



CLIFF DETECTION VIA DEPTH ANYTHING V2

SHIZHEN YANG

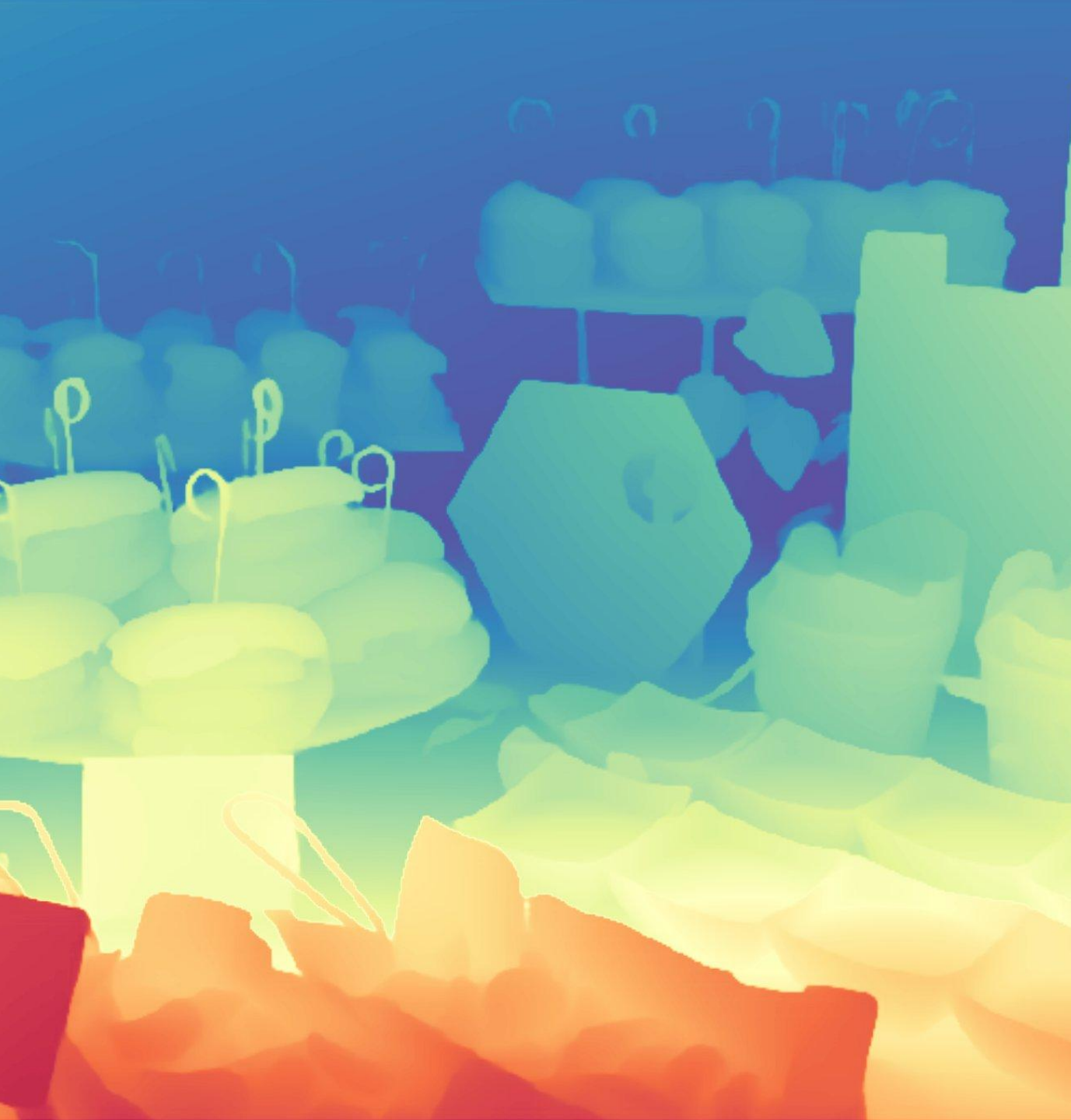
OUTLINE

- Project Motivation
- Depth Anything
- Second-Order Gradient
- Hough Transform
- World Coordinate Projection
- Result
- Future Work

