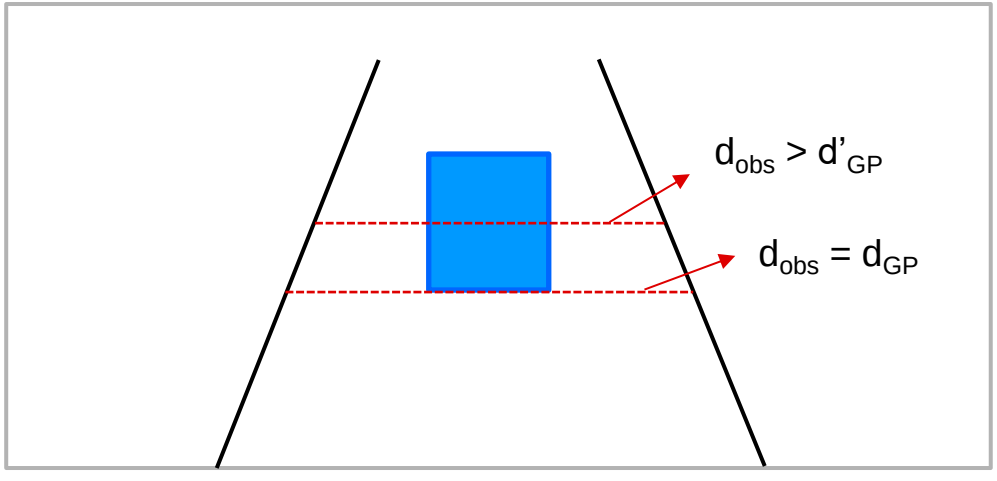
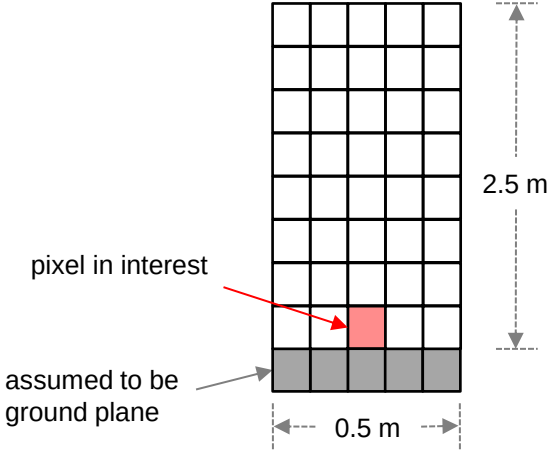
Obstacle / Free Space Detection (1)

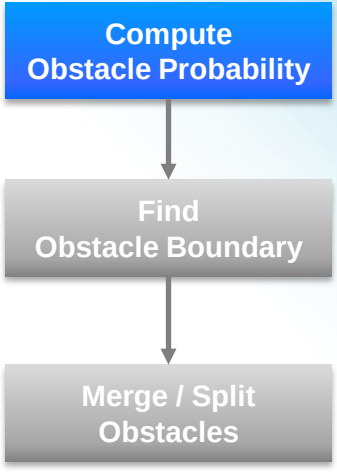
- **Applied on top of GP estimation**
- **Assumptions for obstacle / free space detection**
  - Obstacle has disparity larger than or equal to the disparity of GP it stands on
  - Free space has disparity smaller than or equal to the disparity of an object which stands on that free space



Obstacle / Free Space Detection (2)

- **Compute a probability for every pixel to be an obstacle**
  - Count the probable obstacle pixels, which has larger disparity than ground, in a window
    - Window corresponds to 2.5 m height x 0.5 m width
    - Window size in pixels depends on the depth of a pixel in interest



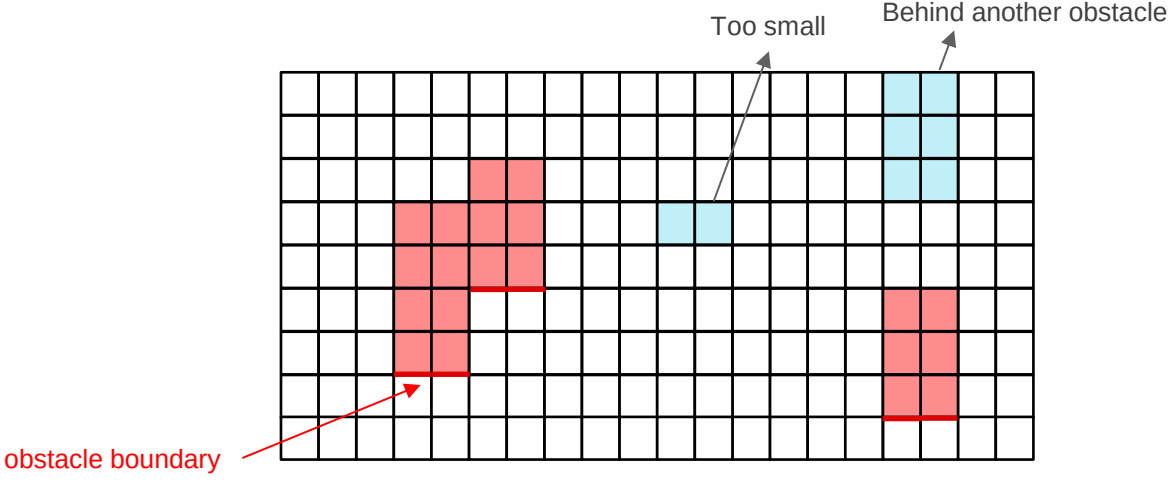


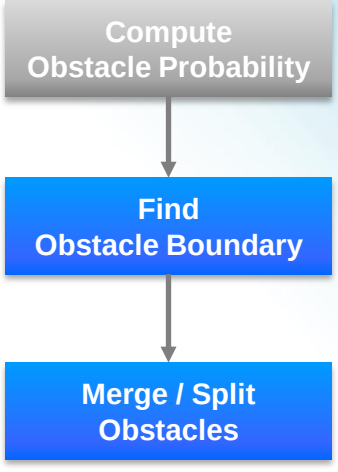
- Obstacle detection -

  - Probability for a pixel to be obstacle, $Pr(obstacle) = \frac{\#\ of\ obstacle\ pixels}{\#\ of\ pixels\ in\ a\ window}$

Obstacle / Free Space Detection (3)

- **Find the closest obstacle boundary in every column**
  - Obstacle should have *n* (> threshold) consecutive probable pixels in a column
  - A starting pixel is an obstacle boundary
- **Merge and split obstacles**
  - Merge obstacles in neighboring columns
  - Split an obstacles based on perspective, etc.





- Obstacle detection -

Obstacle / Free Space Detection (4)

- **Find the closest obstacle boundary in every column**
  - Obstacle should have *n* (> threshold) consecutive probable pixels in a column
  - A starting pixel is an obstacle boundary
- **Merge and split obstacles**
  - Merge obstacles in neighboring columns
  - Split an obstacles based on perspective, etc.