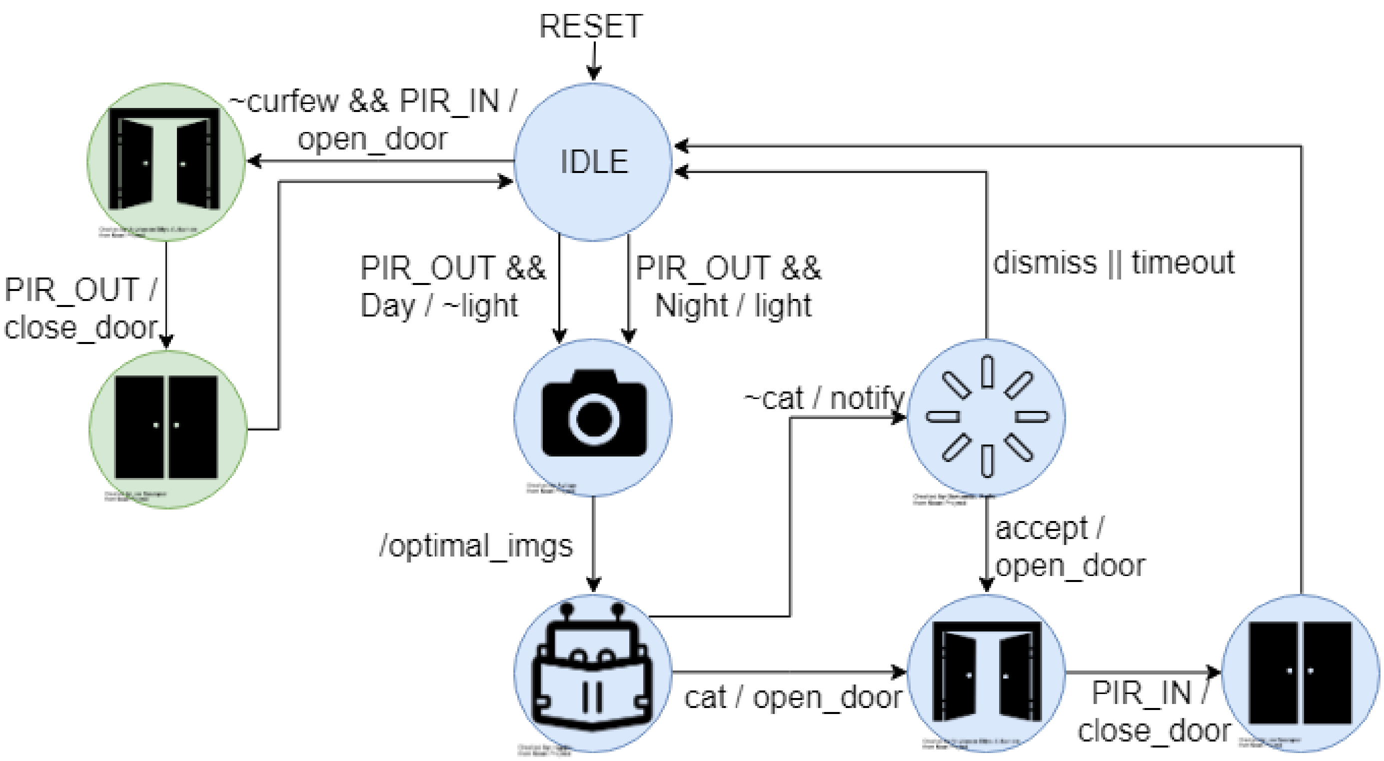

Figure 1: Event State Diagram

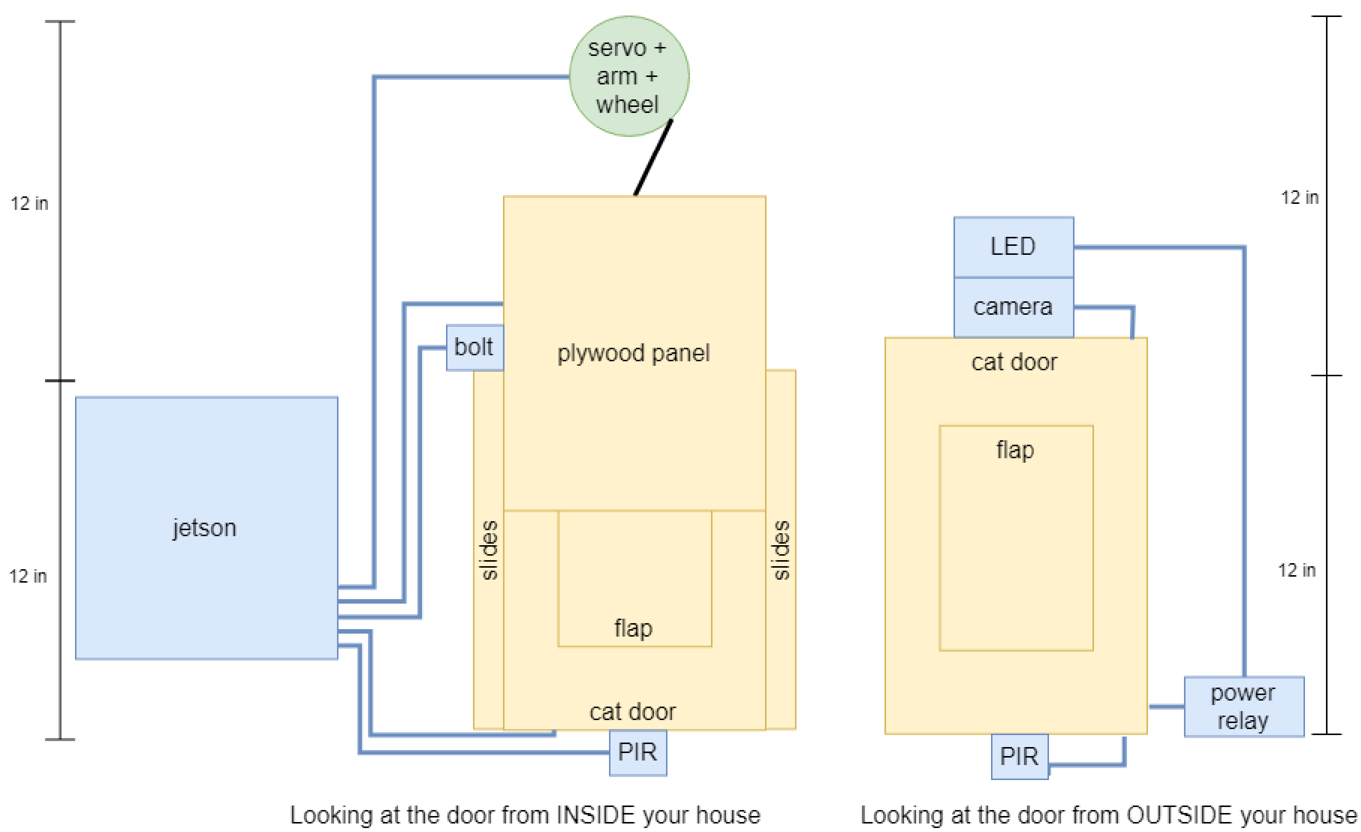

Figure 2: Door Schematic

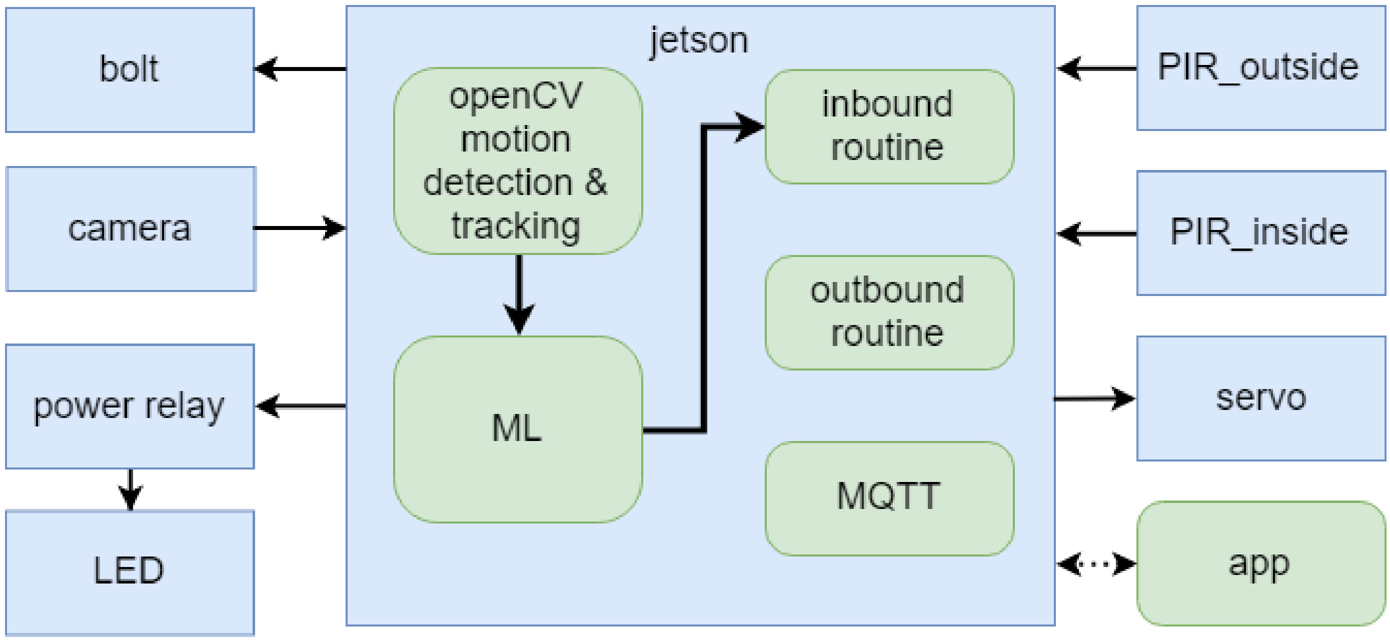

Figure 3: Block Diagram

III. ARCHITECTURE OVERVIEW

Figure 1 illustrates a state diagram of the system. Blue states are for entering, green states are for exiting. For a cat wanting to exit, the PIR sensor mounted on the indoor side of the cat door will detect when the cat has moved close to the door. If it is not curfew, then the door will open. After the cat has passed out of range of the outdoor sensor, the door will close. For a cat wanting to enter, the camera will be detecting and tracking movement, and PIR sensor on the outdoor side of the cat door will detect when a cat has moved close to the door. If it is night time, the light will turn on. This will help the camera to produce better images to feed into the machine learning algorithm. The machine learning algorithm will determine whether or not the door should be opened. In the case that the algorithm decides the door should open, the PIR sensor on the indoor side will indicate when the cat has passed out all the way through the door and the door will close. If the machine learning algorithm decides that the door should not open, a notification will be sent to the pet owner and they can decide if they would like to remote unlock the door or dismiss the notification.

