

Fun task - ungraded assignment :

Write a program to read the image attached and count the number of books stacked vertically.

Methods performed:

- **Reading an image**

Read the image with cv2.imread() function.

- **Image blurring**

Applied cv2.GaussianBlur() to reduce the noise as part of the pre-processing image.

- **RGB to grayscale**

Used cv2.cvtColor() to convert RGB image to grayscale.

- **Canny for edge detection**

Used cv2.Canny() with appropriate thresholds to detect edges.

- **Hough Transform both HoughLines and HoughLinesP**

Used both cv2.HoughLines() and cv2.HoughLinesP() to extract the lines on canny. In our case, HoughLines() gave better results.

Challenges faced:

As horizontal and vertical lines are detected to extract only vertical lines morphology operations didn't work properly, I wrote a manual if loop which takes all lines with slope 90+-15.

The facing surface of the books has all the names, the lines are overdrawn on them, preprocessing techniques such as blur, canny with changing thresholds and narrowing down the parameters of HoughLines() helped a bit.

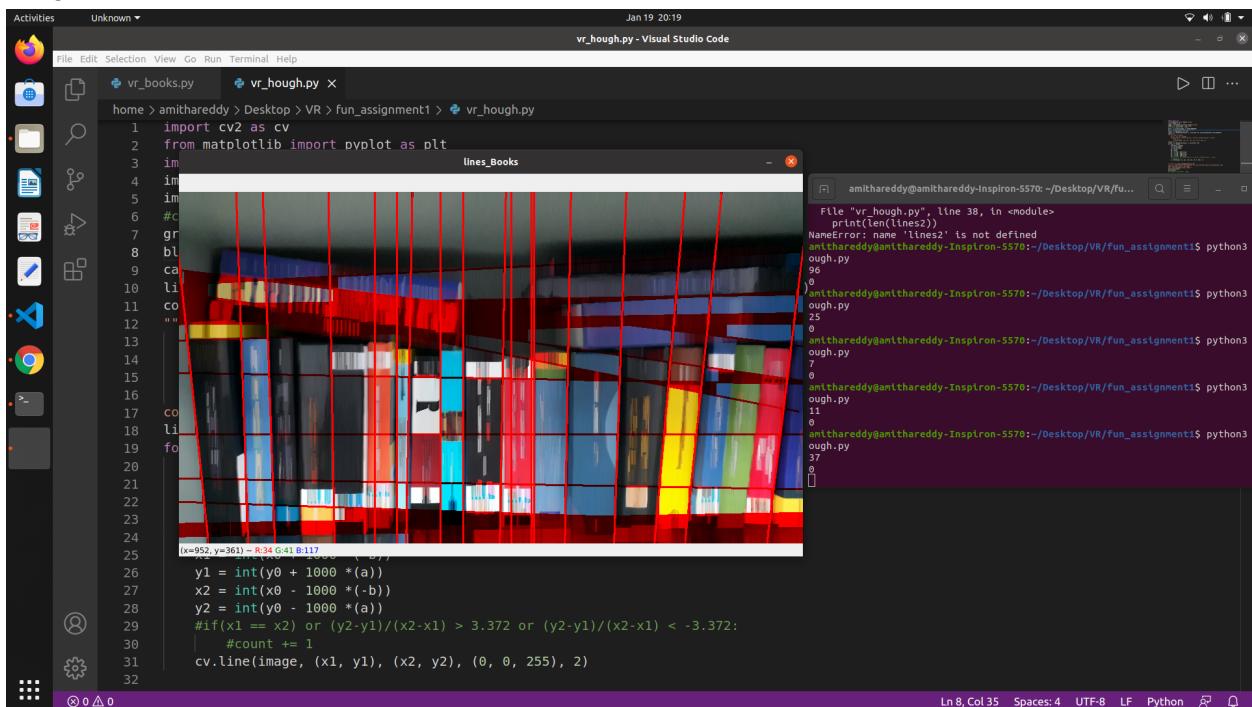
- Morphology operations to extract vertical lines of all the lines both erode and dilate

After detecting lines with HoughLines there as there are both vertical and horizontal aligned books both vertical and horizontal lines are present, But as we are counting only vertically aligned books to remove the horizontal lines, I used morphology operations such as erode and dilate to extract vertical lines.