PROJECT MOTIVATION

When we attempt to let the VEX-AIM robot navigate autonomously, it often drives off the edge of the table and crashes. This happens because the robot lacks built-in depth sensors like LIDAR, and the RGB images captured by its camera alone are insufficient to detect sudden depth changes. As a result, the robot is unable to perceive important obstacles, such as wires or the edge of the table. Without reliable depth awareness, the path planner may unknowingly direct the robot over a cliff, leading to damage and frustration. To prevent such incidents and protect both the robot and the students' efforts, we need a method that allows the robot to detect cliffs and avoid driving off the table — making both the robot and the students happier.

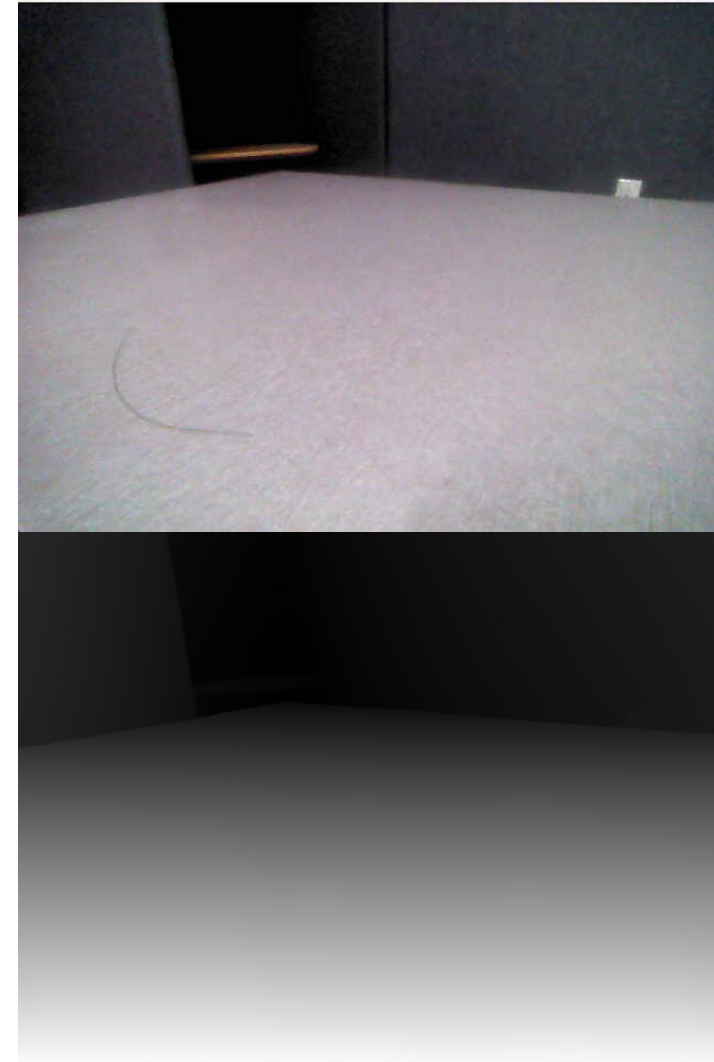


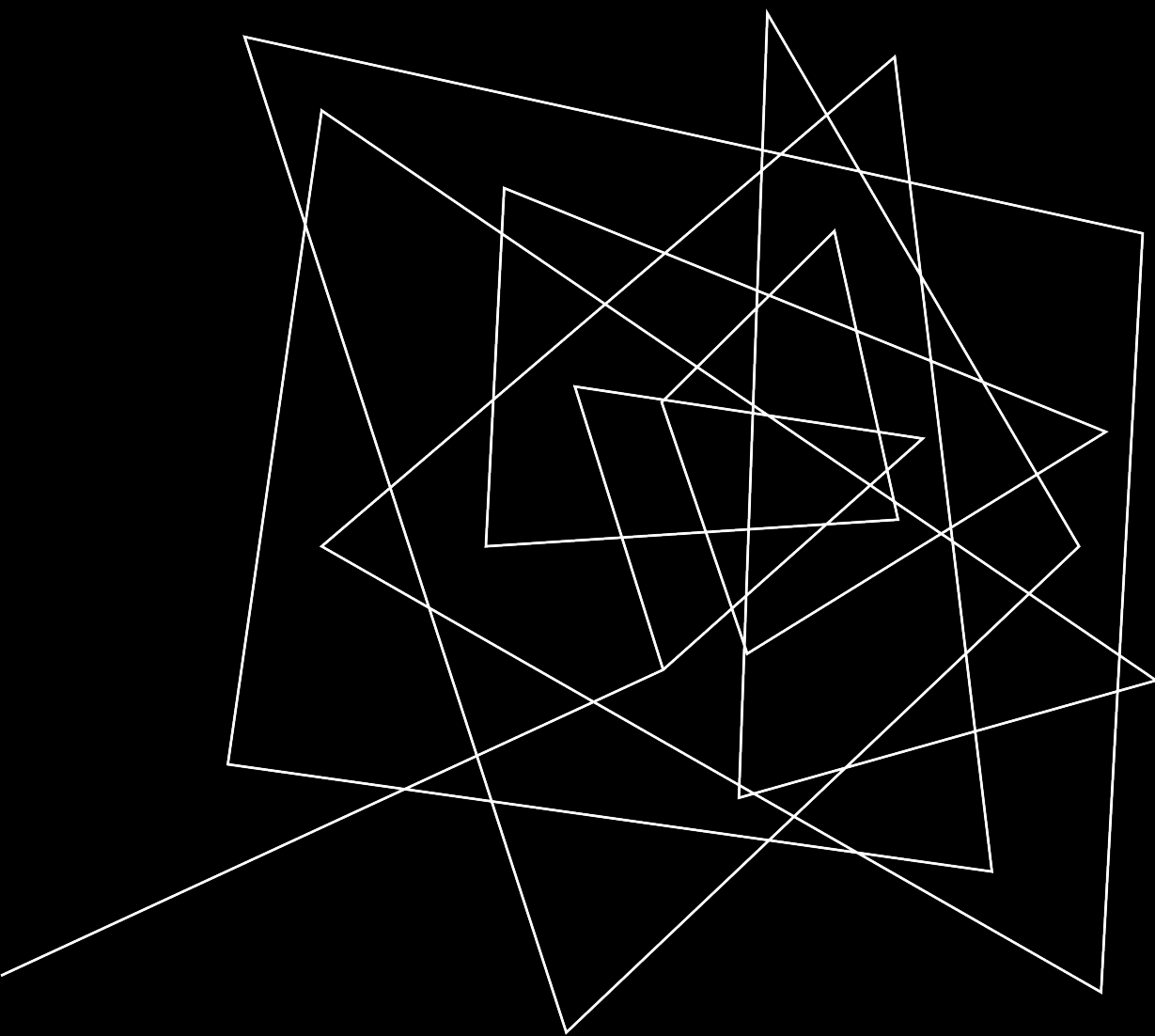


DEPTHANYTHING V2

DEPTHANYTHING V2

Depth Anything V2 is a powerful monocular depth estimation model capable of generating detailed depth maps from a single RGB image. Trained on a large-scale dataset combining synthetic and real-world images, it provides accurate and generalizable depth predictions across diverse environments. In our project, we use Depth Anything V2 to convert the RGB image captured by the robot's onboard camera into a depth map, enabling the robot to perceive three-dimensional structure from a single view. This depth map serves as the foundation for further processing, allowing the robot to understand the geometry of its surroundings without relying on dedicated depth sensors like LIDAR.