Figure 2 depicts how the devices and non electrical parts will be linked. Blue indicated device and yellow indicates non electrical component. Blue lines indicate wires. The diagram is drawn to scale.

Figure 3 describes how software and hardware components will interact. Blue is hardware and green is software. Motion detection and tracking will be implemented in Python and openCV. The machine learning algorithm will be implemented using Python and TensorFlow. The inbound and outbound routines handle the PIR sensors, power relay, solenoid bolt, and servo. The mobile app communicates with the jetson through the MQTT protocol and will be implemented using Xcode in Swift.

IV. DESIGN TRADE STUDIES

A. Motion Detection

Motion detection is an important part of our project because we need to know when an object has passed in front of the door before we can begin to classify the object. While a PIR sensor alone can detect motion within a certain proximity, our primary method of detecting motion will be using a motion detection algorithm on our video feed. The PIR sensor will act as a secondary sensor which turns on the LED lights when something walks in front of it, providing the camera with sufficient lighting to send a visible video feed to the Jetson board.

We chose to use video motion detection over a traditional proximity sensor because video motion detection will provide better images to our machine learning classifier, in addition to detecting motion. When taking pictures of small animals from one to two meters away, the animal may take up less than $\frac{1}{9}$ of the entire image space using a traditional phone camera. By being able to detect where objects are in the image, we can crop that part of the image out and only send the cropped image to our machine learning classifier, resulting in a more accurate and consistent output.

B. Machine Learning Classification

Our project ultimately aims to do facial recognition by examining the specific features of cats (eyes, ears, nose, fur, etc.); however, due to a lack of available data sets, the time frame for the project, and the skill set that we possess, we decided that it was unfeasible to label features of the cats and train on those features in the time frame of the project. Instead, for the first iteration of the cat door, we will classify cats by breed, and only allow specific breeds to enter and exit the house. We will perform two iterations of classification, once for determining if an image is a Cat or a Non-Cat, and once for determining the specific breed of the cat.

