Disparity Map using OpenCV

Soham Bhattacharya*
Ramakrishna Mission Vivekananda Educational and Research Institute

Abstract

Stereo vision enables the perception of depth by simulating binocular vision using two cameras. This project focuses on generating a disparity map from a stereo image pair. By identifying the pixel displacement between corresponding points in the left and right images, a grayscale disparity map is computed, where brighter values indicate closer objects. The resulting disparity map provides a visual representation of depth and is a fundamental component in 3D reconstruction and autonomous navigation tasks.

Keywords: rendering, feature detection, matching keypoints, homography estimation, warping functions

1 Introduction

Disparity mapping is a fundamental process in computer vision for estimating depth from stereo image pairs. Inspired by the human visual system, stereo vision captures the same scene from two slightly different viewpoints. The disparity, or the horizontal shift of corresponding points between the two views, inversely correlates with the depth of the observed objects. Generating a disparity map allows for the creation of depth perception, which is vital in applications such as 3D scene reconstruction, robotics, and autonomous vehicles.

This report presents the implementation of a basic disparity map generation pipeline using OpenCV's stereo block matching algorithms. It emphasizes image rectification, grayscale conversion, and the use of the Semi-Global Block Matching (SGBM) algorithm to compute the disparity between two stereo images. The reference image is Figure **??**

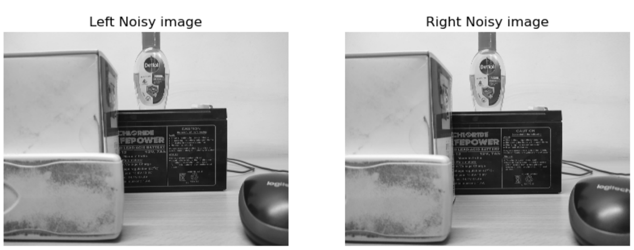

Figure 1: *Original images*

The method for generating a disparity map involves the following major steps:

Reading and Preprocessing Images

The stereo image pair is first read and converted to grayscale using:

$$I_{gray} = 0.299R + 0.587G + 0.114B$$

This simplifies processing and focuses on intensity differences.

Stereo Matching using SGBM

OpenCV's `StereoSGBM_create` is used to compute disparity. Important parameters include `minDisparity`, `numDisparities`, `blockSize`, `P1`, and `P2`. The disparity for each pixel is calculated by minimizing matching costs along scanlines:

$$d(x, y) = x_L - x_R$$

Normalization and Visualization

The disparity map is normalized to enhance visibility:

$$D_{norm} = \frac{D - D_{min}}{D_{max} - D_{min}} \times 255$$

The final image presents depth information with brighter regions indicating closer objects. This is optional, and hence I have not chosen to do so because, the results with the original image almost remained the same.

2 Method Overview

1. Image Acquisition: Two rectified grayscale images of the same scene are captured from slightly different viewpoints. Stereo vision relies on horizontal alignment of these images.

```
imgL = cv.imread('left_image.jpg',
    cv.IMREAD_GRAYSCALE)
imgR = cv.imread('right_image.jpg',
    cv.IMREAD_GRAYSCALE)
```

2. StereoBM Parameter Configuration: A StereoBM matcher is created with specific parameters. The cost function used for matching is optimized semi-globally.

```
cv2.StereoBM_create(numDisparities=128, blockSize=9)
```

3. Disparity Computation: Disparity is calculated using:

$$\text{Disparity}(x, y) = x_{\text{left}} - x_{\text{right}}$$

Which is related to depth via:

$$\text{Depth} = \frac{f \cdot B}{\text{Disparity}}$$

```
disparity = stereo.compute(imgL, imgR)
```

4. Normalization and Visualization: The raw disparity map is scaled and optionally colored.

```
disparity_normalized = cv.normalize(disparity,
    None, 0, 255, cv.NORM_MINMAX)
color_map = cv.applyColorMap(
    np.uint8(disparity_normalized),
    cv.COLORMAP_JET)
```

3 Results

The generated disparity map, as shown in Figure **??**, successfully visualizes the depth variation between two stereo images. Distinct objects at different depths are clearly discernible through variations in gray levels, where closer objects appear brighter and distant objects darker. After tuning parameters such as `numDisparities` and `blockSize`, the output disparity map produces a more accurate and noise-free representation of the scene's depth, resembling the structure observed in the sample reference. The final map

*e-mail:bhattacharyasoham026@gmail.com

provides sufficient contrast and demarcation of object boundaries, useful for further 3D reconstruction or depth-based segmentation tasks.

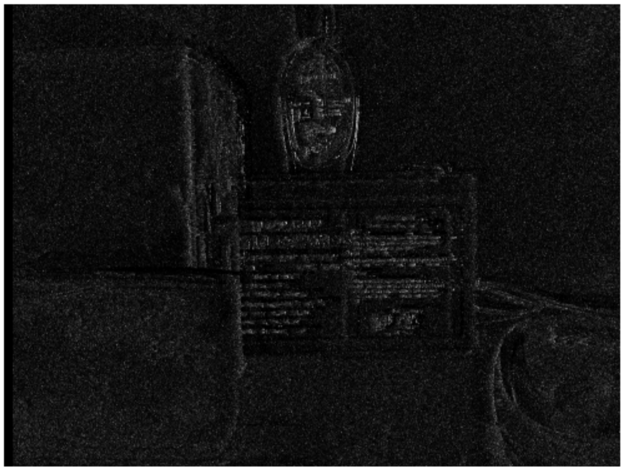

Figure 2: *Disparity Map of based on ORIGINAL image*