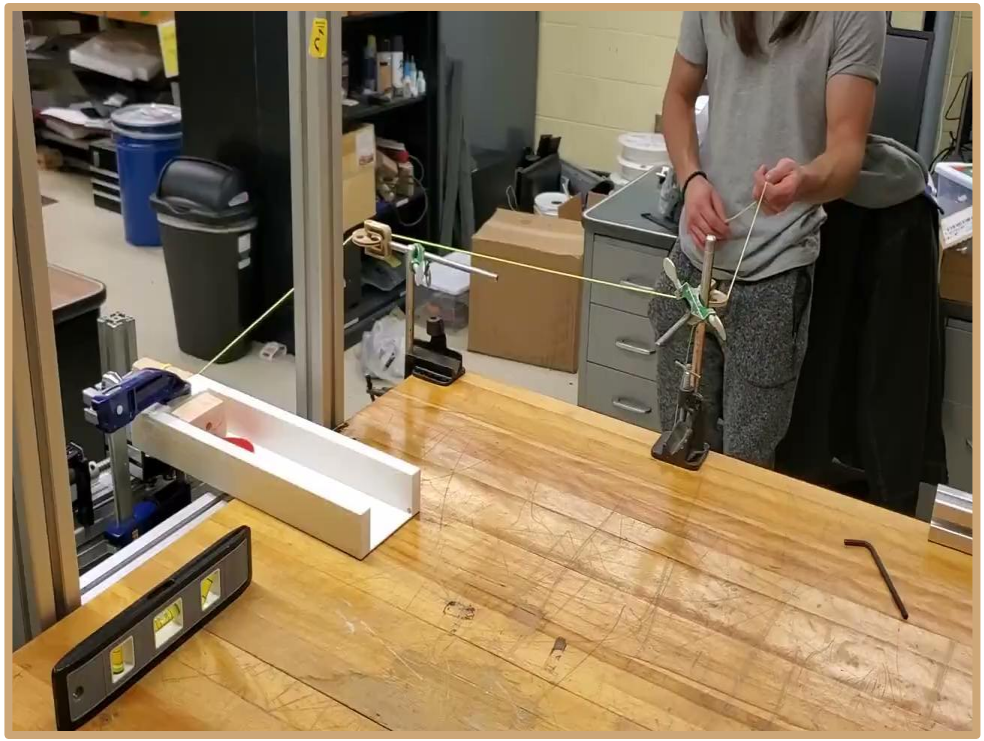
Data Analysis Teaser

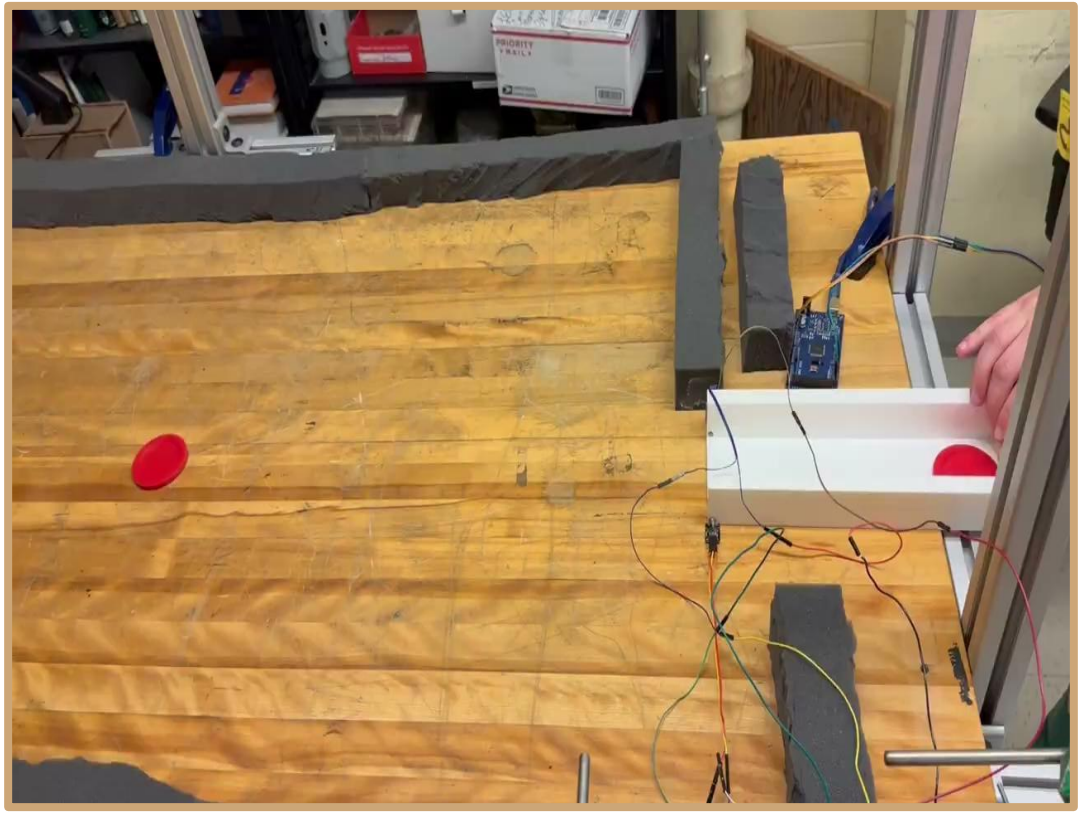
By: Logan Barrus, Phil Coady, Aassik Pazhani, Javier Tort, Jason Vazquez