Convolutional neural networks are the most popular and successful image classifiers today and will act as our machine learning classifier [2]. Most convolutional neural networks use at least one of each of the following layers: convolution, activation, max pooling, fully connected (dense), and softmax inference. We decided that having one of each layer would suffice because the sample space of possible images is small - our camera is stationary and will be focused on an area close to the ground. Therefore, classification will naturally be more accurate and not require a complicated neural network. Animal recognition research done at the University of Zilina [10] showed that a convolutional neural network with one more convolution and activation layer than our neural network successfully recognized animals on average 95% of the time. If our classification algorithm fails to meet the 95% mark, we can add one or more convolution and activation layers to improve the accuracy of our classification at the cost of speed.

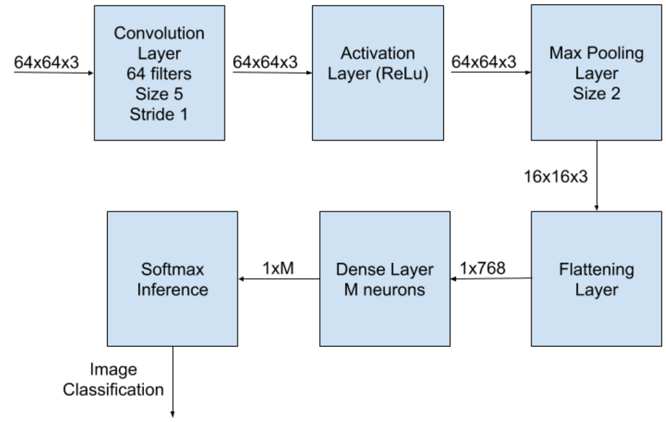

Figure 4: Convolutional Neural Network

C. Testing Tradeoffs

In order of preference, the testing options are as follows: live animals, taxidermy, video feeds, printed high resolution pictures, and stuffed animals. Irene's friend has a cat and Jings friend has a cat, but animals arent allowed on campus. Thus, we will record footage of said cats interacting with the system and the system responding appropriately. For raccoons, we wont be able to find live raccoons to test our system on. A taxidermy raccoon is as close as we could get to a live raccoon, but they are expensive. We have decided to look for videos of raccoons facing headfirst at the camera such that it is very similar to the video feed the camera would have captured of a raccoon. Stuffed animals are not desireable as a test subject because they do not look like real animals. A machine learning classifier that classifies a cat plushies as a real plushie is a poorly built classifier. We would be better off printing high resolution photos of cats and raccoons.

D. Hardware Tradeoffs

We want to minimize the latency of our computer vision and ML algorithms because we want to be able to open the door for a valid cat as it is walking up to the door, without having the cat needing to wait. We estimate that the cat will be within range of the camera for a total of 1.2 seconds.

Through our research we determined that a Raspberry Pi would allow us to compute around 1 frame per second, which is too slow because we could potentially only receive one image during the 1.2 second span and this image might not give a good indication of whether the animal is valid or not. Similarly, we looked into Odroid which is a board similar to the Raspberry Pi, but much more powerful. This would likely yield us 2-3 frames per second. Still, we are unsure if this frame rate is fast enough and we want to be sure that we are going to get at least one good image for our algorithms.

We then looked into GPUs, which are processing units designed for image processing. Nvidia makes the most commonly-used and best documented GPUs. In addition, one of our group members has experience with Nvidia GPUs. We found the Jetson family, which are GPUs created for the embedded systems world. Specifically, we chose the Jetson TX2, which has 256 Cuda cores, because based off of our research we will be able to process 15 frames per second. Furthermore, Nvidia has a library called TensorRT, which compliments TensorFlow. This library can be used in conjunction with TensorFlow to optimize the ML algorithm computation for Nvidia GPUs. We will be using this to improve the latency of our algorithm.

Power is not listed as a requirement or metric because we will never be making a decision based on power. The device will be plugged into a wall outlet.

E. Door Construction Tradeoffs

The two options for pet doors are a swinging door and a lifting door. Online reviews for swinging doors indicated that raccoons were smart enough to lift latches and strong enough to brute force their way through cat doors. We decided to go with a lifting door using drawer slides oriented vertically. This will be much harder for a raccoon to muscle through. A solenoid bolt will be used to prevent a raccoon from sliding the door upwards.

