

# A Computer Vision and AI Based Solution to Determine the Change in Water Level in Stream

Priyanjani Chandra\* – Pratoool Bharti\* (Advisor) – Michael E. Papka\*† (Advisor)

\*Northern Illinois University †Argonne National Laboratory

## Abstract

In this project, we design and implement a computer vision and AI based solution to automatically detect the water level in the stream. We employ template matching algorithms of edge map images to find the water level coordinates. Finally, we trained a linear regression based model on these coordinates to achieve a straight line that represents the water level. We achieved 0.949 R<sup>2</sup> score when compared to the ground truth for 200 images across several days.

## Motivation

- Detecting real-time water level in rivers or creeks located in urban areas is crucial, as floods are being a constant threat to the human community living
- Existing solutions are neither cost-effective nor automated and require high maintenance
- We want to provide a more reliable and automated solution
- We want to deploy it as an edge computing solution to provide a real-time alert system for high water level

## Resources

- Array of Things (AoT) Node
  - Sensors – 2 5MP cameras, environmental, air quality, and light and infrared sensors
  - Two computers – 4 cores and 8 cores, 1 GPU and 1 microcontroller board
- Hartley – A resource of ddiLab at NIU
  - 16 CPU cores, 2 GPUs, 512 GB RAM



Figure 1: AoT Node installed on a light pole in front of the Computer Science building at NIU

## Data

- Images are collected from AoT node and are stored in Hartley for research purposes.
- Frequency – 1 Hz.
- Resolution – 96 dpi; Size – 2592 x 1944.
- 2,580,468 images collected in 2019.
- Collected daily from June 29, 2020.



Figure 2: A picture taken by the downward facing camera of the AoT node.

## Problem Definition

- Let **R** is the reference image in which the height of water level  $h_r$  is pre-calculated.
- For a new input image **I**, our goal is to find the difference in height of water level  $\Delta h$ , between **R** and **I**, such that we can compute the height of water  $h_i$  level in **I** using  $h_i = h_r + \Delta h$ .
- $h_r$  is a constant.  $h_i$  and  $\Delta h$  are algebraic variables in the range of  $(-\infty, +\infty)$ .