Setup

- Launch is wired through a string-trigger system
- Pucks are fired with a spring-loaded pinball plunger
- Barriers are made out of foam

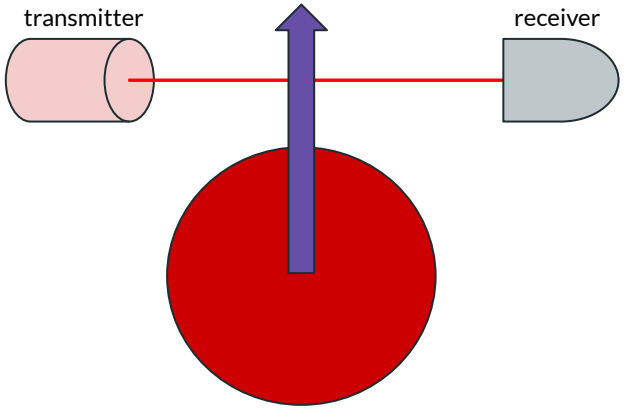


The Collision



Finding Velocity (Initial)

- Initial Velocity found through Laser system
- Unfortunately in our most recent data collection, there was an issue in writing the initial velocity data to a new CSV.
- On the right is data collected from preliminary tests.



velo (m/s):	Average Velocity (m/s):
0.07	1.2035
1.48	
0.09	
0.5	
1.59	
0.6	
0.2	
1.63	
0.16	
0.33	
0.36	
4.13	
0.27	
0.06	
1.82	
1.88	
2.3	
3.1	
3.44	
0.06	

```
if (value1==1) {
  if (check1 == 1) {
    endTime1 = millis();
    check1 = 0;
    Serial.println("this is sensor1");
    Serial.println(endTime1);
    Serial.println(startTime1);
    Serial.println(endTime1 - startTime1);
    myFile.println((diameter/(endTime1 - startTime1))* 1000);
```

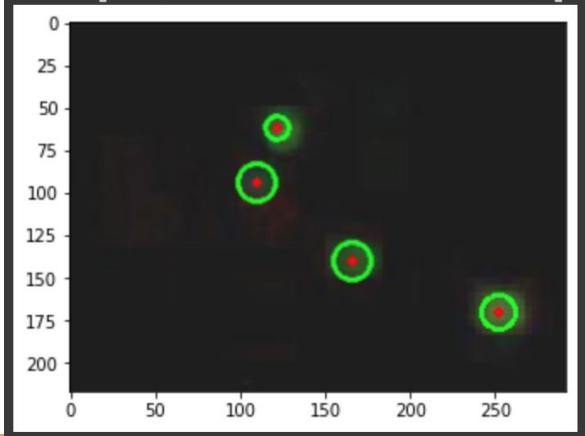
Finding Velocity (Final)

Finding Velocity (Final)

Finding Velocity (Final)