V. SYSTEM DESCRIPTION

A. Motion Detection

The motion detection algorithm will be implemented in Python using the OpenCV library. First, we can store a weighted average of previous frames and call this our background frame. With the weighted average, the script can dynamically adjust to the background, even as the time of day changes along with the lighting conditions. Then we compare the background frame to the current frame by subtracting. If the delta is above a certain threshold, then we have detected motion as a substantial difference in the image. We know where the motion occurred in the frame, so we can crop that part of the image out and send it to the Convolutional Neural Network Classifier.

B. Convolution Neural Network

Once motion has been detected, it will feed a set of 8 images to the Convolutional Neural Network for classification. For our first inference function, we will classify between 6 number of entities, as our cat door will be stationary in front of a door and the number of different objects that could come into the camera view is few. Adding more entities is trivial, as we can download a set of images of the new entity from an online database and retrain our model. The current list is as follows:

- Cat
- Dog
- Squirrel
- Raccoon
- Shoes
- Legs

For our second inference function, we will classify between 12 different cat breeds:

- Abyssinian
- Bengal
- Birman
- Bombay
- British Shorthair
- Egyptian Mau
- Maine Coon
- Persian
- Ragdoll
- Russian Blue
- Siamese
- Sphynx

C. Automatic Door

A PIR sensor will be mounted to the bottom of the door on the indoor side in order to know when the door needs to open for a cat wanting to exit the house. The camera will be mounted on the top of the outdoor side and angled downwards. The camera will be used in determining when the door needs to open for a cat wanting to enter the house. A door switch will be used for when the servo needs to lock after a cat has finished entering or exiting the house.

Passive infrared sensors detect changes in infrared radiation. All objects with a temperature above absolute zero emit heat energy in the form of radiation, so a PIR sensor can be used to sense movement of people, animals, or other objects.

The LED Photography Lighting Kit is commonly used for photography studio, lighting for video, images, collocation with all kinds of tabletop studio, and video shooting. It will provide consistent lighting for the camera.

D. iPhone Application

We want the user to be able to communicate to the smart pet door at any time, not necessarily within proximity of the door. Bluetooth communication is done within close proximity within devices, whereas Wifi works as long as both devices have a connection. Therefore, the smart pet door will be connected to Wifi through the Jetson development board. We are assuming that the whole house has a Wifi connection. We want the Jetson to be able to receive commands from the user and send data back to the user. We will be using the MQTT protocol to communicate between the two devices as the cocoaMQTT library can be implemented in Swift (for the iPhone app) and in Python (for the Jetson). This is a pub/sub form of communication which is easy to use if a user has more than one pet door.

We want the user to be able to manually lock or unlock pets leaving the house, set an automatic timer to lock or unlock pets leaving the house, add or remove a pet with its breed type, and to view the statistics for the door use. We will be implementing all of these features through an iPhone app using Xcode in Swift.