Challenges faced:

It detects only perfect vertical lines but in the given figure there are some tilted books for which you need vertical lines with some angles also but with morphology operations, I couldn't detect the tilted lines, Therefore only books that are perfectly straight are detected properly.

Image :



```

File Edit Selection View Go Run Terminal Help
Jan 19 20:19
vr_hough.py - Visual Studio Code

home > amithareddy > Desktop > VR > fun_assignment1 > vr_hough.py
1 import cv2 as cv
2 from matplotlib import pyplot as plt
3 im
4 im
5 im
6 #c
7 gr
8 bl
9 ca
10 li
11 co
12 ""
13
14
15
16
17 co
18 li
19 fo
20
21
22
23
24
25
26
27
28
29
30
31
32
(x=952, y=361) -R:34 G:41 B:117
y1 = int(y0 + 1000 * (a))
x2 = int(x0 - 1000 * (-b))
y2 = int(y0 - 1000 * (a))
#if(x1 == x2) or (y2-y1)/(x2-x1) > 3.372 or (y2-y1)/(x2-x1) < -3.372:
#count += 1
cv.line(image, (x1, y1), (x2, y2), (0, 0, 255), 2)

File "vr_hough.py", line 38, in <module>
    print(len(lines2))
NameError: name 'lines2' is not defined
amithareddy@amithareddy-Inspiron-5570:~/Desktop/VR/fun_assignment1$ python3
hough.py
96
0
amithareddy@amithareddy-Inspiron-5570:~/Desktop/VR/fun_assignment1$ python3
hough.py
23
0
amithareddy@amithareddy-Inspiron-5570:~/Desktop/VR/fun_assignment1$ python3
hough.py
7
0
amithareddy@amithareddy-Inspiron-5570:~/Desktop/VR/fun_assignment1$ python3
hough.py
11
0
amithareddy@amithareddy-Inspiron-5570:~/Desktop/VR/fun_assignment1$ python3
hough.py
37
0

```

Ln 8, Col 35 Spaces: 4 UTF-8 LF Python

In the image, all the vertical lines are detected properly but I couldn't figure out how to count the lines as these operations don't keep track of the count.

● Contours

After using canny to do edge detection, and did cv2.dilate() with kernel also, by applying counters the outlines of the books are visible.