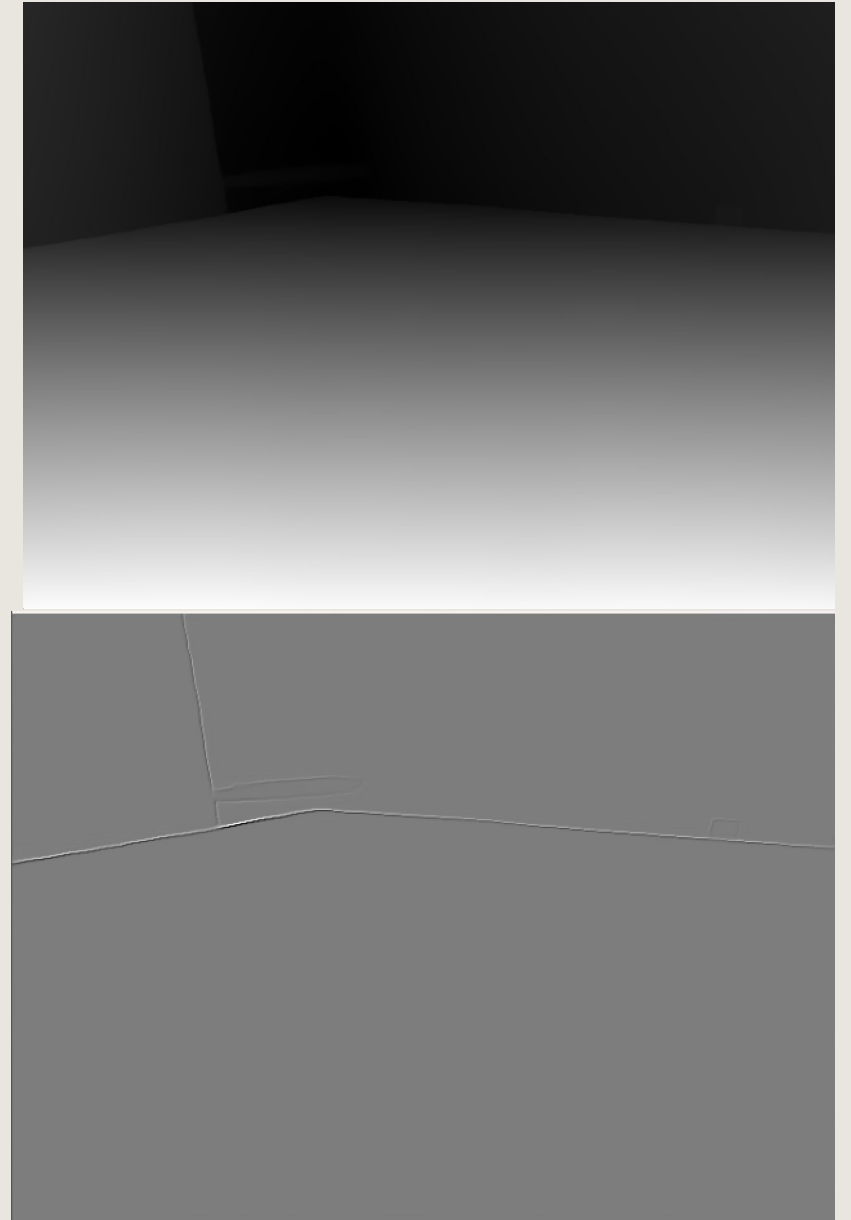




**SECOND-ORDER
GRADIENT**

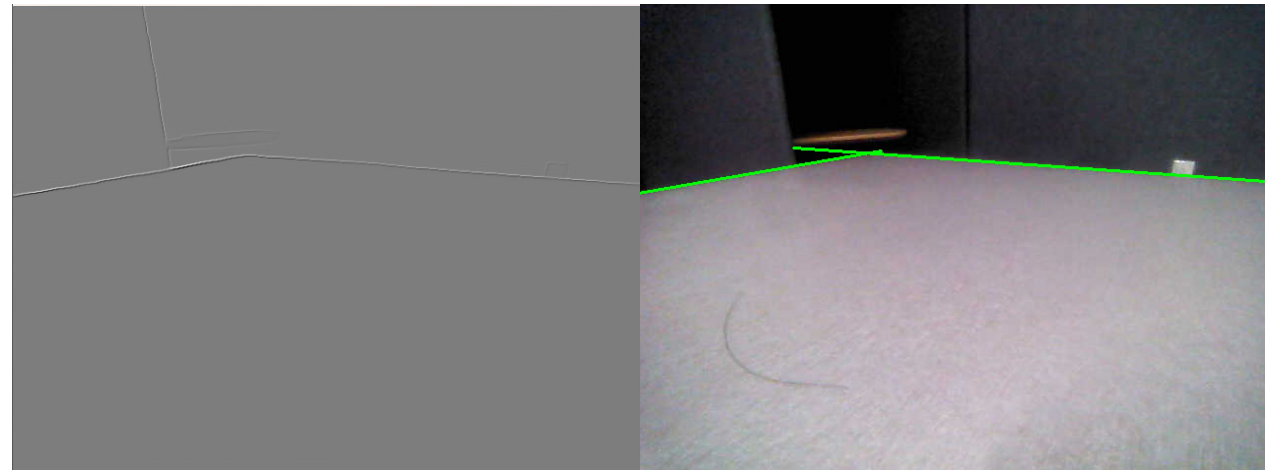
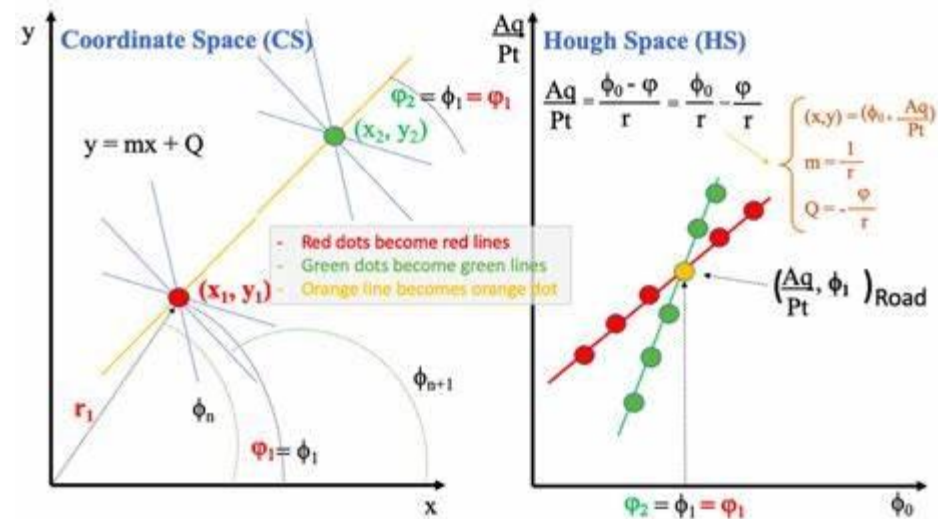
SECOND-ORDER GRADIENT

In image processing, a gradient refers to the rate of change in pixel intensity across space, often used to identify edges or boundaries within an image. In the context of depth maps, the gradient highlights regions where the depth value changes rapidly—such as the transition from a tabletop to empty space at the edge. However, raw gradients can still include gradual slopes or surface texture changes that are not actual environmental boundaries. To sharpen our focus on abrupt structural changes, we compute the second-order gradient, which measures how quickly the gradient itself changes. This enhances high-contrast transitions while suppressing smooth variations, making it especially effective at isolating sharp edges like cliffs, drop-offs, or the side of a table in a depth image.



