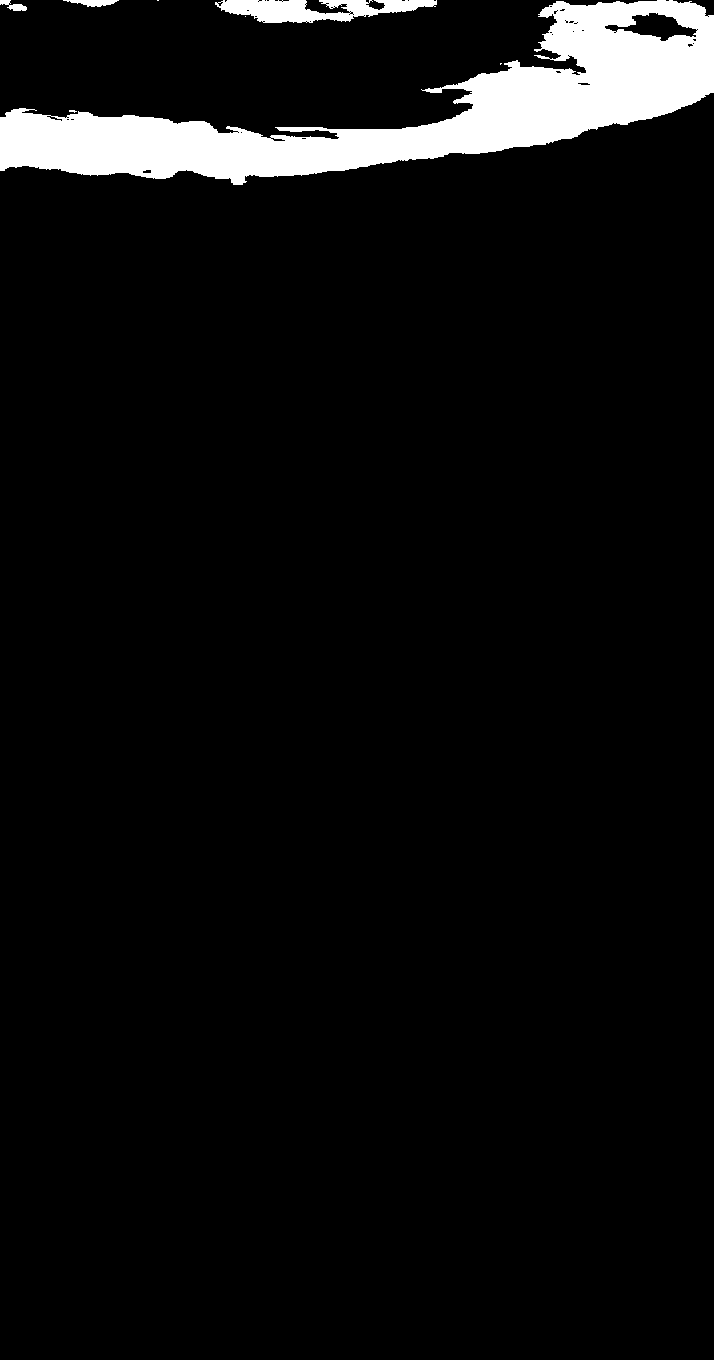
# Methods

We wanted to collect surface elevation data of the waves being produced in the wave tank to better understand its behavior and quantify its shape. We used image processing techniques in MATLAB to calculate the surface elevation of the water from 18 runs worth of images. Each run contains around 50,000 images that capture a 2D projection of a 3D water surface. We want to calculate the minimum water elevation in that projection. Below is an example of a picture that was taken and the elevation data that needs to be extracted from it:



*Figure 1: original image and the data we want (location of red line)*

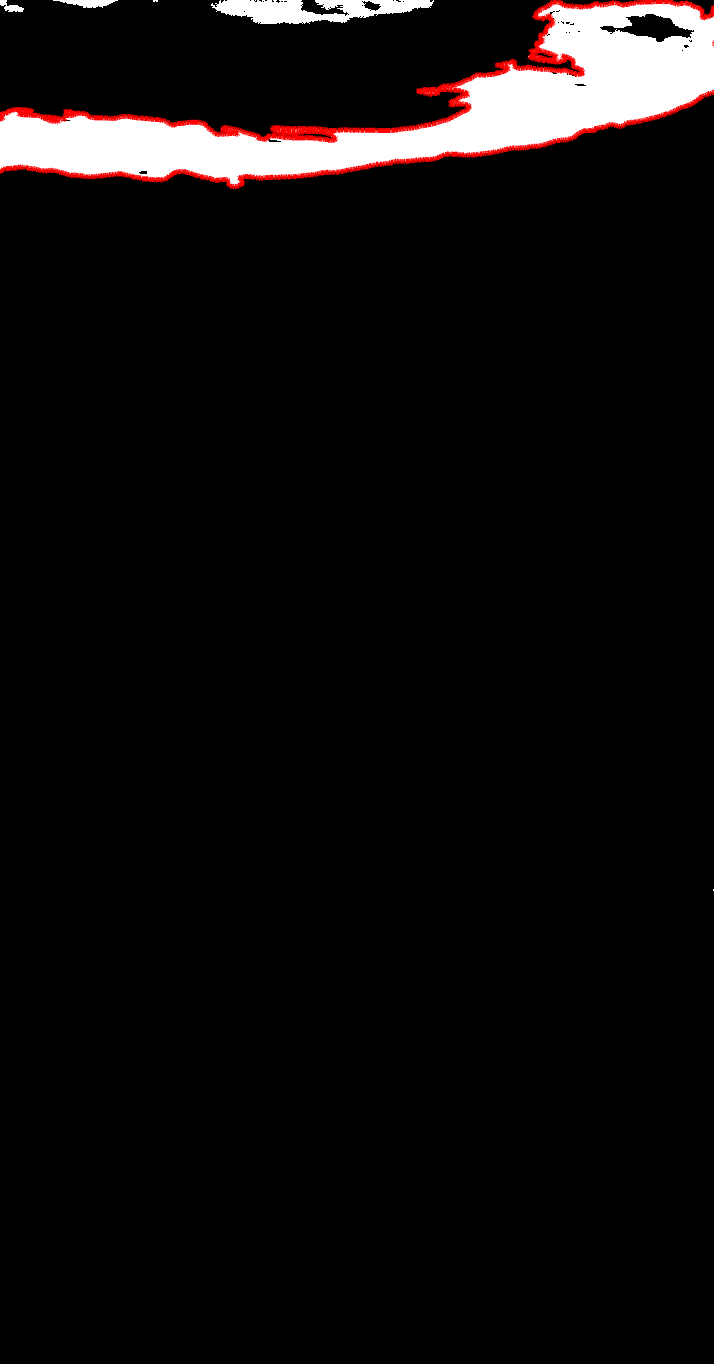
We binarized the images so the projection of the water surface was white and everything else was black. We binarized the images with an adaptive threshold function in MATLAB. This function determines the threshold for a pixel based on a small neighboring region. This was preferable because the images we were working with didn’t have uniform lighting conditions. The binarized images were eroded and dilated to remove noise. We also filled the holes in the binarized image to make identification of the water surface as easy as possible. Below is the binarized, eroded, dilated, and filled image:



*Figure 2: original image and final binarized image*

In some of the images, we found that the code was identifying the wrong contour altogether. We mitigated this by masking a portion of the image based on the elevation of the previous image. For example, if the highest elevation in a previous image is 48 pixels, we could mask 45 pixels of the current image so the program has a higher chance of identifying the correct contour. We used elevation information from a previous image because the surface elevation information is continuous in time.

After binarizing, eroding, dilating, filling, and masking the image, we calculated the contours (collection of all continuous points along the boundary of a white object). Several contours were identified, but only the largest contour was of interest to us because that contour contained the surface elevation data we wanted. Below is an image highlighting the largest contour:



*Figure 3: original image and outline of largest contour*

To calculate the minimum water elevation in the projection, we found the minimum value of the contour for every pixel in the width of the image. This ultimately gave us the surface elevation data we needed (shown below):