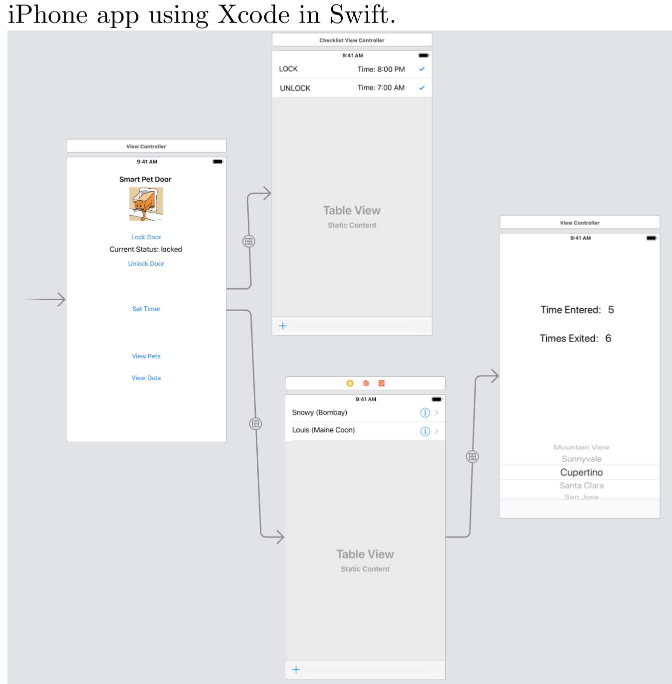

Figure 5: Wireframe for iPhone app

E. System Hub

Our system will need to communicate over Wifi to a phone, receive camera footage, apply our Computer Vision and ML algorithms, control the servos for the door, turn on and off and LED, and receive PIR data. Originally, our plan was to use a Raspberry Pi to accomplish all of these requirements. However, after selecting the Jetson TX2 to compute the Computer Vision and ML algorithms, we discovered the Jetson TX2 developer kit. In addition to having the Jetson TX2 GPU, this board also has a quad-core Arm based processor, 8 GB of memory, 32 GB of flash storage, USB ports, GPIOs, and Wifi capabilities. It also runs a Linux OS. In fact, this developer kit has the same set of features as a Raspberry Pi, that we would need to use for this project. Therefore, we concluded that the Raspberry Pi would add unnecessary complexity to our system. As previously discussed, we will be using the MQTT protocol for Wifi communication. We want a camera that will be able to compute at least 15 fps as we expect our algorithms to be able to process images at 15 fps. In addition, the video quality we need is approximately 100 pixels by 100 pixels. Modern cameras easily satisfy both those requirements, we decided to go with a 720p, 30 fps camera that uses USB to communication with the developer kit. Next, the servos for the door, the LED, and the PIR sensors will be communicating using with the developer kit over the GPIO pins. We want to be able to detect movement within a 2 meter radius of the door as this is when we wish to activate the opening or closing of the doors. We chose the PIR sensor because it uses heat signatures within a specified radius to detect how far movement is. It outputs the values using a single wire.

VI. PROJECT MANAGEMENT

A. Schedule

Our schedule is broken into three parts - implementation, integration, and testing. We spend approximately 4 weeks building our individual parts, 2 weeks integrating them together, and 2 weeks for testing and slack. See Gantt chart in Appendix A.

B. Team Member Responsibilities

Each team member is primarily responsible for technical portions of the project, and secondarily responsible for the integration portions and the management portions of the project.

Irene is primarily responsible for constructing the door and detecting motion. Philip is primarily responsible for implementing the mobile app, integrating all of the software components within the Jetson (sensor triggers and algorithms), and accelerating the ML computation on the GPU. Jing is primarily responsible for training the machine learning classifier and integrating it with the Jetson GPU. Everyone is responsible for keeping each other updated with their progress. Every week, we have three meetings in person and one meeting online.

C. Budget

Part Name	**Quantity**	**Price**
Logitech C270 Camera	1	\$19.86
PIR (motion) sensor	2	\$19.90
Cat door: 5.125in x 7.5in	1	\$29.95
Metal Gear Servo	1	\$19.95
Machined Aluminum Servo Arm	1	\$4.50
Bottom Mount Drawer Slides	1	\$5.28
Plywood	4	\$23.45
LED	1	\$34.99
Power Relay	1	\$19.85
Solenoid	1	\$18.69
Jetson TX2	1	Subsidized
Total Price		\$196.42

D. Risk Management

One scheduling strategy that we applied was budgeting the project's features rather than budgeting the project's time. For many engineering projects in real life, deadlines are pushed backwards when tasks cannot be complete. However, given the set time frame of this project, deadlines cannot be pushed back. To compensate, we thought of the tasks which absolutely needed to be complete for a minimum "demo-able" product.

1. Motion is not detected by algorithm → use PIR sensor to detect motion. This triggers the camera to take a photo that does not get cropped.
2. Classifier does not work → Input a default image to our classifier to always return a correct result.
3. GPIO pins on Jetson unable to communicate with sensors (such as door servo, LED, PIR) → use a Raspberry Pi which will communicate to the Jetson

One difficult challenge we faced was maintaining clear goals and expectations across the team. During the first few weeks of the project, team members had different ideas of what needed to be done, or what standards needed to be met. To account for this, we held discussions with Professor Bain[12], who gave us various approaches to overcome project management challenges. To tackle communication and management issues, we employed two strategies to ensure that tasks are completed more efficiently without any confusion among team members. First, we decided to have one more online meeting every week, in addition to our three meetings in person, so that team members could update each other on their progress. Second, we changed our task list to be a list of goals rather than actions. For example, rather than the task being "Implement and test algorithm," we worded it to be "Be able to pass tests for the algorithm." The first phrase is a simple command, but doesn't specify what the goal is, whereas the second phrase provides the team member with an idea of what constitutes the task as complete.

