231501037

EXP NO: 07
DATE: 12-09-2025

Various image segmentation algorithm

Aim: Implement various image segmentation algorithm

Algorithm:

1. Read and preprocess image (resize, blur).
2. Apply thresholding (Otsu/manual) or clustering (K-means).
3. Label segments using connected component analysis.
4. Optionally use edge detection before segmentation.
5. Visualize segmented regions with colors.
6. Display final segmented image.

Code:

```
import cv2, matplotlib.pyplot as plt
img = cv2.imread('image.jpg', 0)
edges = cv2.Canny(cv2.GaussianBlur(img, (5,5), 0), 100, 200)
plt.imshow(edges, cmap='gray'); plt.title("Canny Edges"); plt.show()
import cv2, numpy as np, matplotlib.pyplot as plt
img = cv2.imread('image.jpg', 0); h, w = img.shape
seed, th, mask = (h//2, w//2), 5, np.zeros_like(img)
stack = [seed]
while stack:
    x, y = stack.pop()
    if 0<=x<h and 0<=y<w and mask[x,y]==0 and abs(int(img[x,y])-int(img[seed]))<th:
        mask[x,y]=255; stack += [(x+dx,y+dy) for dx in [-1,0,1] for dy in [-1,0,1]]
plt.imshow(mask, cmap='gray'); plt.title("Region Growing"); plt.show()
```

231501037

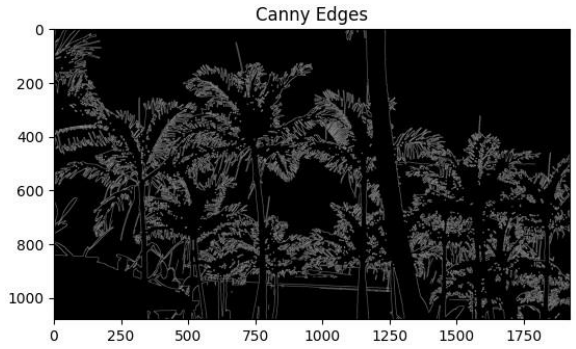
```
import cv2, numpy as np, matplotlib.pyplot as plt
img = cv2.imread('image.jpg'); gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
_, th = cv2.threshold(gray,0,255,cv2.THRESH_BINARY_INV+cv2.THRESH_OTSU)
dist = cv2.distanceTransform(th,1,5); _, fg = cv2.threshold(dist,0.7*dist.max(),255,0)
bg = cv2.dilate(th,(3,3),3); unk = cv2.subtract(bg.astype(np.uint8),fg.astype(np.uint8)); _,
markers = cv2.connectedComponents(np.uint8(fg))
markers = cv2.watershed(img, markers+1); img[markers==-1]=(255,0,0)
plt.imshow(cv2.cvtColor(img, cv2.COLOR_BGR2RGB)); plt.title("Watershed"); plt.show()
```

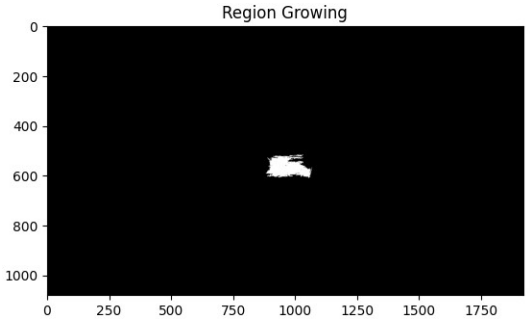
```
from skimage import io, segmentation, color
img = io.imread('image.jpg')
seg = segmentation.felzenszwalb(img, scale=100, sigma=0.5, min_size=50)
io.imshow(color.label2rgb(seg, img)); io.show()
```

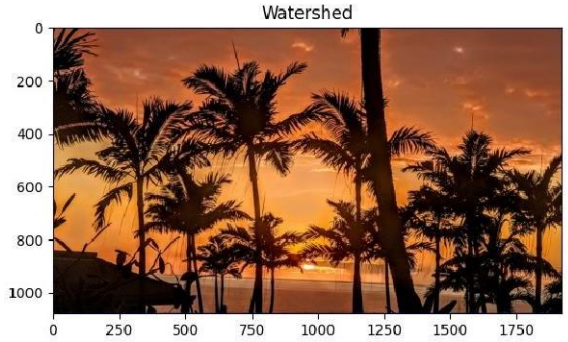
```
import cv2, numpy as np, matplotlib.pyplot as plt
img = cv2.imread('image.jpg'); mask = np.zeros(img.shape[:2], np.uint8)
bgd, fgd = np.zeros((1,65), np.float64), np.zeros((1,65), np.float64)
rect = (50,50,img.shape[1]-100,img.shape[0]-100)
cv2.grabCut(img, mask, rect, bgd, fgd, 5, cv2.GC_INIT_WITH_RECT)
mask2 = np.where((mask==2)|(mask==0),0,1).astype('uint8'); res = img*mask2[:,:,None]
plt.imshow(cv2.cvtColor(res, cv2.COLOR_BGR2RGB)); plt.title("GrabCut"); plt.show()
```

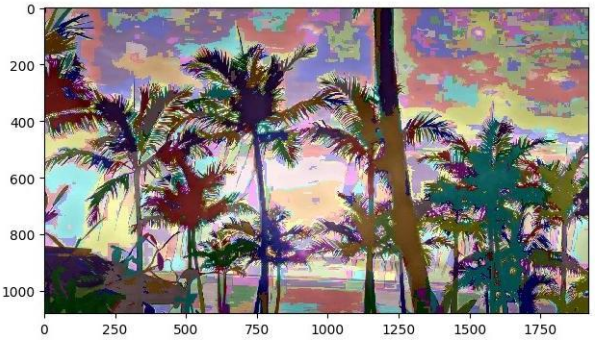
231501037

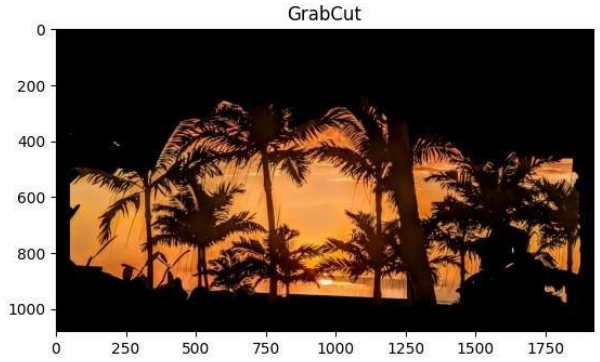
Output:











Result: Thus, various image segmentation algorithm was implemented successfully.