Aim: To study the Depth Estimation.

Objective: To Capturing Frames from 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 a grayscale image that represents the distance of objects from the camera. Each pixel's value indicates the depth or distance of the corresponding point in the real world. Darker areas typically correspond to objects that are farther away, while lighter areas correspond to objects closer to the camera.

2. Point Cloud Map:

A point cloud is a collection of 3D points in a coordinate system, usually obtained from depth information captured by cameras or sensors like LiDAR. Each point in the point cloud corresponds to a specific location in 3D space and may include attributes like color or intensity.

3. Disparity Map:

A disparity map is a 2D representation that encodes the difference in horizontal pixel coordinates between the left and right images of a stereo camera setup. It's used to estimate the depth of objects in the scene. Greater disparities indicate closer objects.

4. Valid Depth Mask:

A valid depth mask is an image or binary mask that identifies which pixels in a depth map have reliable depth values. This mask helps filter out areas with unreliable or missing depth information, which might occur due to occlusion or limitations of the depth sensing technology.

5. Creating a Mask from a Disparity Map:

This involves using a disparity map to create a mask that highlights certain depth ranges. Pixels within a specified disparity range are included in the mask, while others are excluded. This canhelp isolate objects of interest based on their distance from the camera.

6. Masking a Copy Operation:

Masking and copy operation involves using a mask to selectively copy certain parts of an image to another image. The mask determines which pixels are copied and which are ignored. This technique is often used to overlay or modify specific regions in an image.

7. Depth Estimation with a Normal Camera:

Depth estimation using a normal camera typically involves using image processing techniques to infer depth information from a single 2D image. Methods might include analyzing object sizes, perspective, shadows, and textures to make educated guesses about relative distances.

Code:

```
import numpy as np
import cv2
from matplotlib import pyplot as plt
# Read two input images
imgL = cv2.imread('left.jpg')
imgR = cv2.imread('right.jpg')
# Convert the images to grayscale
grayL = cv2.cvtColor(imgL, cv2.COLOR_BGR2GRAY)
grayR = cv2.cvtColor(imgR, cv2.COLOR_BGR2GRAY)
# Initiate a StereoBM object
stereo = cv2.StereoBM_create(numDisparities=16, blockSize=15)
# Compute the disparity map
disparity = stereo.compute(grayL, grayR)
# Normalize the disparity map for visualization
normalized_disparity = cv2.normalize(disparity, None, alpha=0, beta=255,
norm_type=cv2.NORM_MINMAX, dtype=cv2.CV_8U)
# Display the disparity map using matplotlib
plt.imshow(normalized_disparity, cmap='gray')
plt.title('Disparity Map')
```

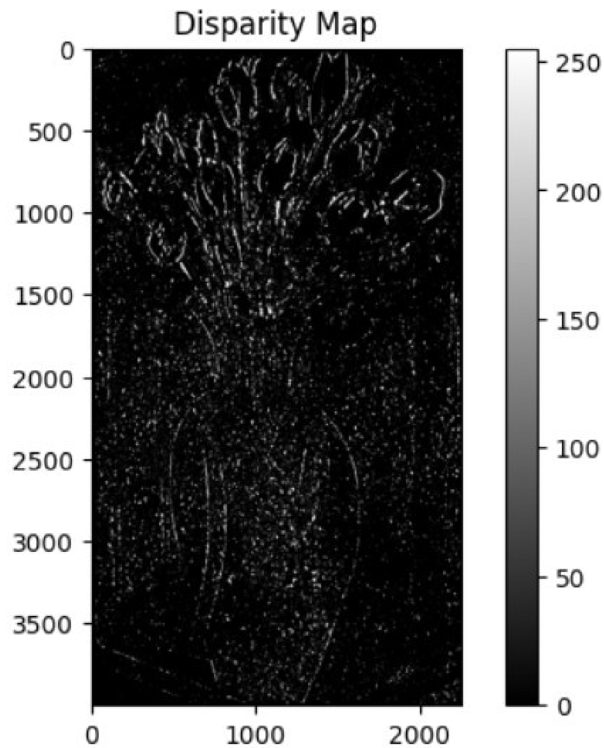
```
plt.colorbar()

plt.show()

# Print the shape of the disparity map

print("Disparity Map Shape:", disparity.shape)
```

Input:

Right View

Left View

Output:




```
Disparity Map Shape: (4000, 2250)
```

Conclusion:

The process of depth estimation using stereo images and the StereoBM algorithm was explored. The StereoBM algorithm compared pixel intensities in the left and right images to calculate disparities, which represent the horizontal pixel shifts between the images. The normalized disparity map was displayed using a grayscale colormap. The colormap provided insight into the depth relationships within the scene, highlighting objects with varying disparities. The main steps of the experiment included image loading, conversion to grayscale, creation of a StereoBM object, computing the disparity map, normalizing and visualizing the disparity map, and examining its shape.