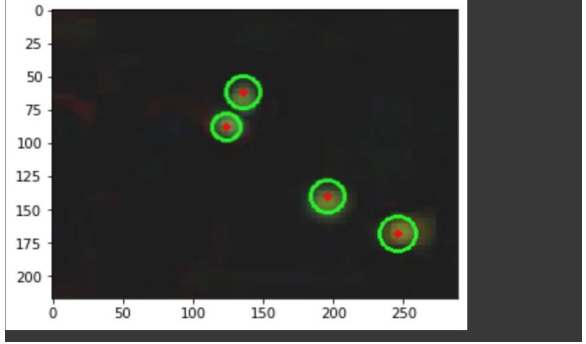
```
filename = 'impact1.mov' #insert file here
video_data = vread(filename)
start = 2
end = 3
coordinates,shapes= circ(start,end,1,2,30,20,18)
print(filename)
print('north puck:',velo(coordinates,shapes,11,start,end,25,30,2,3,6,7),'inches per second')
print('south puck:',velo(coordinates,shapes,11,start,end,25,30,0,1,4,5),'inches per second')
```

```
impact1.mov
north puck: 39.231707077621174 inches per second
south puck: 114.13796482007358 inches per second
```

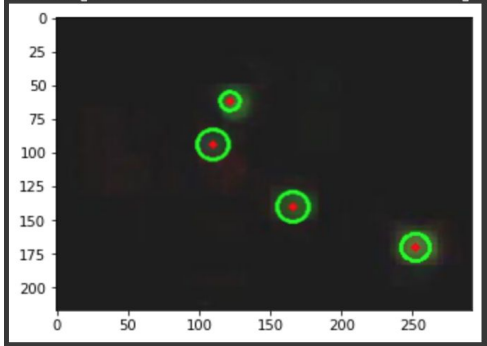


3/22 Final Velocity Data

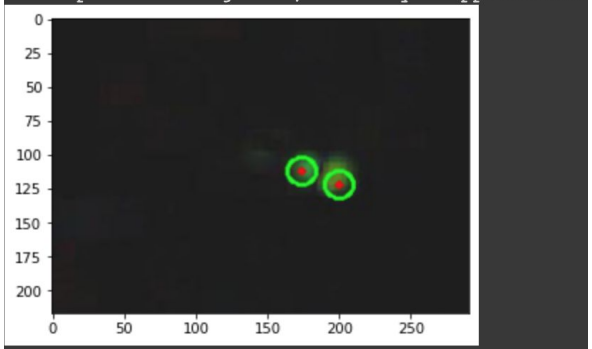
```
north puck: 33.35679934167473 inches per second
south puck: 70.97346436506977 inches per second
```



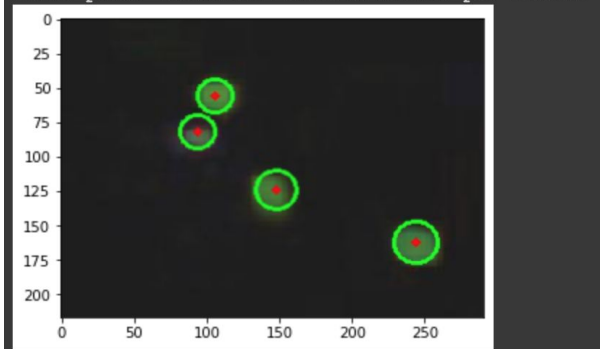

```
impact1.mov
[166, 140, 110, 94, 252, 170, 122, 62]
(217, 292, 3)
north puck: 39.231707077621174 inches per second
south puck: 114.13796482007358 inches per second
```




```
south puck: 34.846193821041616 inches per second
north puck: unrecognized, velocity = approximatel
```