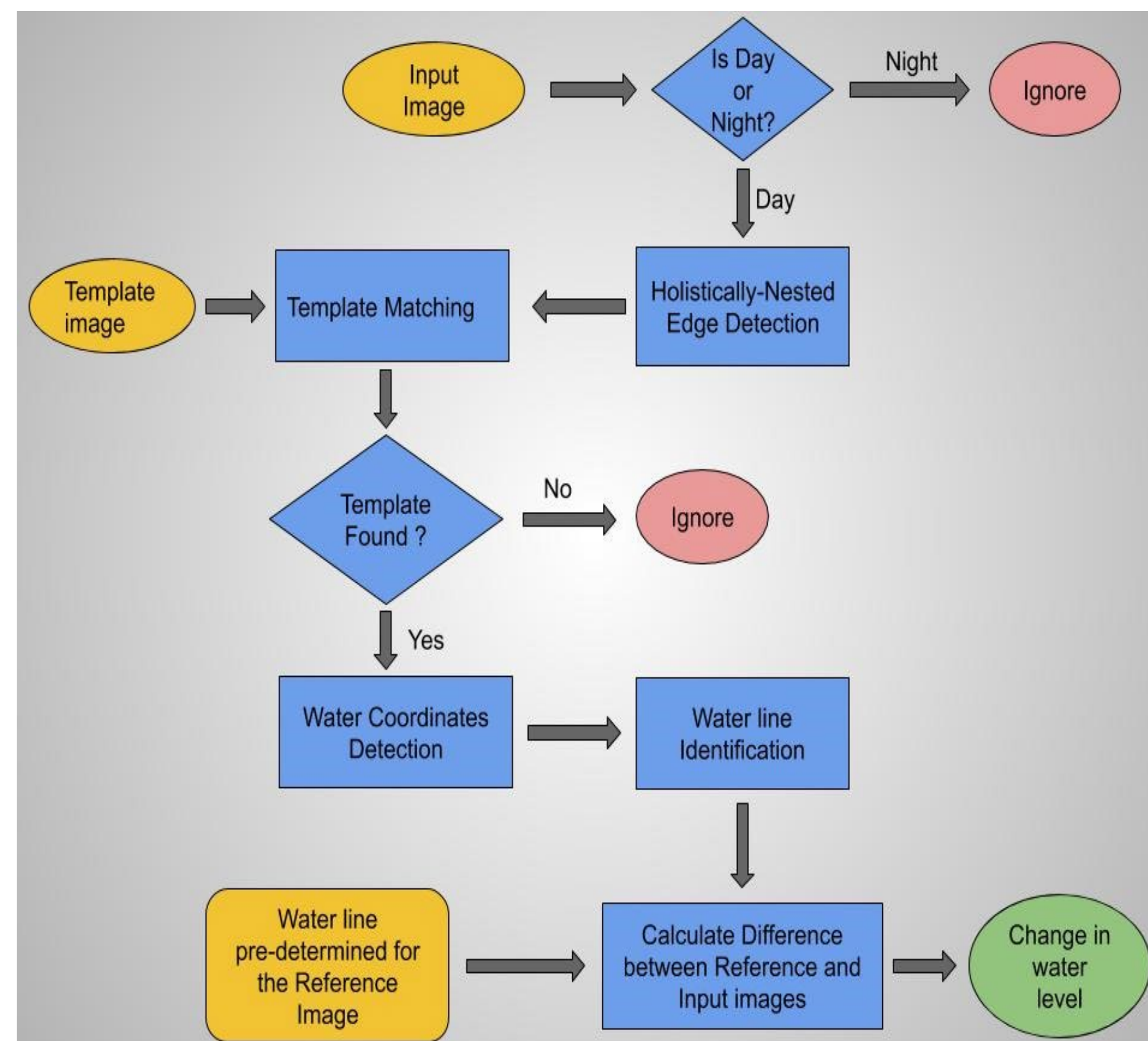


Figure 3: Water Level Detection Algorithm's detailed approach described in a flowchart.

## Edge Detection

- The Input image is processed through Holistically-Nested Edge Detection (HED)
- HED takes the RGB image as an input and provides rich edges of the image.
- It implements multi-scale feature learning and image-to-image classification based on fully convolutional neural networks and deep layer supervision.