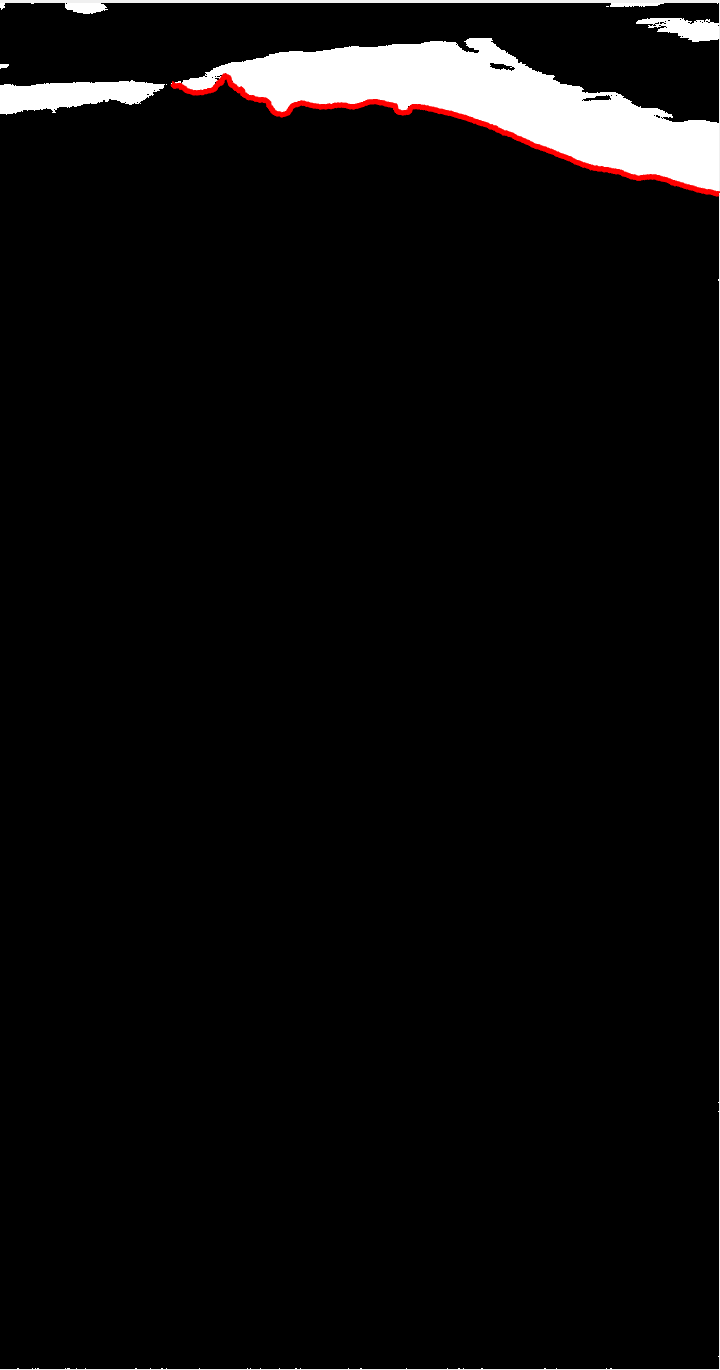
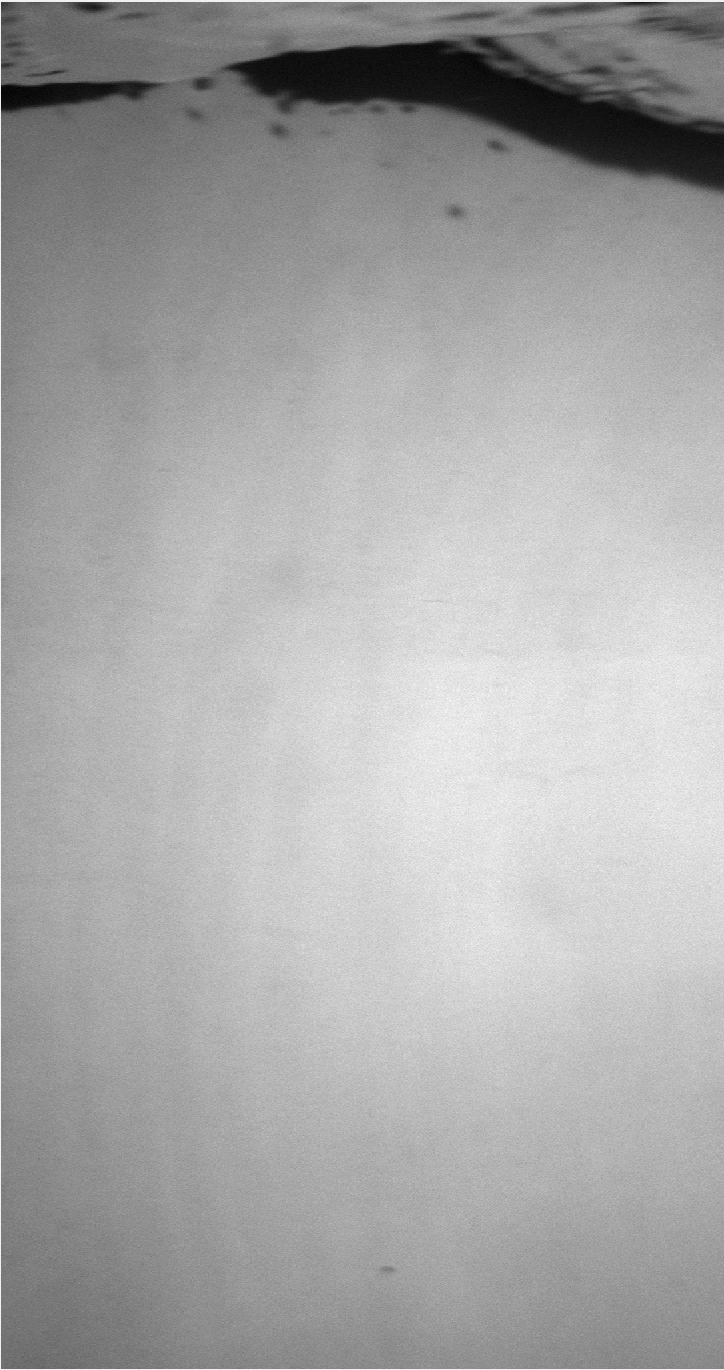


