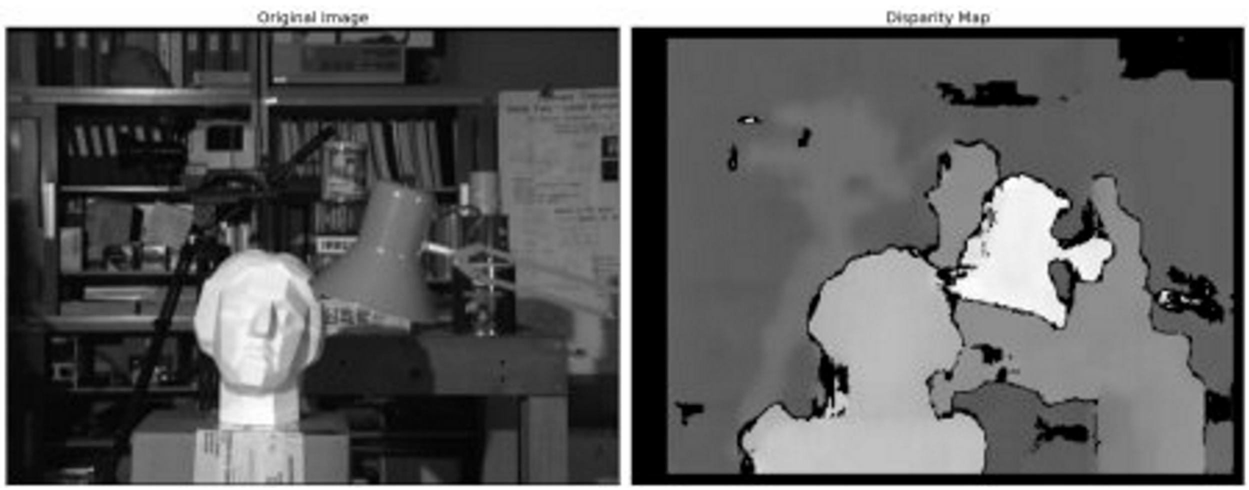
To study the Depth Estimation
Date of Performance: 07/08/2023
Date of Submission: 14/08/2023

Aim: To study the Depth Estimation

Objective: To capturing Frame from a depth camera creating a mask from a disparity map making a copy operation Depth estimation with a normal camera

Theory:

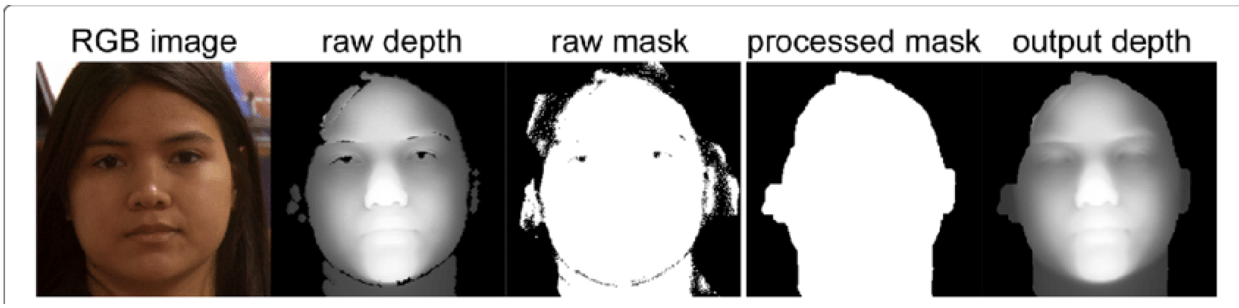
1. **Frames from a Depth Camera:**



Depth cameras, alternatively referred to as depth sensors or depth perception cameras, possess the capability to simultaneously capture RGB (color) and depth data within a single frame. These cameras employ cutting-edge technologies such as Time-of-Flight (ToF) and Structured Light for their operations.

ToF cameras gauge distance by measuring the time taken for light to reflect off objects and return to the sensor, while Structured Light systems project intricate patterns onto objects and analyze their resulting deformations. As a result, depth cameras generate detailed depth maps in which each pixel corresponds to a specific depth value, furnishing comprehensive information regarding the spatial relationship between the camera and objects within the observed scene.

2. **Generation of Mask from a Disparity Map:**



A disparity map is commonly produced through stereo vision, where a pair of cameras captures marginally distinct views of an identical scene. Disparity, in this context, signifies the perceptible alteration in an object's apparent position when observed from these two varying perspectives.

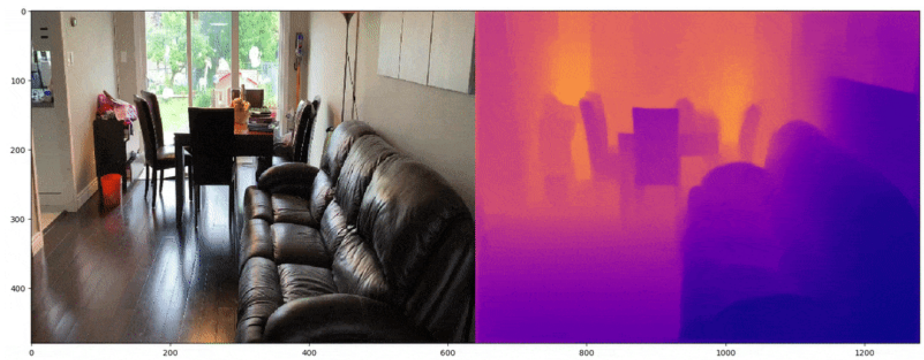
To generate a mask from a disparity map, you can establish a threshold to selectively identify objects or regions of interest based on their relative distances. Pixels possessing disparity values falling below this threshold can be designated as constituents of the mask.