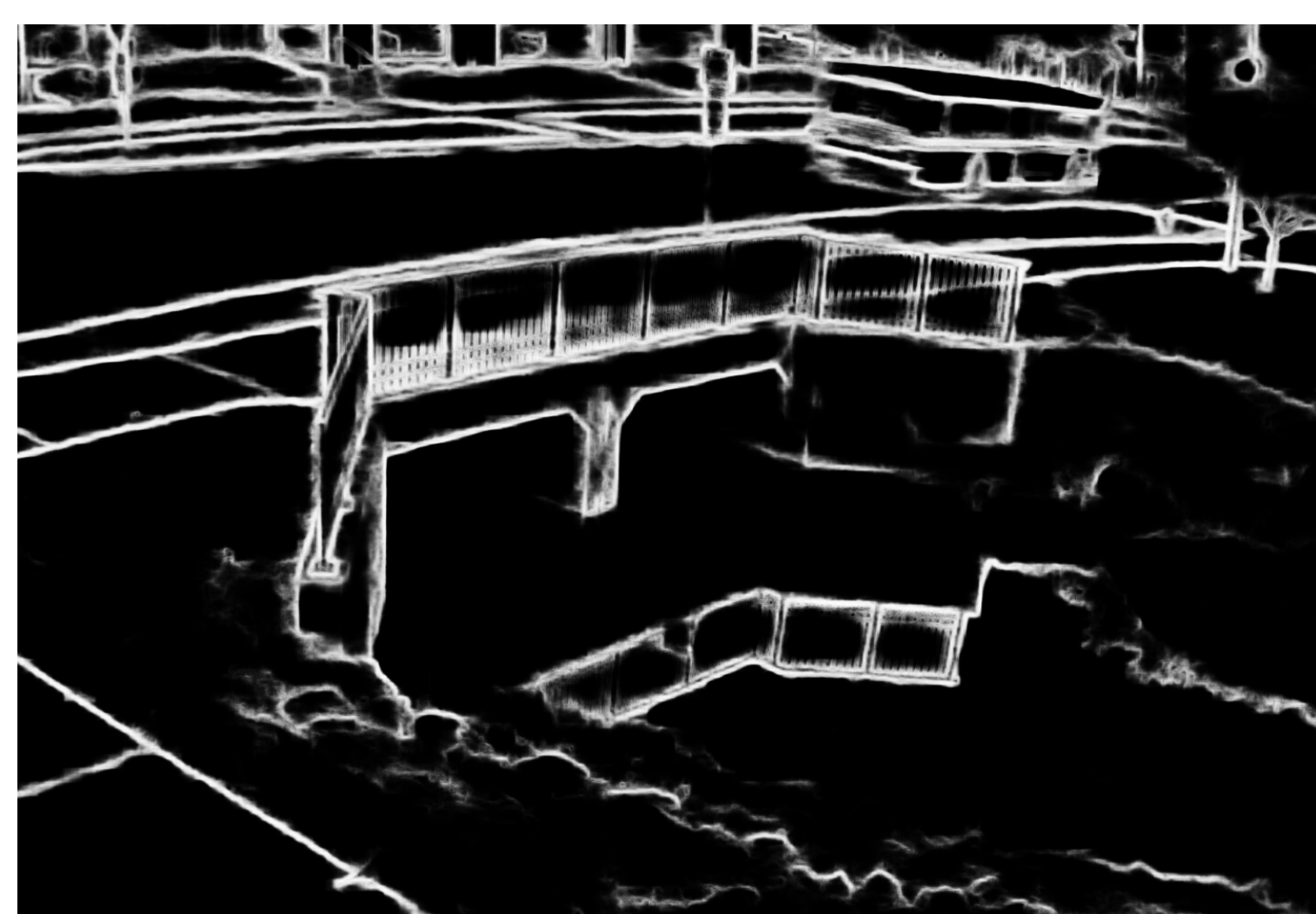


Figure 4: Input image after application of HED

## Template Matching

- Required to identify the bridge where it meets water.
- Used template matching technique provided by OpenCV
- Region of interest in the HED image is identified by this technique.
- Normed square difference and normed cross correlation are the two metrics used in matching.

