BLG 513E HW2

Bird Counting Algorithm

Eren Kılıç
504201553
Department of Computer Engineering
Istanbul Technical University

1-) INTRODUCTION

In this assignment, we are asked to implement a bird counting algorithm from scratch using segmentation methods. We have given three input images to test our program. The number of birds in each image are 10, 3, and 22, respectively. Our aim was to compare our program with the actual result as can be reviewed in section 3.

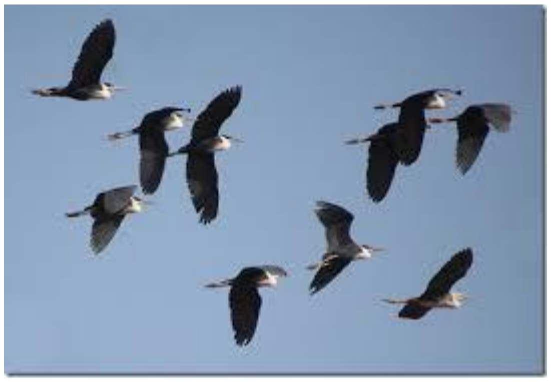


Figure 1: *bird_1.jpg*

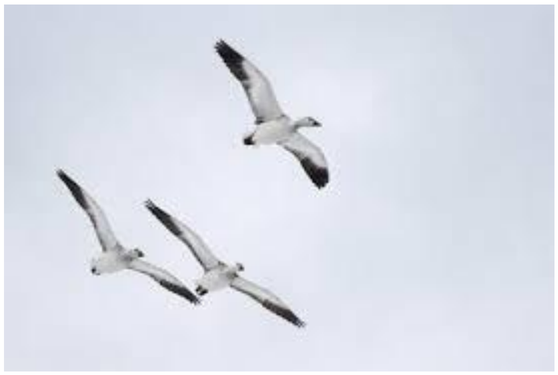


Figure 2: *bird_2.jpg*

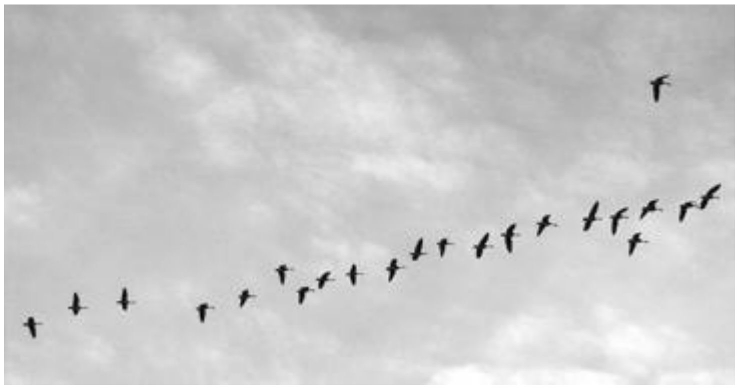


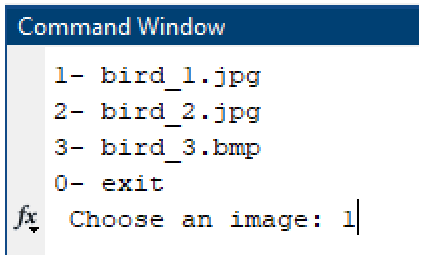
Figure 1: *bird_3.bmp*

2-) STEPS FOLLOWED

Basically, the program has 6 steps, and they are executed to find the number of birds in an input image and compare its actual number.

STEP 1: Menu & Input Read

In the beginning of the code, the program asks the user to select an image as input as shown in Figure 4.



```
1- bird_1.jpg
2- bird_2.jpg
3- bird_3.bmp
0- exit
 Choose an image: 1
```

Figure 4: Menu

When the enter the number, the corresponding image is read using MATLAB Image Processing toolbox's built-in function ``*imread*". Since the images already are already introduced at the beginning of the report, the outputs of step 1 did not put in this section. (They will be the same)

STEP 2: Converting to Grayscale

Input image has 3 channels (R, G, B) and the program converts it to grayscale image using MATLAB Image Processing toolbox's built-in function ``*rgb2gray*".

