Internship 2022

# Progress report format for team meeting

Name: ADHO MAMO

## Tasks completed last week

* [#92] Calibration of camera to take distance measurement

In order to determine the distance from our camera to a known object or marker, we are going to utilize triangle similarity.

The triangle similarity goes something like this: Let’s say we have a marker or object with a known width W. We then place this marker some distance D from our camera. We take a picture of our object using our camera and then measure the apparent width in pixels P. This allows us to derive the perceived focal length F of our camera:

F = (P x D) / W

For example, let’s say I place a standard piece of 8.5 x 11in piece of paper (horizontally; W = 11) D = 24 inches in front of my camera and take a photo. When I measure the width of the piece of paper in the image, I notice that the perceived width of the paper is P = 248 pixels.

My focal length F is then:

F = (248px x 24in) / 11in = 543.45

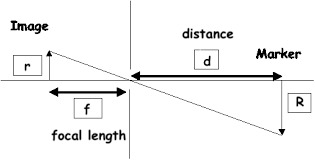
As I continue to move my camera both closer and farther away from the object/marker, I can apply the triangle similarity to determine the distance of the object to the camera:

D’ = (W x F) / P

Again, to make this more concrete, let’s say I move my camera *3 ft*(or 36 inches) away from my marker and take a photo of the same piece of paper. Through automatic image processing I am able to determine that the perceived width of the piece of paper is now 170 pixels. Plugging this into the equation we now get:

D’ = (11in x 543.45) / 170 = 35in

Or roughly 36 inches, which is 3 feet.



* [#93] Object detection in the e-tricycle

As discussed we will use the ultrasonic sensors at the front to detect any obstacle on the path of the tricycle.

A buzzers to will be used to notify the operator of any obstacle.

However there are challenges involved in this operation:

* The tricycle will stop instead of decelerating in front of a dynamic object
* There will be unnecessary shut down of the tricycle as the current model cuts current supply to the motor for it to stop.
* It would also be difficult to pack close range near other objects for example a wall.

## Tasks in this week

* [#88] Raspberry Pi and Navit

# Timeline

|  |  |  |
| --- | --- | --- |
| Month | Intern week | Tasks |
| Jan |  |  |
| Week 1 | Identification of parts and drawing of the chassis diagram. |
| Week 2 | Circuit diagram and acquisition of parts. |
| Week 3 | Definition of the path to be followed by the robot car.  Laser cutting of the parts. |

|  |  |  |
| --- | --- | --- |
| Feb | Week 4 | * Assembly of the robot * Ultrasonic program implementation |
| Week 5 | * GPS and compass navigation * Path definition |
| Week 6 | Object identification using computer vision. (Raspberry pi & camera) |
| Week 7 | Transmission of live feed and data from the robot (transmitter and receiver) |
|  | Week 8 | Object dection (static and dynamic) |