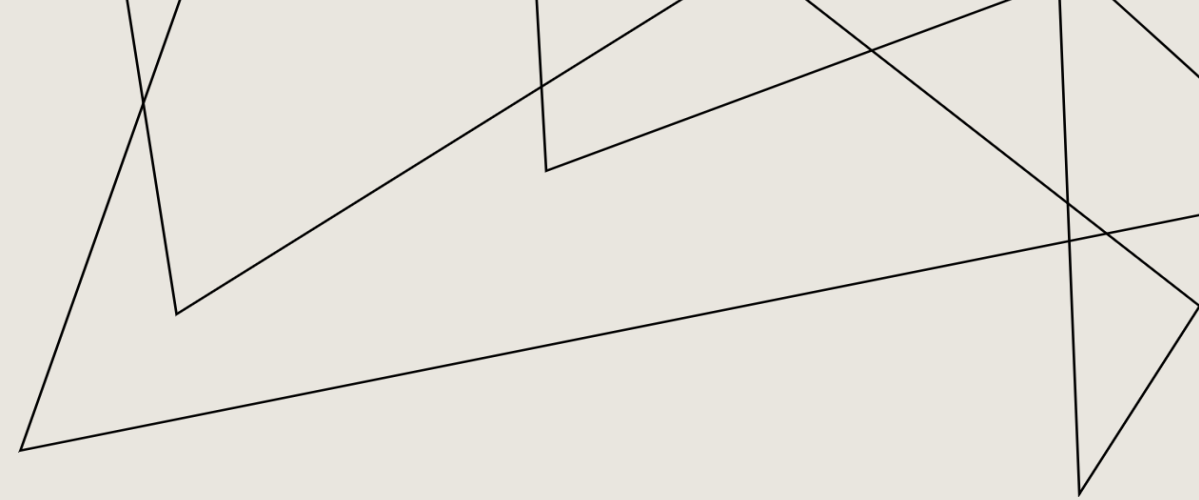
HOUGH TRANSFORM

The Hough Transform is a classic computer vision technique used to detect geometric shapes, most commonly straight lines, in an image. It works by transforming edge points in the image space into a parameter space defined by angle and distance (typically θ and ρ). In this space, points that lie along the same straight line in the image will intersect at a common parameter pair, effectively "voting" for the presence of that line. In our project, after processing the depth image to highlight strong edges, we apply the Hough Transform to identify straight-line segments that correspond to real-world structural features—such as the edge of a table or a sudden drop-off.



WORLD COORDINATE PROJECTION

Once the edge lines are detected in the image, the next step is to project them into the robot's world coordinate system. Using a built-in projection method provided by the VEX-AIM framework, we convert the pixel coordinates of each line endpoint into real-world positions on the ground plane. These projected lines are then added to the robot's world map as solid walls without doorways. This means the path planner will treat them as impassable obstacles, ensuring that the robot does not attempt to drive across them. As a result, the robot becomes aware of environmental boundaries like table edges and can safely navigate without risking a fall.



RESULT

In our final implementation, the robot successfully identified edge lines from the depth image and projected them into its world map as obstacles. These edges were integrated as impassable walls, and the robot's path planner correctly avoided them during navigation, preventing it from driving off the table. However, the system is not without imperfections. Some of the detected lines had slight inaccuracies in angle or position, leading to minor misalignments in the world map. Additionally, because the system relies solely on sharp depth changes, it occasionally misidentified distant vertical features—such as the edge of a whiteboard behind the table—as cliffs. Despite these limitations, the overall system greatly improved the robot's awareness of its surroundings and its ability to avoid dangerous drop-offs.