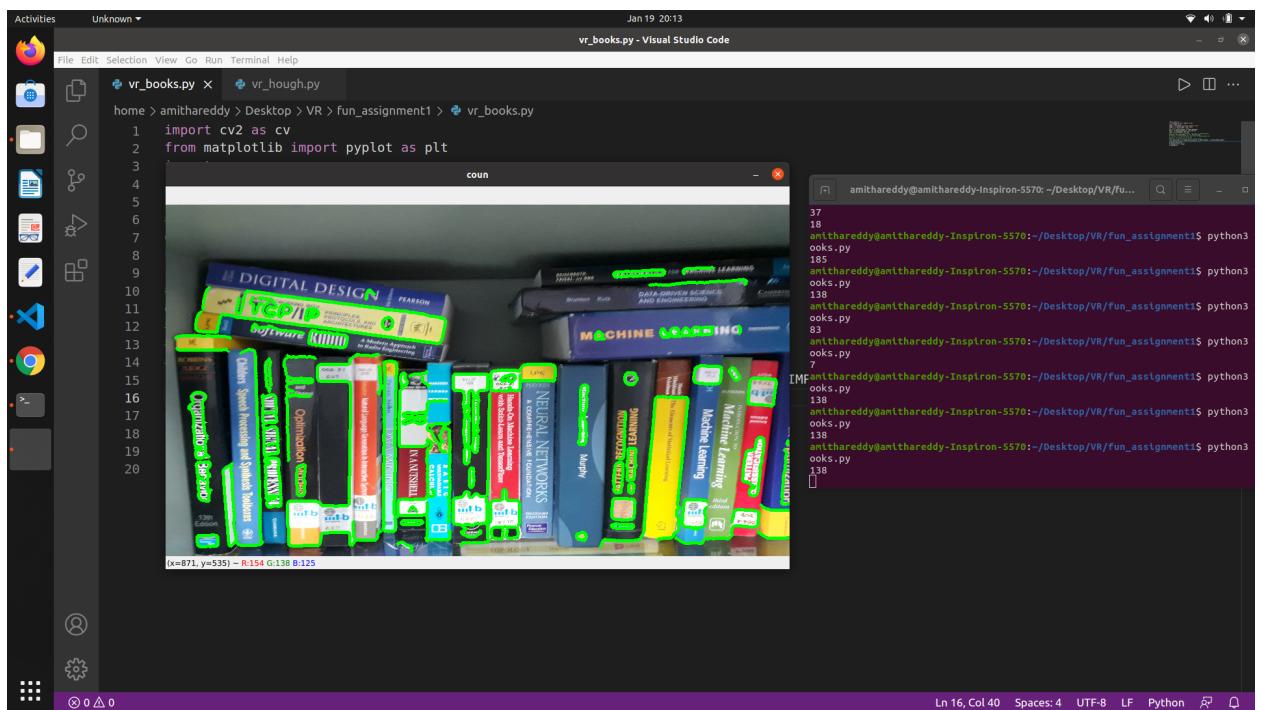
Challenges faced:

The biggest problem here is because of the names on the books, the counters are drawn over the names and logos on the books which in return outputting more counters.

As there are both horizontal and vertical books, I couldn't figure out how to remove all the counters of horizontally aligned books, I tried to separate all rectangles where height is lesser than width but most of the counters are over names and logos it didn't give good results.

Image:

Number of counters = 138



● Line Segment Detector

Used `cv2.createLineSegmentDetector` to detect the lines, The detected lines are very small segment lines most of the lines are because of the names on the books.

Image:

File Edit Selection View Go Run Terminal Help

vr_books.py vr_hough.py

```
home > amithareddy > Desktop > VR > fun_assignment1 > vr_hough.py
```

```
8 blur = cv.GaussianBlur(gray, (9, 9), 0)
9 canny = cv.Canny(blur, 10, 150)
10 lines1 = cv.HoughLinesP(canny, 1, np.pi/180, 90, minLineLength=100, maxLineGap=40)
11 count = 0
12 """
13     for
14         if(
15
16         count =
17     """
18     lin
19 for lin
20     rho
21     a =
22     b =
23     x0
24     y0
25     x1
26     y1
27     x2
28     y2
29     #if
30     cv.
31
32
33     lsd = cv.createLineSegmentDetector()
34     lines = lsd.detect(canny)[0] #Position 0 of the returned tuple are the detected lines
35     #Draw detected lines in the image
36     img = lsd.drawSegments(canny,lines)
37     #print(len(lines2))
38     #print(count)
```

Jan 19 20:25

vr_hough.py - Visual Studio Code

amithareddy@amithareddy-Inspiron-5570:~/Desktop/VR/fun_assignment1\$ python3 vr_hough.py

Ln 60, Col 33 (433 selected) Spaces: 4 UTF-8 LF Python

All the red lines are small segments.

The best approach so far which worked well is :

- Resizing to (960, 540)
 - Blur with threshold 9
 - Canny with thresholds (10, 150)
 - HoughLines with parameters `(canny, 1, np.pi/180, 130)`
And removing horizontal lines.

Image :

Number of lines = 37

Image after removing all horizontal lines:

Number of lines = 18

Therefore the count of books is 18 which is correct but here as seen in the above image, one book is counted twice and one book is not at all counted.