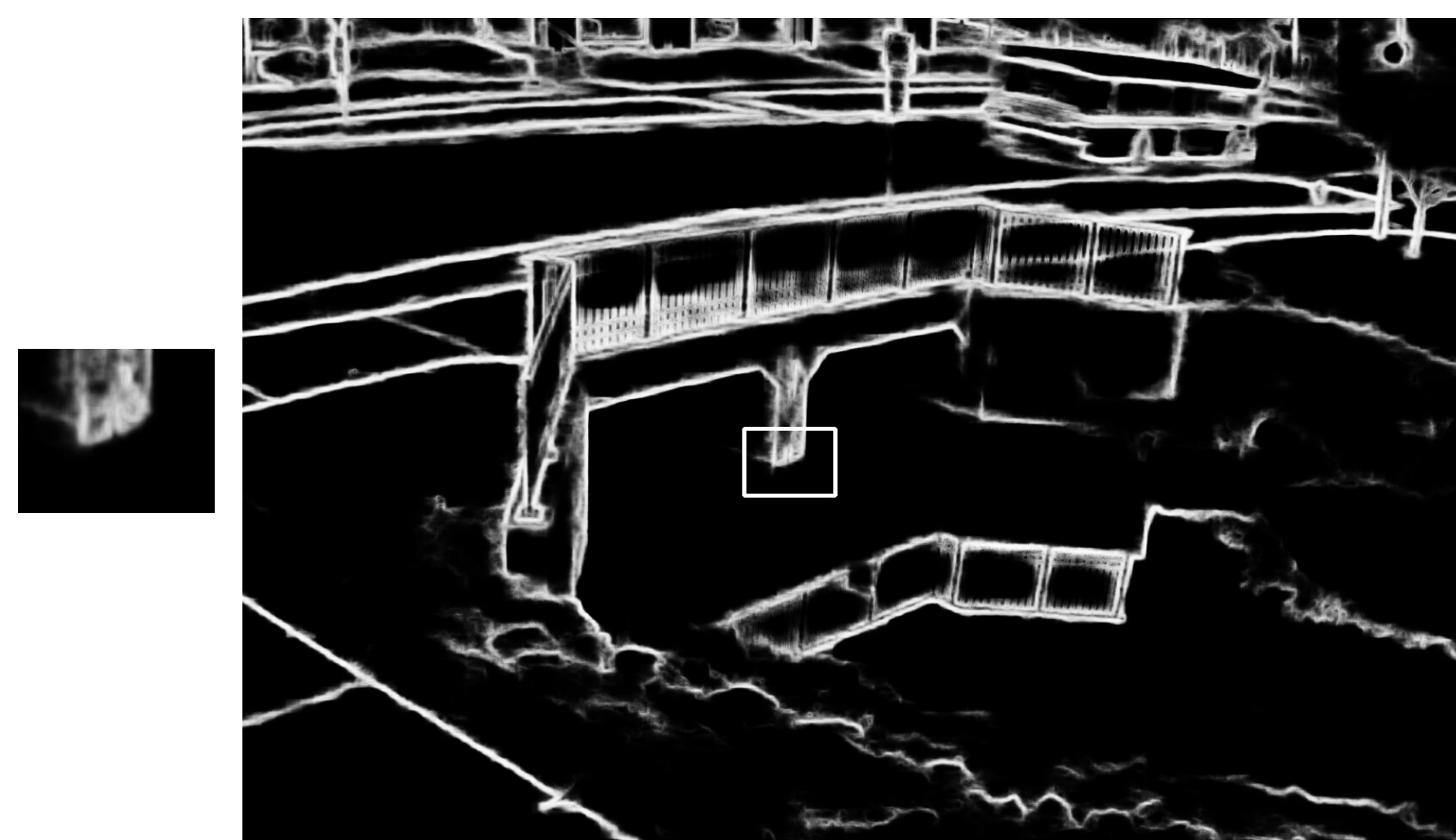


Figure 5: Template Image taken from the reference image and the match in the input image

## Water Coordinates and Water Line Detection

- Water coordinates are detected in the image based on the large variation observed in the pixel values of the water and bridge locations in the image.
- A reference water line is obtained using Linear Regression for the reference image.
- Sliding windows with 50% overlap with 10 coordinates in each window are considered for detecting the water line.
- One window which has the minimum average distance from it's coordinates to the line formed by slope and intercept of the reference line is selected.
- The difference between the reference line and the line of the selected window in the images is the observed change in water level.