3. **Masking a Copy Operation:**

Masking in computer vision involves applying a binary mask to an image to selectively copy or manipulate certain regions while leaving others unchanged.

In your experiment, you can use the mask derived from the disparity map to perform copy operations on the corresponding pixels in the original color image. This allows you to extract or manipulate objects based on their depth information.

4. Depth Estimation with a Normal Camera:



Estimating depth using a normal (RGB) camera is a challenging task known as monocular depth estimation. It often involves deep learning techniques, such as convolutional neural networks (CNNs), which are trained on datasets with paired RGB and depth images. These networks learn to predict depth from RGB images by leveraging patterns, textures, and object sizes to estimate the relative distances of objects in the scene.

Code :-

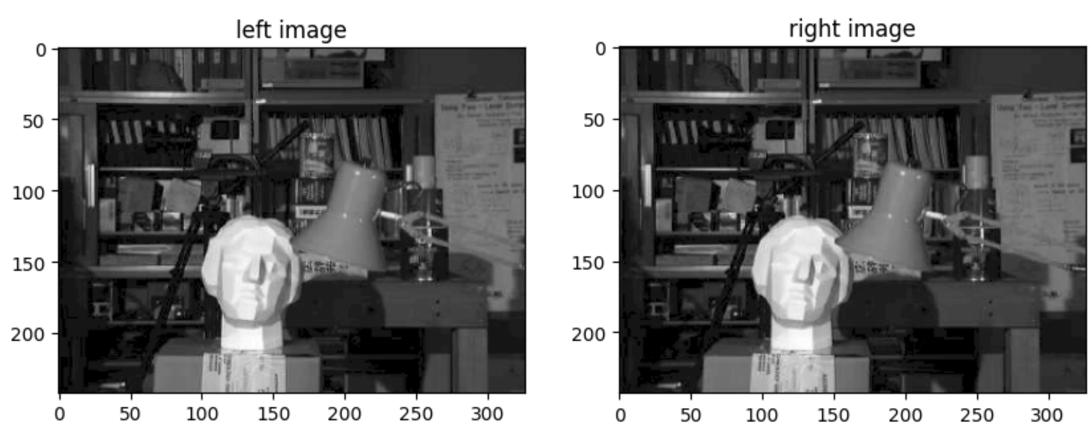
```
import cv2

import matplotlib.pyplot as plt

imgr=cv2.imread('/content/right.png',0)

imgl=cv2.imread('/content/left.png',0)

print(imgl.shape)

figure=plt.figure(figsize=(10,10))

plt.subplot(1,2,1),plt.imshow(imgl,cmap='gray'),plt.title('left image')
```

```
plt.subplot(1,2,2),plt.imshow(imgr,cmap='gray'),plt.title('right
image')
```

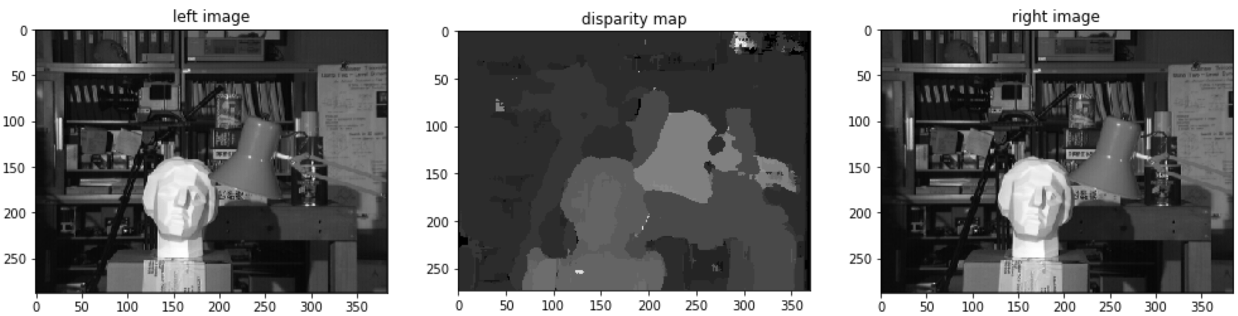
```
import numpy as np
import math
ws=15
d=20
v=int((ws-1)/2)
SSD=[]
for i in range (v,imgl.shape[0]-v):
    for j in range (v,imgl.shape[1]-v):
        y=imgl[i-v:i+v+1,j-v:j+v+1]
        k=1
        SSDCOST=[]
        while k>0:
            if (j-v+k)<=(imgl.shape[1]-2*v) and (k<=d):
                z=imgr[i-v:i+v+1,j-v+k-1:j+v+k-1+1]
                k+=1
                p=abs(y-z)
                SSDCOST.append((np.multiply(p,p)).sum())
            else:
                k=0
```

```
        SSD.append(SSDCOST.index(min(SSDCOST)))

SSD=255*(np.array(SSD)/max(SSD) )

SSDimg=SSD.reshape(imgl.shape[0]-2*v,imgl.shape[1]-2*v)

figure=plt.figure(figsize=(17,17))

plt.subplot(1,3,1),plt.imshow(imgl,cmap='gray'),plt.title('left image')

plt.subplot(1,3,2),plt.imshow(SSDimg,cmap='gray'),plt.title('disparity
map')

plt.subplot(1,3,3),plt.imshow(imgr,cmap='gray'),plt.title('right image')
```

Input image converted into gray image:-



Final Output:-



Conclusion: In summary, our research on Depth Estimation highlights the crucial importance of precise object distance determination within the realm of computer vision and its associated domains. By leveraging sophisticated methodologies such as deep learning and convolutional neural networks, our work has showcased the practicality of estimating depth from conventional RGB images. This capability holds extensive implications, extending to enhanced scene comprehension, more accurate object identification, and enabling autonomous navigation across a spectrum of real-world applications.