SAVE LUNA FROM COLLIDING

Ashwary Anand, Other Author/ Contributor Names in Order of their contribution to the project *

***Abstract*— The main work done in this task was to be able to tell whether the objects seen by Luna were how far from it and using the photographs captured by its cameras in the left and right eye.**

I. METHODOLOGY USED IN THE CODE

1) Loading Images: Two grayscale images, `left_image` and `right_image`, are loaded from files named 'left.png' and 'right.png', respectively, using OpenCV's `cv2.imread()` function.
2) Image Matching Function (`image_match()`):
   - This function takes the left and right images, along with parameters for grid size and maximum disparity, as inputs.
   - It initializes an empty matrix `matrix` to store the disparity map, where each pixel represents the disparity value.
   - The function iterates over each pixel in the left image, excluding a border of size `size1` to avoid out-of-bounds errors.
   - For each pixel, it extracts a grid of size `grid` centered around it from the left image.
   - Then, it iterates over a range of disparities (from 0 to `max_disparity`).
   - For each disparity value, it extracts a corresponding grid from the right image.
   - It calculates the absolute difference between the left and right grids and selects the disparity value that minimizes this difference.
   - Finally, it stores the best disparity value scaled to the range [0, 255] in the `matrix`.
3) Stereo Matching:
   - The `image_match()` function is called with the left and right images, along with the specified grid size and maximum disparity range.
   - This computes the disparity map for the stereo pair.
4) Applying Color Map:
   - OpenCV's `cv2.applyColorMap()` function is used to apply a color map to the disparity map (`matrix`).
   - In this case, the "Jet" color map (`cv2.COLORMAP_JET`) is applied, which maps disparity values to colors ranging from blue to green, and finally to red.
5) Displaying the Depth Map:
   - The resulting depth map (colorized disparity map) is displayed using `cv2.imshow()` function.
   - The program waits for a key press (`cv2.waitKey(0)`) before closing the window.
   - After the window is closed, the resources are released using `cv2.destroyAllWindows()`.

Overall, this code performs stereo matching between two images to estimate the disparity between corresponding points, and then visualizes the disparity map as a colorized depth map for visualization purposes.

*Write anyone who might have helped you accomplish this eg any senior or someone