Experiment no 4
To study the Depth Estimation
Date of Performance :07/08/2023
Date of Submission :14/08/2023

Aim: To study the Depth Estimation

Objective: To Capturing Frames form a depth camera creating a mask from a disparity map Masking a copy operation Depth estimation with normal camera

Theory:

1. Depth map

A depth map is an image representation where each pixel contains information about the distance from the camera to the corresponding point in the scene. It provides a 3D-like understanding of the captured scene.

2. Point cloud map

A point cloud is a collection of 3D points that represent the surfaces of objects in a scene. It is often generated from the depth map and can be used for various applications like 3D reconstruction and visualization.

3. disparity map

A disparity map is derived from a stereo pair of images captured by two cameras placed a known distance apart. It represents the pixel-wise horizontal shift between the images and can be used to calculate depth information.

Creating a Mask from a disparity map

The disparity map can be thresholded to create a valid depth mask. Pixels with a certain disparity value range are considered valid and used for depth estimation.

Masking a Copy Operation

By applying the valid depth mask to an image, you can copy or paste objects from one scene to another while maintaining depth consistency. This can be useful for augmented reality applications.

Depth estimation with a normal camera

While a depth camera directly provides depth information, estimating depth from a single normal camera requires additional techniques. This can involve using stereo vision, structure-from-motion, or deep learning-based methods.

Code:

```
import cv2
from google.colab.patches import cv2_imshow
import numpy as np

# Load the disparity map and RGB image
disparity_map = cv2.imread("/content/macaw.png", cv2.IMREAD_GRAYSCALE)
rgb_image = cv2.imread("/content/macaw.png")

if disparity_map is None or rgb_image is None:
    print("Image loading failed.")
else:
    # Create a valid depth mask from the disparity map
    valid_mask = disparity_map > 0

    # Apply depth mask to the RGB image
    depth_estimated_image = np.copy(rgb_image)
    depth_estimated_image[valid_mask & (disparity_map < 100)] = [128, 0, 
128]  # Purple color (BGR format)

    # Display the original disparity map, RGB image, and depth-estimated 
image
    down_width = 600
    down_height = 500
    down_points = (down_width, down_height)

    resized_down = cv2.resize(disparity_map, down_points, 
interpolation=cv2.INTER_LINEAR)
    resized_down2 = cv2.resize(rgb_image, down_points, 
interpolation=cv2.INTER_LINEAR)
    resized_down3 = cv2.resize(depth_estimated_image, down_points, 
interpolation=cv2.INTER_LINEAR)

    print("Disparity Map")
    cv2_imshow(resized_down)

    print("RGB Image")
    cv2_imshow(resized_down2)

    print("Depth-Estimated Image")
    cv2_imshow(resized_down3)
```

Output:

Input Image:

Disparity Map:

Depth-Estimated Image:

Conclusion:

In this study, we explored the concept of depth estimation using disparity maps and depth masks in the context of computer vision. We aimed to capture a comprehensive understanding of depth information within a scene and how it can be harnessed to enhance image manipulation and perception.