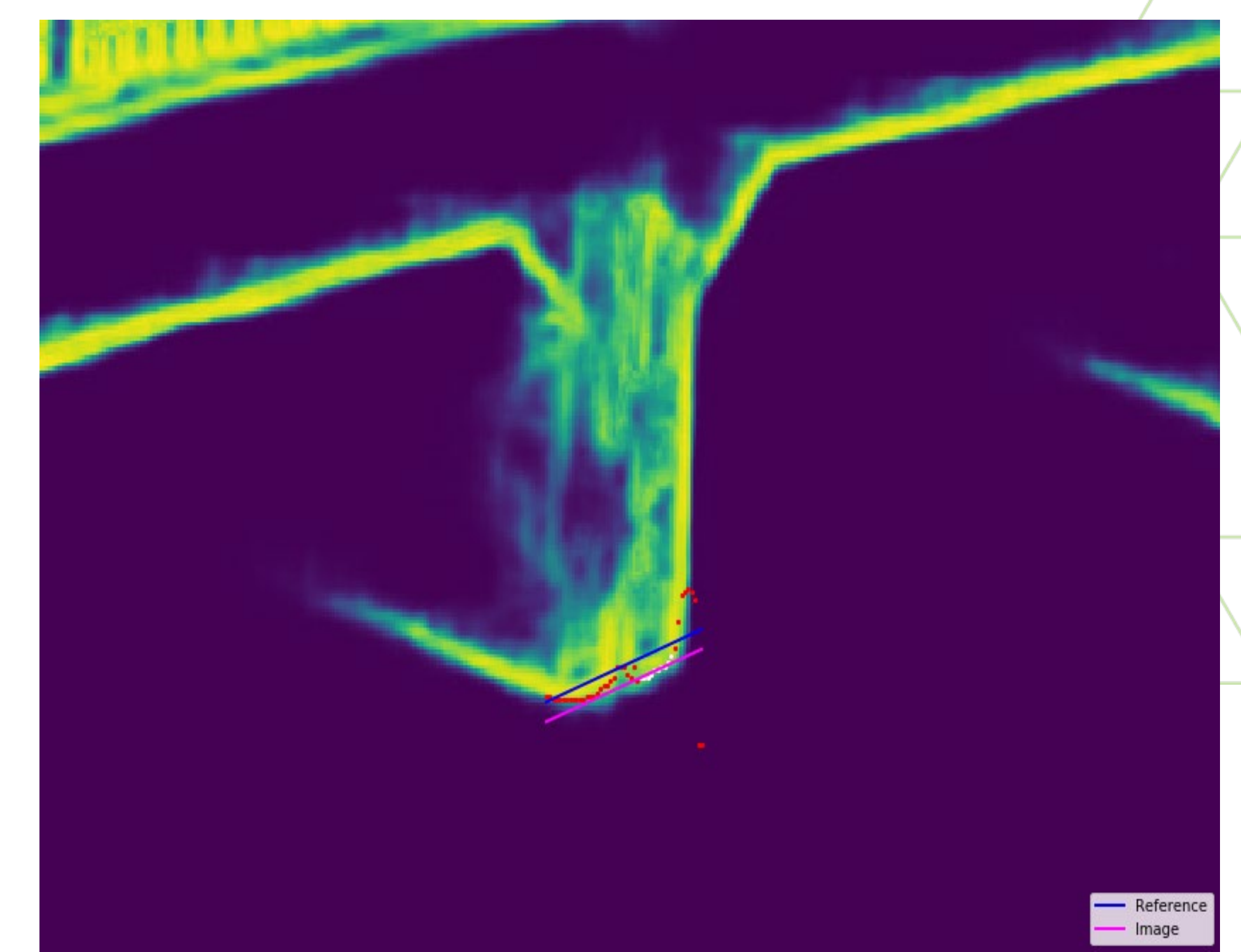
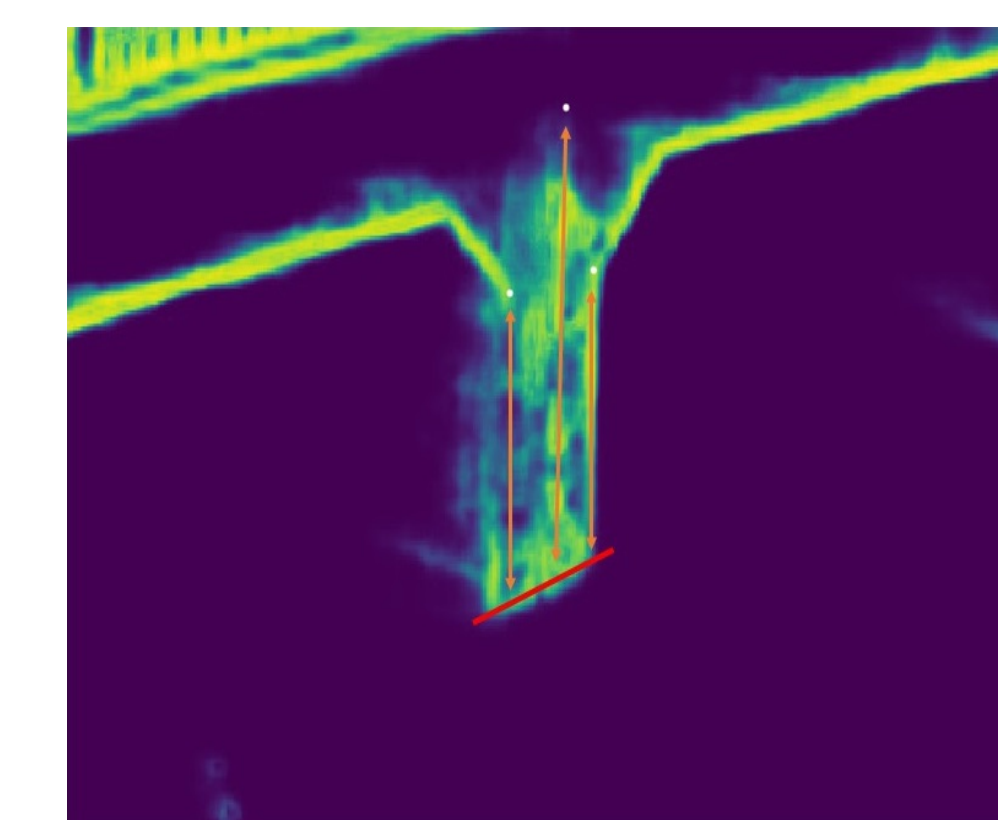


Figure 6: Water coordinates detected, and water line identified for the input image based on the reference water line

## Performance

- 200 images were selected over a period of 50 days and ground truth is calculated using the images.
- The height of the bridge is calculated using the algorithm and compared the results.
- We achieved high correlation with the ground truth for 200 images.
- The results from the algorithm, for 68,890 images from 96 days, are following the trend of the water level data provided by the United States Geographical Survey (USGS) for the south branch Kishwaukee river in DeKalb, IL.