VII. RELATED WORK

A. Regular cat door

The obvious original design is a simple cat door with a flap, that a cat or any animal is able to go through. The clear problem is that any animal can get through the door, and can potentially wreak havoc within the house. In addition, some cats never gain the knowledge or confidence to use a cat door because it is not intuitive for them.

B. RFID activated cat door

An alternate to the cat door is an RFID activated cat door. This system works by placing a collar on the cat with an embedded RFID tag. The benefit to this system is that it only allows your cat in, assuming it is not next to the door with another unwanted animal nearby. The downside with this system is that a lot of cats easily lose their collars or simply refuse to wear one. In addition, as previously stated, some cats do not like using a door with a flap.

VIII. SUMMARY

A. Future Work

i. Self Learning Algorithm: If we were to do this project again, we would design a self learning algorithm. For the first few times a cat walked up to a door, the camera would take a picture and the processor would notify the owner. The owner would see the picture and have the option to open the door remotely. Over time, the algorithm would learn who should and shouldnt be let in, notifying the owner less and less.

ii. IR Lighting: An alternative lighting mechanism would be IR lighting. IR wavelengths of 850nm and 940nm (also called NIR Near InfraRed) are commonly used in machine vision. IR reduces color of objects, glare, and reflections. IR has a longer wavelength than visible light which usually results in a greater transmission of light into a material through materials like paper, cloth and plastic. IR wavelengths react differently on materials and coatings than visible light, so certain defects and flaw detection can be identified with IR where visible light did not work. One drawback would be that IR lighting changes the color of the cats fur in the image and therefore, our machine learning model would have to be trained on IR images. This dataset is hard to find.

iii. Zero Weight Counterbalance Spring: The door does not have a safety mechanism for opening and closing. In order to protect cats passing through the door, a zero weight counterbalance spring would be used with the doors lifting panel and servo. This would allow the door to open faster, but close slowly. The door would spring upwards rather than down on the cat in the event of a malfunction.

iv. Audio Processing of Cat Meows: We set a target false positive of raccoons let in 5% of the time, because this is already a great improvement over current cat doors, both smart and non-smart. But this means that if a raccoon attempts to enter a house through the cat door once a day, then a raccoon will succeed once every three weeks. The machine learning model could be used in conjunction with audio processing of cat meows in order to increase the accuracy and decrease the percentage of false positives. Cats have complex vocal chords, and therefore different cats have different purrs and meows[8]. A project has been done to build a classifier for meows and woofs[6]. Future work would build on this research.

References

[1] National Geographic. *Family of Urban Raccoons.* Dec. 28, 2016. URL: https://youtu.be/gzioRuXObOM?t=301.

[2] Sustkever Krizhevsky and Hinton. *ImageNet Classification with Deep Convolutional Neural Networks.* URL: https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf.

[3] Litter-Robot. *Why Are Cats and Dogs Scared of Fireworks?* URL: https://www.litter-robot.com/blog/2018/06/29/cats-and-dogs-scared-of-fireworks/.

[4] Cat Mate. *Elite I.D. Disc Cat Flap with Timer Control.* URL: https://www.amazon.com/Cat-Mate-Elite-Timer-Control/dp/B000XPSH34.

[5] PetSafe. *Electronic SmartDoor, Automatic Dog and Cat Door.* URL: https://www.amazon.com/PetSafe-Electronic-SmartDoor-Automatic-Activated/dp/B000WJ0IGA.

[6] Summer K. Rankin. *Classification Of Meows And Woofs.* Nov. 7, 2017. URL: https://www.summerrankin.com/dogandponyshow/2017/10/16/catdog.

[7] Nathan P. Snow. "Strength Testing of Raccoons and Invasive Wild Pigs for a Species-Specific Bait Station". In: *Wildlife Society Bulletin* 41.2 (2017). DOI: https://www.aphis.usda.gov/wildlife_damage/nwrc/publications/17pubs/REP\%202017-031.pdf.

[8] Melinda Story. *Kitty Sounds Cat Noises and their Meanings.* July 14, 2017. URL: https://www.thehappycatsite.com/cat-noises/.

[9] SureFlap. *Microchip Cat Flap.* URL: https://www.amazon.com/SureFlap-Microchip-Cat-Door-White/dp/B003EGIM3O.

[10] Trnovszky. *Animal Recognition System Based on Convolutional Neural Network.* URL: https://www.researchgate.net/publication/320162958_Animal_Recognition_System_Based_on_Convolutional_Neural_Network.

