

The implementation of Metaballs using Marching Squares

Zach Port-Lincoln — zport-lincoln.148020@student.uwtsd.ac.uk

Conclusion—

The project has the starting of metaballs in a 2D environment.
The metaballs bounce around the screen merging with all other balls that overlap.
Everything is rendered in only straight lines using the Marching Squares technique.

Introduction —

Here you will see how I have made the start of running Marching Squares in order to produce metaballs / Blobby objects.
Issues and results will be displayed visually where possible.

Aims and objectives—

The aim is to produce metaballs in a 2D environment

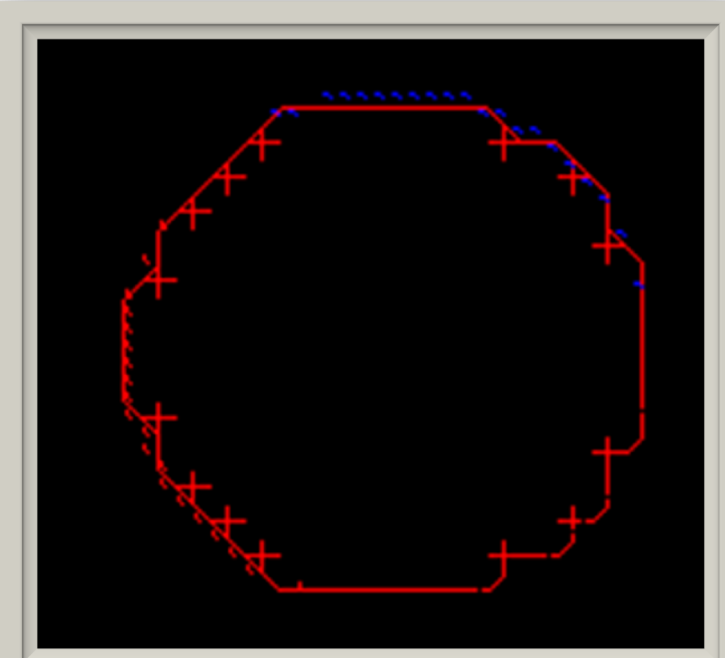
Objectives;