Metrics	Value
R <sup>2</sup> Score	0.949
Root mean squared error	4.3247
Mean absolute error	3.5091

Figure 7: Image (left) shows the height of bridge calculated using the algorithm. Table (right) shows the metrics for correlation.

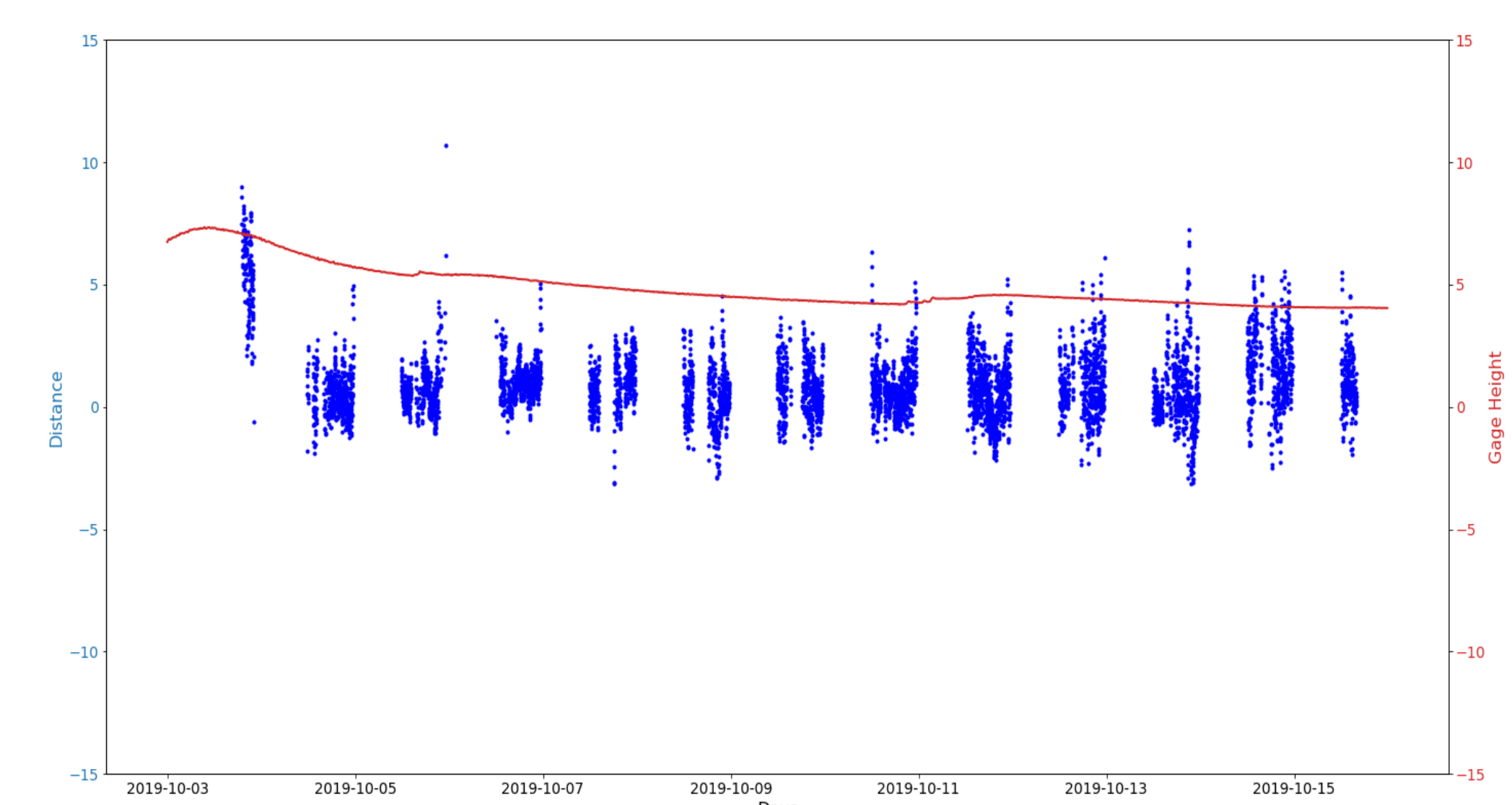


Figure 8: Graph showing the correlation between the algorithm output and USGS data during October 2019