[11] MKI tv. *Raccoon eats grapes with his little hands.* Nov. 21, 2013. URL: https://youtu.be/jyCmnvBbm8Q.

[12] MKI tv. *Raccoon eats grapes with his little hands*. Nov. 21, 2013. URL: `https://youtu.be/jyCmnvBbm8Q`.

Appendix A

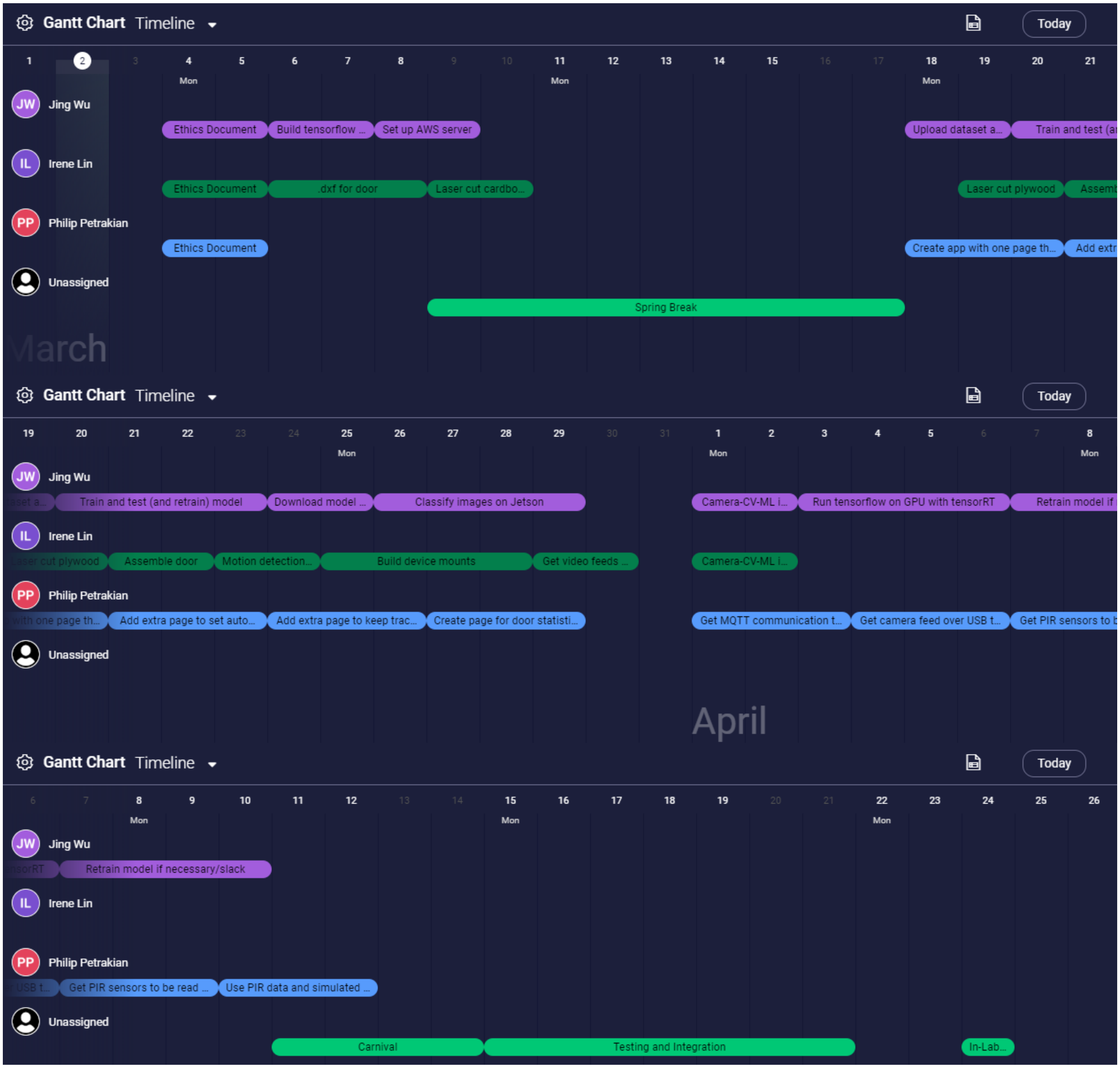

Figure 6: Gantt Chart