

CSE 344: Computer Vision

Homework 1 ; Arka Sarkar 2018222

Question 2

Algorithm design

To figure out the individual components, we need to keep a visited array to track the number of pixels visited. Once we encounter a pixel that is 1.0 and not visited we need to find out all the neighboring pixels that are 1.0 and connected to it, to achieve this we maintain a stack and stacking all the 1.0 not visited pixels, and in this manner when we continuously pop out the elements we would have covered all the pixels in a single connected component.

Maintain a counter to count the number of individual components and output it as the final result.

Pseudo code for finding out individual components in a binary image is :

$A \leftarrow (\text{rows}, \text{cols})$ Size of Image

Visited \leftarrow array of size(A)

Answer \leftarrow array of size(A)

c= 1

for i \leftarrow 1 : rows :

 for j \leftarrow 1 : cols:

 if(img[i,j] = 0.0):

Mark as visited

 Else if(img[i,j] is visited):

 continue

 else:

 stack = deque()

 stack.append((i,j))

 while(len(stack)!=0):

 curr = stack.pop()

 if(visited[curr[0],curr[1]] == 0):

 visited[curr[0],curr[1]] = 1

 m,n = curr[0],curr[1]

 answer[m,n] = c

```

for x in range(m-1, m+2):
    for y in range(n-1, n+2):
        if(x == m and y == n):
            continue
        else:
            if(x <0 or x > rows -1):
                continue
            if(y <0 or y > cols -1):
                continue

            if(img[x,y] == 1.0):
                stack.append((x,y))

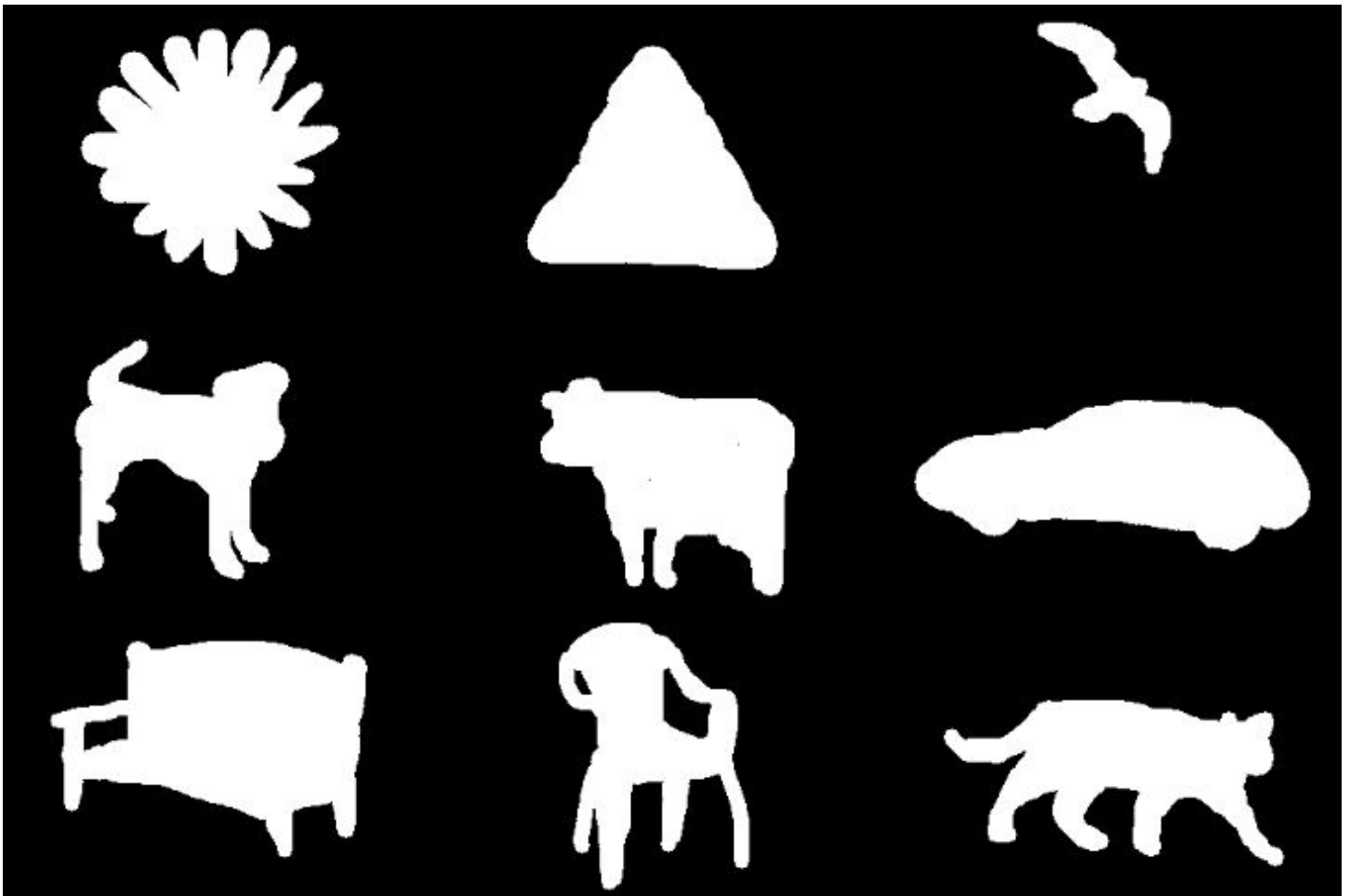
```

$c = c + 1$

Answer $\leftarrow \max(\text{answer})$

Input / Output

Input Image :



Code Output :

```
E:\sem6\ComputerVision\Homeworks\HW1>python 2018222_HW1.py  
(639, 960)  
(639, 960)  
Number of connected components : 9.0  
  
E:\sem6\ComputerVision\Homeworks\HW1>
```