Portfolio 2: Particle Tracking Velocimetry for a bouncing ball.

Introduction:

To create an experiment to calculate the restitution coefficient of a bouncing ball. I chose to analyse the effect of changing the height on the bouncing of the ball to analyse the heights and the movement of the ball.

How the code works:

Firstly, the code starts by setting up the 'Video Reader' object in MATLAB which analyses the video frame by frame and it provides the videos framerate which can be used later in the code. After this a place for the results to be viewed needs to be set up so creating a figure window and an empty array for which the positions of the centroid will be stored. Next a while loop which continually blurs, greyscales and reduces the noise of the video by using a gaussian filter this makes the ball detection more robust. By using another object in MATLAB, the centroid of the ball can be discerned from each frame, and it is stored in the array. Finally, after all the frames have been processed MATLAB then produces a graph which follows the movement of the ball which you can take the height for the ball after each bounce and from this the restitution coefficient can be calculated.

Experiment design.

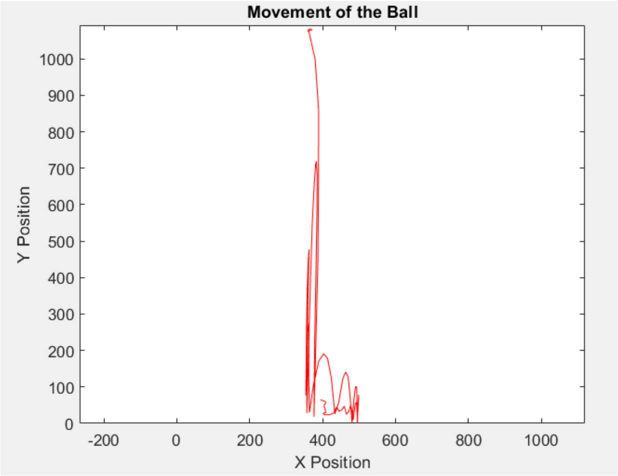
The ball was dropped from 3 different heights which were measured at 45cm, 30cm and 15cm respectively for videos pingpong1, pingpong2 and pingpong3. They were all bounced on a wooden table with a black background. To ensure the experiment is consistent I used the same ball and on the same wooden surface. And I also did 3 repeats from each height to make sure the bouncing on the table was in a consistent pattern and not erratically and random. The videos chosen had the most consistent bouncing without rolling away. I also recorded the videos in the same lighting so the particle tracking code could be consistent for each run. To calculate the restitution coefficient a simple equation can be used with the height values derived from the graphs.

$$E = \sqrt{\frac{h_F}{h_i}}$$

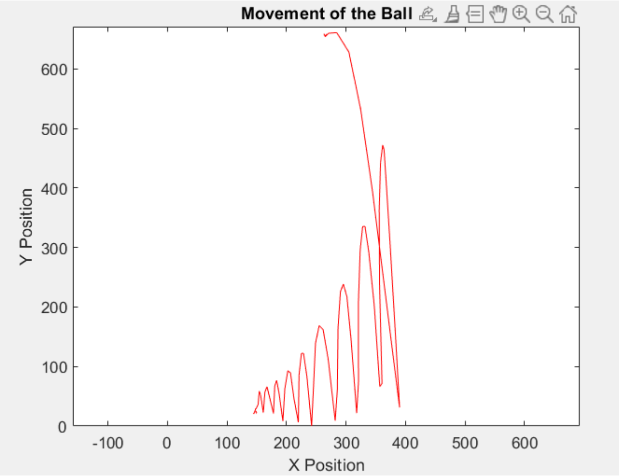
Restitution coefficient calculations

This is with the assumptions of conservation of energy such that air resistance and other losses are negligible

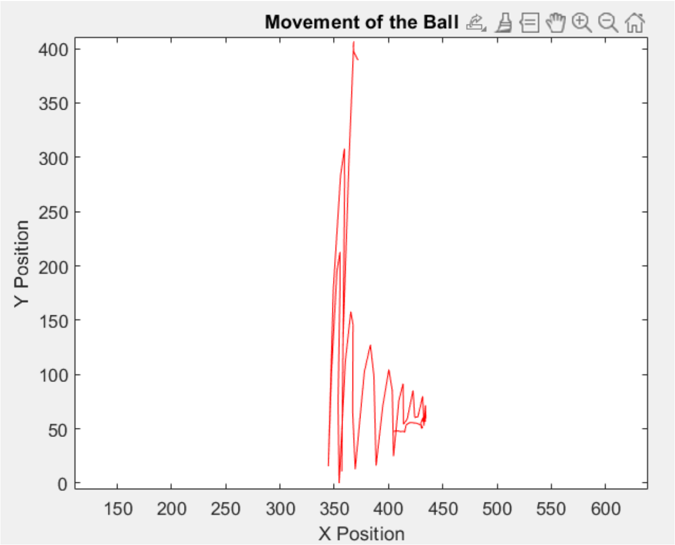
For drop 1 – the restitution coefficient is 0.817



For drop 2 – the restitution coefficient is 0.847



For drop 3 - the restitution coefficient is 0.861



Analysis of results

These results are quite accurate as I am expecting a close to 1 for my restitution coefficient. This is due to the fact that I am using a hard surface. Also, a ping pong ball is something which must be bouncy which explains the high restitution coefficient. However, the restitution coefficient should be slightly higher as some energy is lost as the trajectory is not perpendicular to the plane which it is bouncing on. Furthermore, in real life the ball will be spinning and there is air resistance which also will affect my coefficient.