Set up Marching squares

Draw a circle using Marching squares

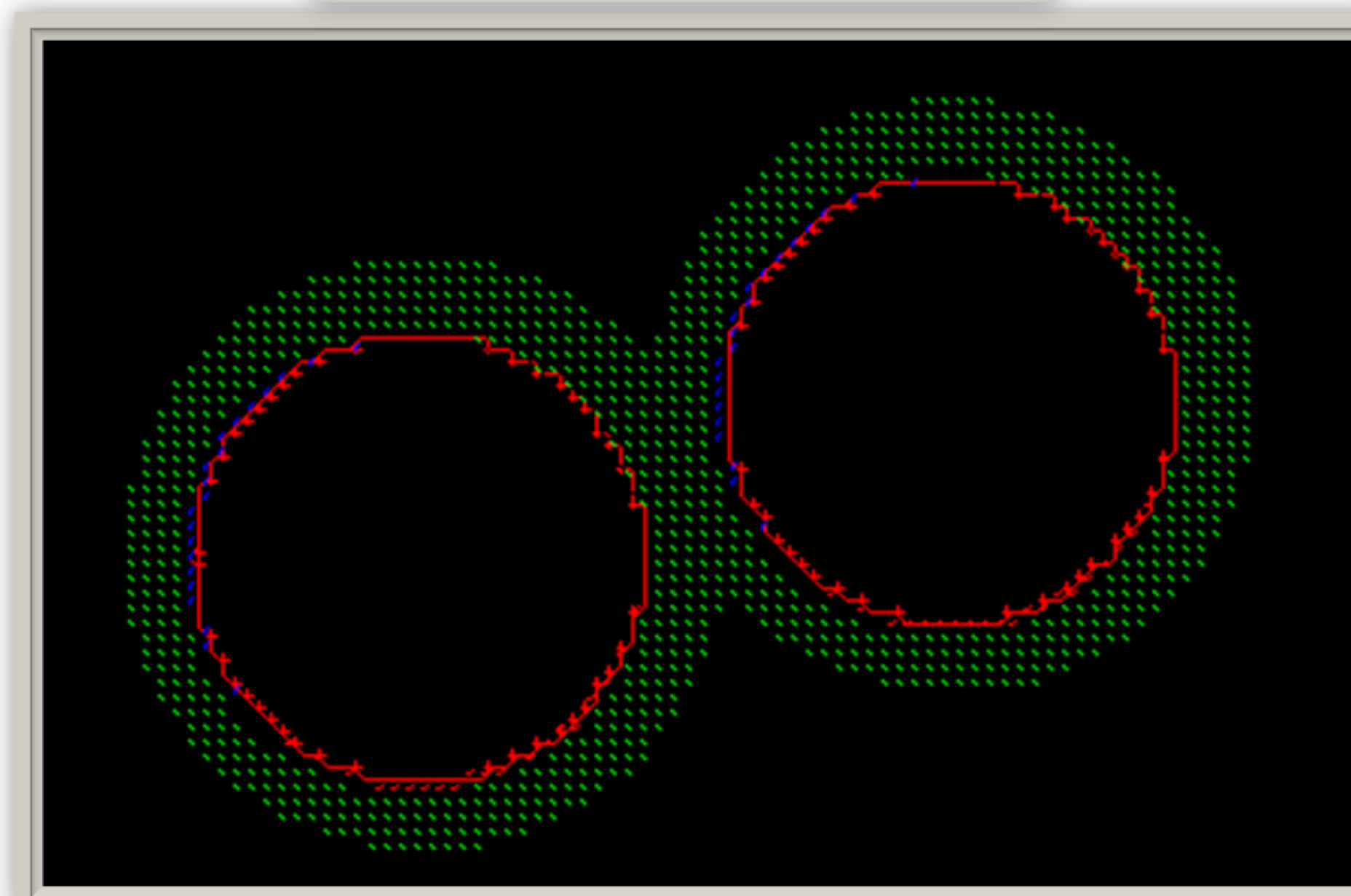
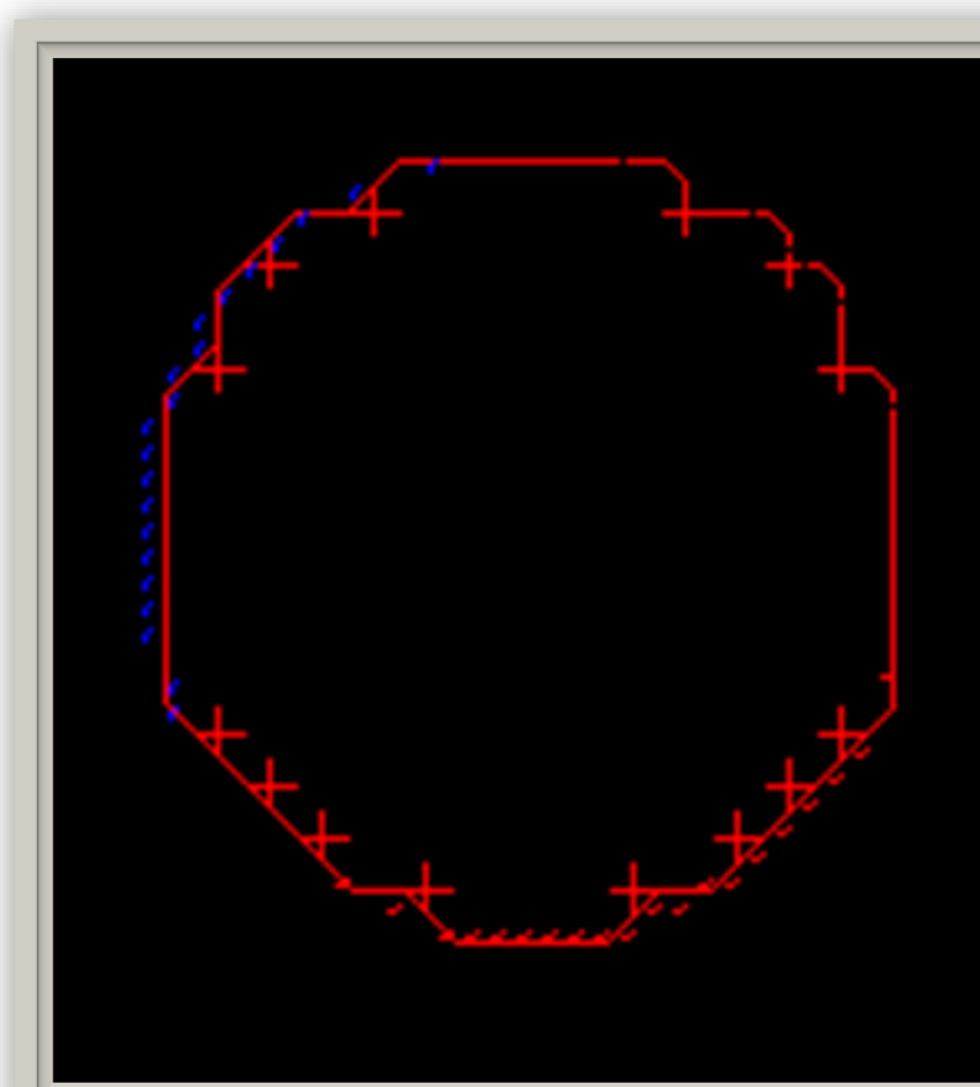
Animate circle

Introduce the metaball algorithm into the circles



Method—

The screen was split up into an invisible grid.
Circle was drawn using a radius of a circle's X and Y points.
Circle was then animated to bounce around the screen.
A starting attempt at the metaballs technique has been made.



Issues and Results—

One of the main issues was implementing the metaballs. Extra research was needed in order to fully implement the algorithm.
Another issue was that all the lines were being drawn regardless of if a diagonal line was being drawn first.

What needed to happen was on the diagonals were to be drawn and not the inner lines. This is visually represented on the left, the top image.

Influence radius was created as the start of implementing the metaballs. This is the green outline on the circles in the bottom picture.

User inputs were created to slow down the code. Pause the code and to randomise the velocities