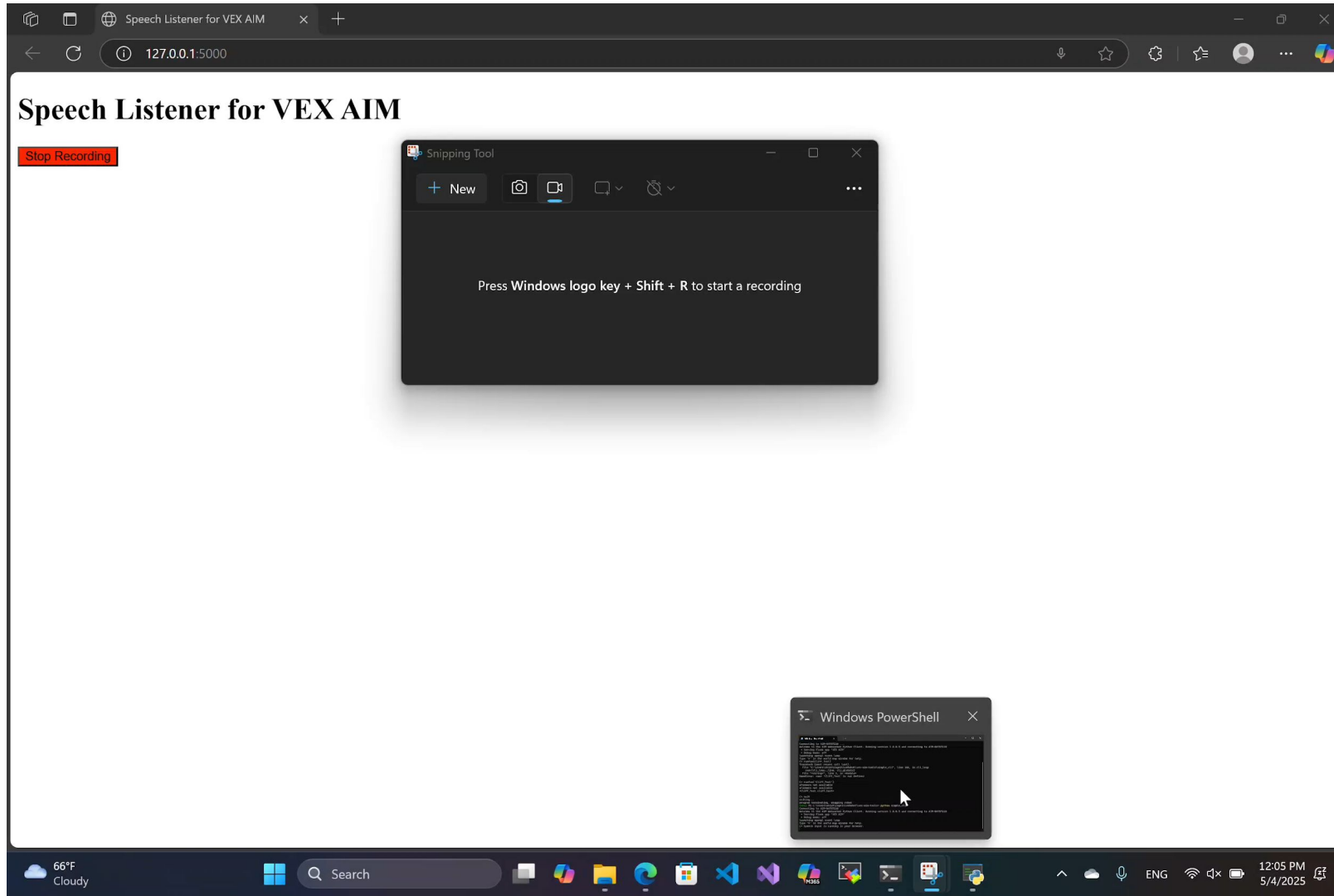


FUTURE WORK

In the future, we aim to improve the precision of edge detection by filtering out unrelated or distant lines that do not represent actual cliffs. This could involve incorporating semantic context or additional geometric constraints to distinguish valid edges from background features. Another important direction is supporting curved edges, which would allow the system to adapt to environments with non-linear boundaries, such as round tables. By extending beyond straight-line detection, the robot could build a more accurate and flexible understanding of its surroundings.



Demo Video



A series of white, thin, overlapping geometric lines on a black background, forming a complex, abstract shape on the left side of the slide.

THANK YOU