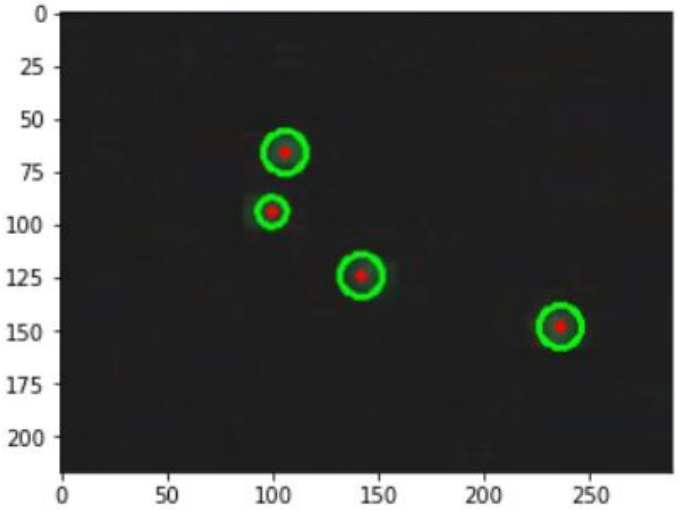


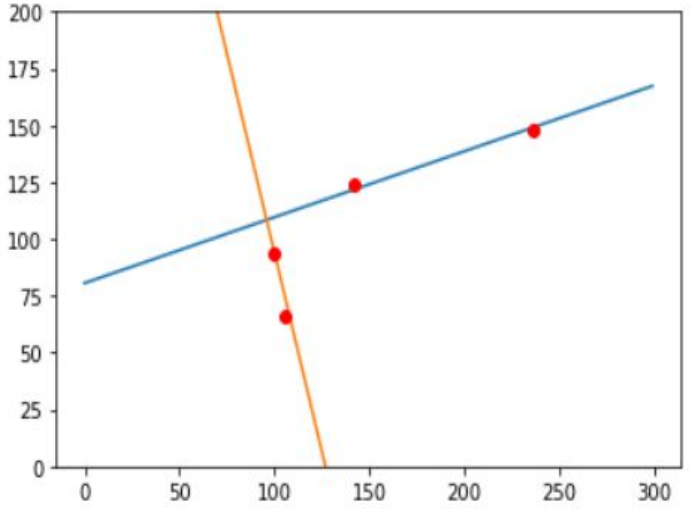

```
north puck: 20.84407733858566 inches per second
south puck: 53.05316775454142 inches per second
```



Finding Scattering Angle

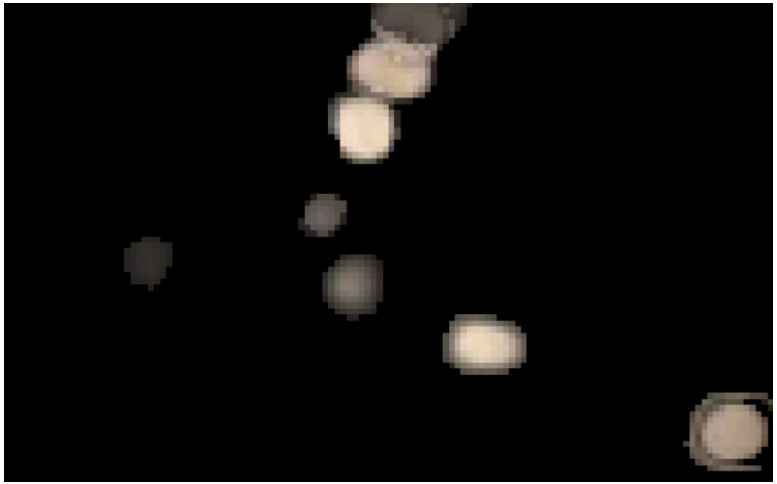
- Used the circle data that we got by using ‘Houghcircles’
- Draw a line between the pucks’ locations using the selected two frames

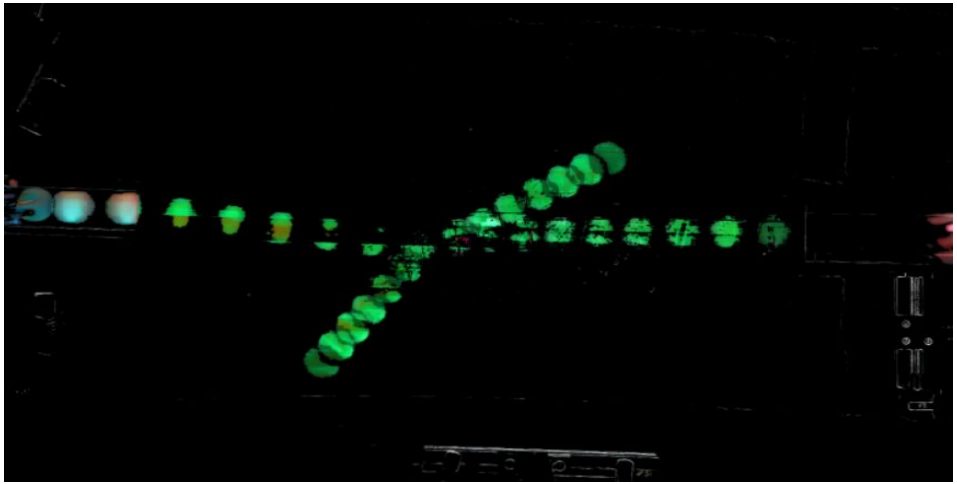




More Precision... More Advanced camera

We could use HoughLines which are more apt to find lines if we have more frames.