*Figure 4: original image and outline of the minimum elevation*

We collected the surface elevation data for every image in each of the runs. The surface elevation data for each of the 18 runs was saved as a separate matrix. Then, the data were smoothed with a moving average filter.

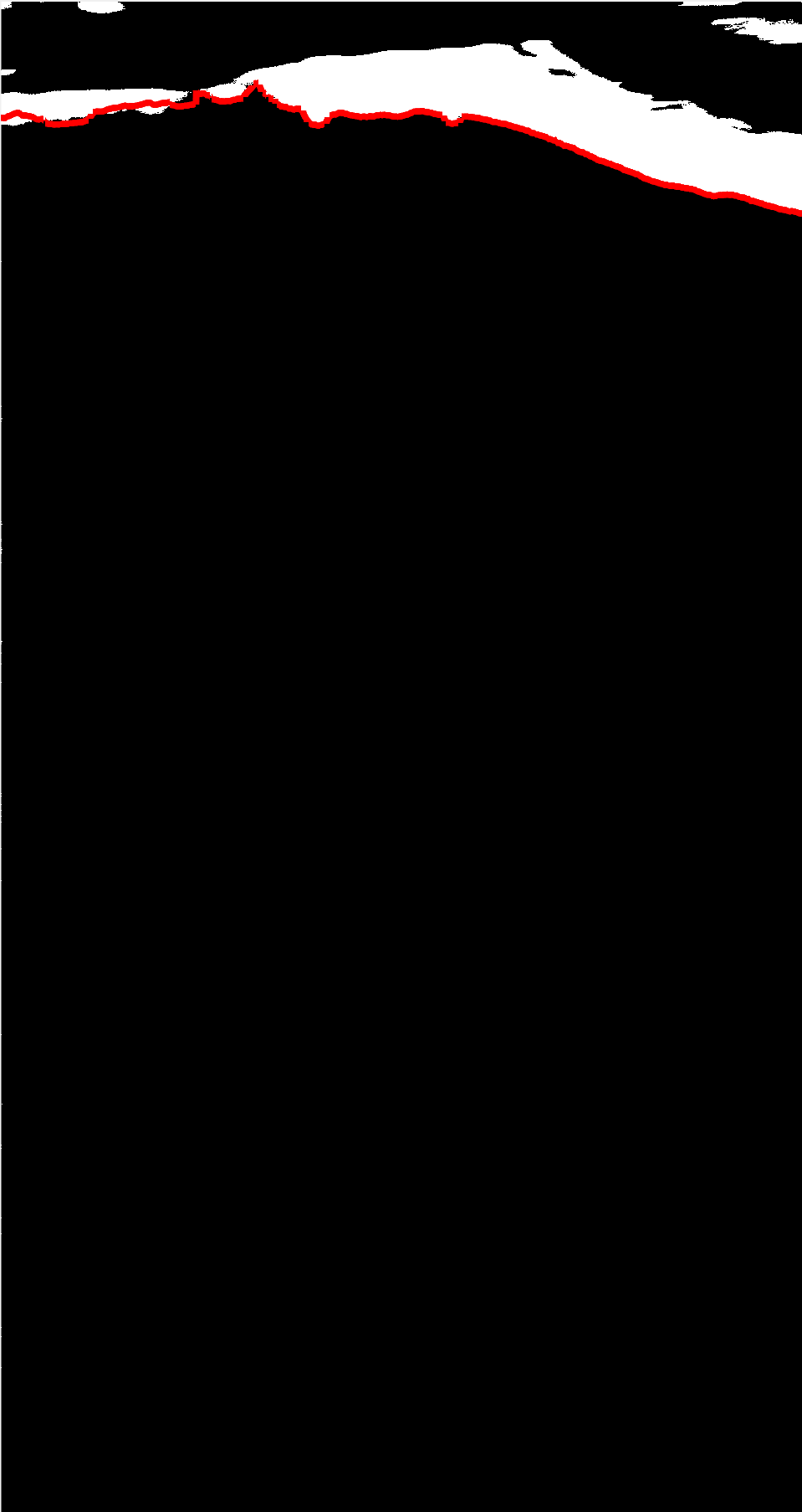
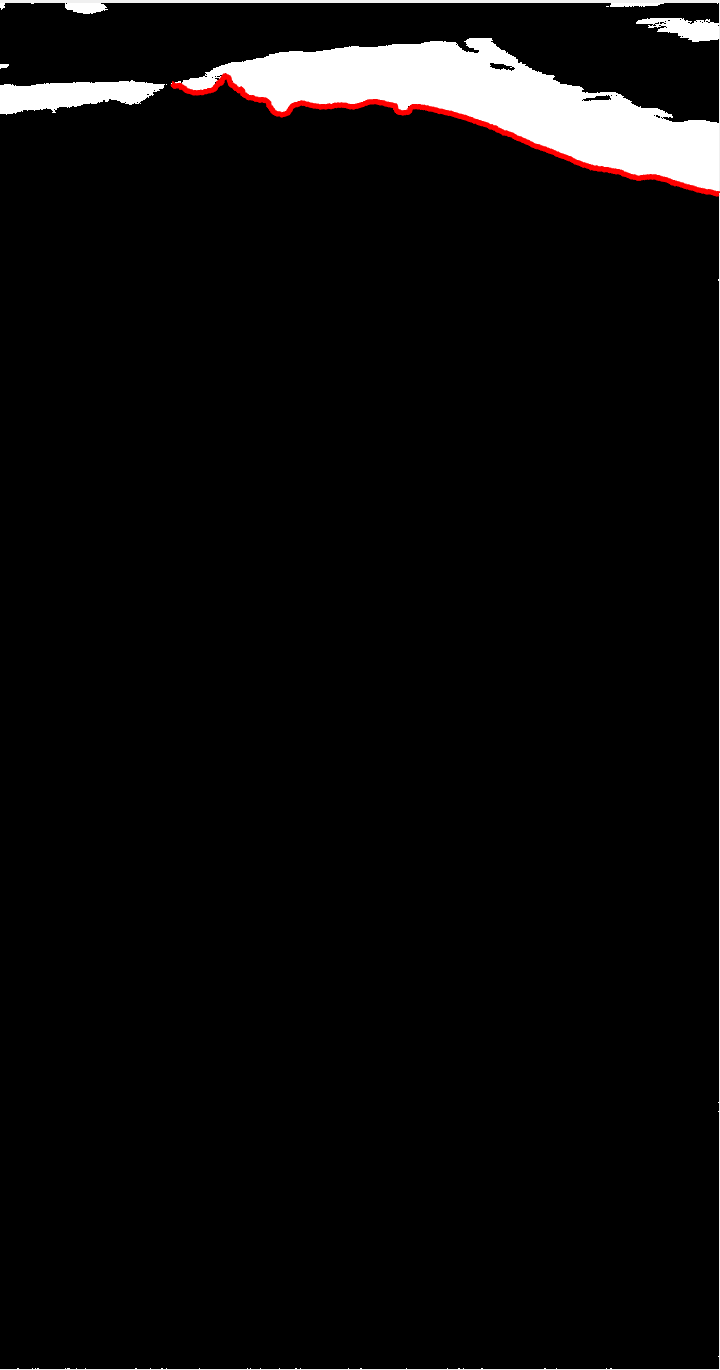
## Non-Detectable Surface Elevation

Many of the binarized images in our dataset did not have a clear white and black boundary, which meant we didn’t have elevation data for a chunk of the water surface. Below is an example of such an image:



*Figure 5: original image with elevation break and outline detected contour*

We fixed this issue after collecting all of our other data. The elevation data that could not be calculated initially were left as NaN values. As part of our post-processing steps, we calculated these NaN values via interpolation and extrapolation in time. Interpolated values were calculated using a shape-preserving piecewise cubic function. Extrapolated values were calculated by using nearby data points. Below is the same image, but with interpolated and extrapolated values:



*Figure 6: original elevation data and final elevation data after post-processing steps*

## Calculating Wave Statistics

After we had all the data, we calculated some wave statistics to quantify the artificial waves in our wave tank. We specifically calculated the [...]