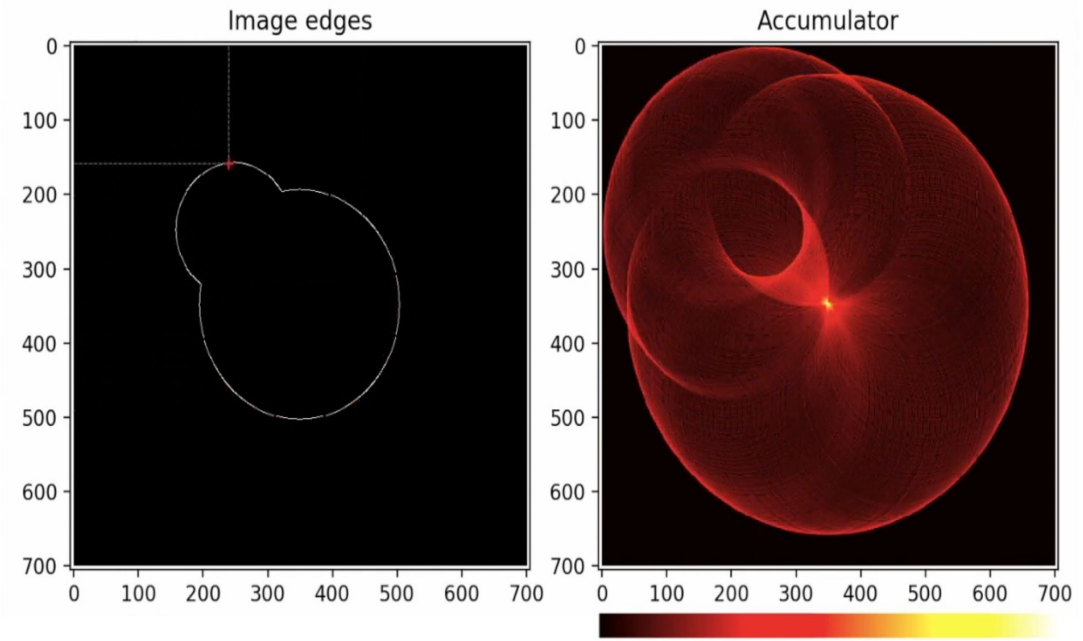


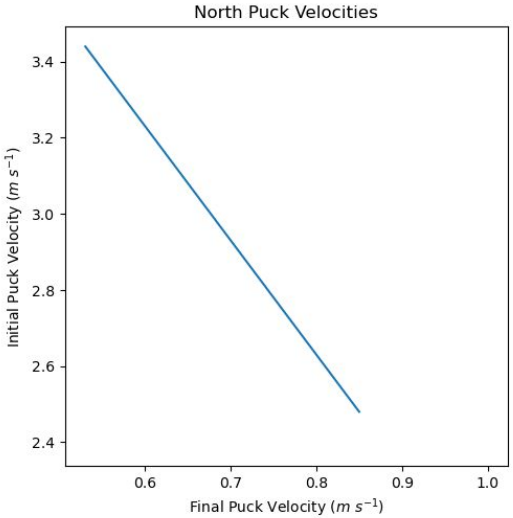


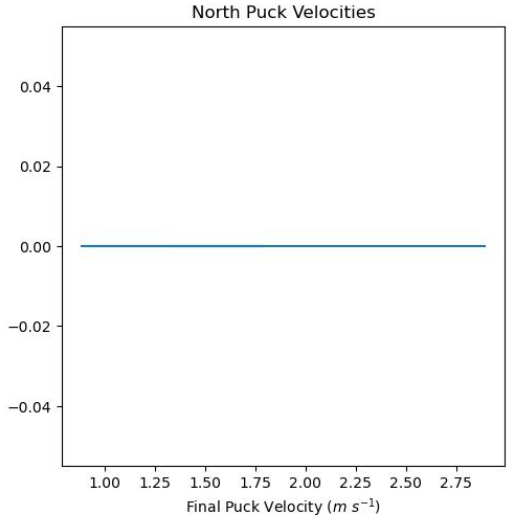
Houghcircles

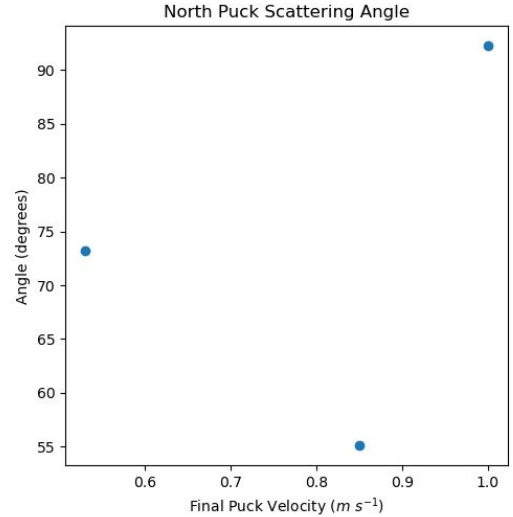
- CNN algorithm discussed by group 1 last class
- Draws circles of given radius range and chooses the points where the circles meet the most

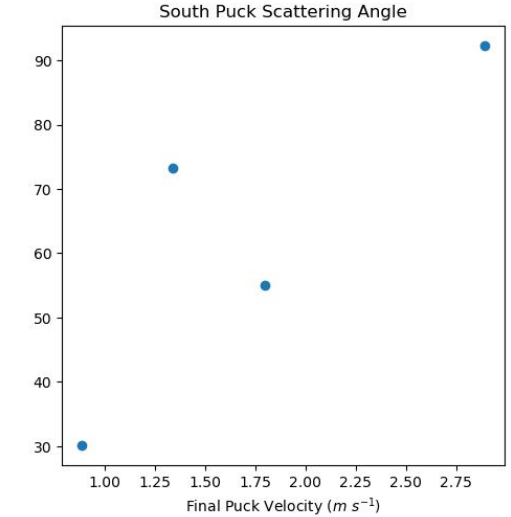
(Extra information: Houghlines first detect points and make edges with the given number of points threshold)



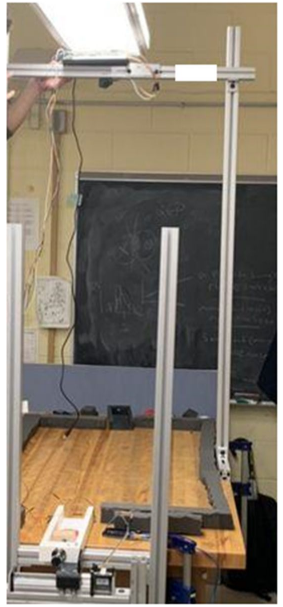

North Puck Velocities
Initial Puck Velocity ($m\ s^{-1}$)
Final Puck Velocity ($m\ s^{-1}$)
3.4
3.2
3.0
2.8
2.6
2.4
0.6
0.7
0.8
0.9
1.0


North Puck Velocities
Final Puck Velocity ($m\ s^{-1}$)
0.04
0.02
0.00
−0.02
−0.04
1.00
1.25
1.50
1.75
2.00
2.25
2.50
2.75


North Puck Scattering Angle
Angle (degrees)
Final Puck Velocity ($m\ s^{-1}$)
90
85
80
75
70
65
60
55
0.6
0.7
0.8
0.9
1.0


South Puck Scattering Angle
Final Puck Velocity ($m\ s^{-1}$)
90
80
70
60
50
40
30
1.00
1.25
1.50
1.75
2.00
2.25
2.50
2.75

Next steps



- For our data to be transmitted and processed with sufficient speed, the PCB has to be close to the camera
- Set up everything on the PCB and ensure that everything works well together
- 3D print second launcher and connect to pulley system for simultaneous launch
- Run multiple trials varying our parameter b.
- Fine tune data analysis and representation

Thank You