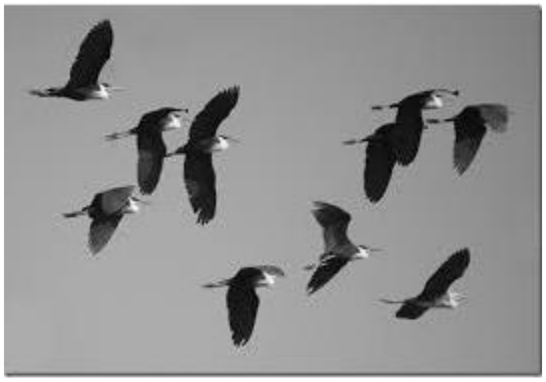


Figure 5: Grayscale (*bird_1.jpg)*

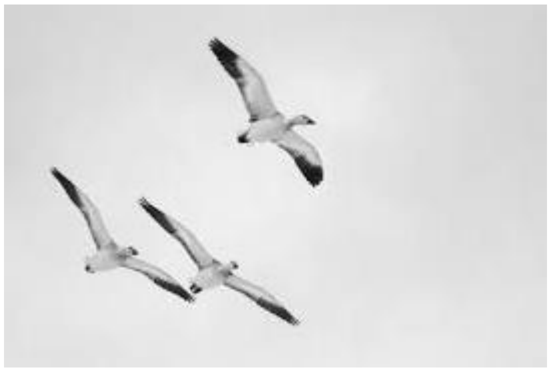


Figure 6: Grayscale (*bird_2.jpg)*

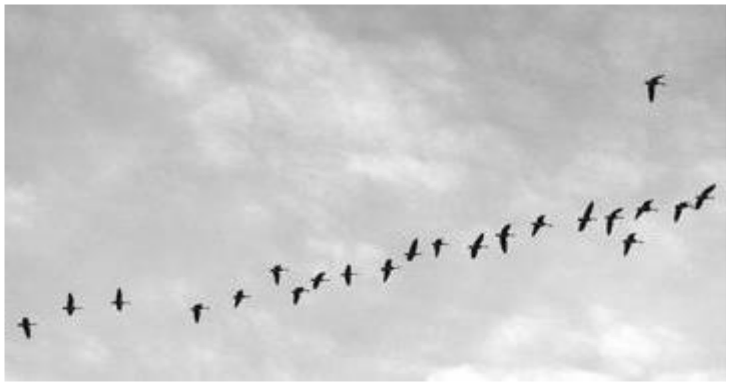


Figure 7: Grayscale (*bird_3.bmp)*

STEP 3: Gaussian Blurring

To remove noises from the image, Gaussian blurring is applied to the grayscale image.

5x5 kernel is produced using the formula below where the sigma value is 1.2.

$$H_{ij} = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(i-(k+1))^2 + (j-(k+1))^2}{2\sigma^2}\right)$$

Then it is convolved to the image.

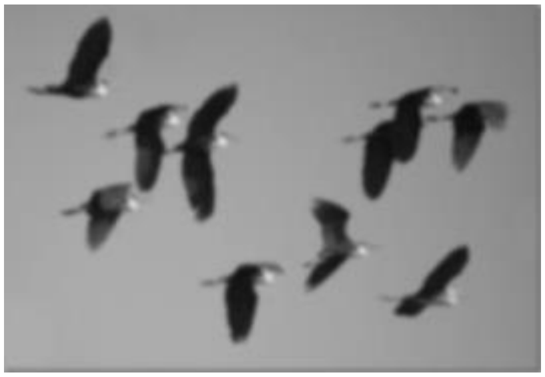


Figure 8: *Gaussian blurred image 1*

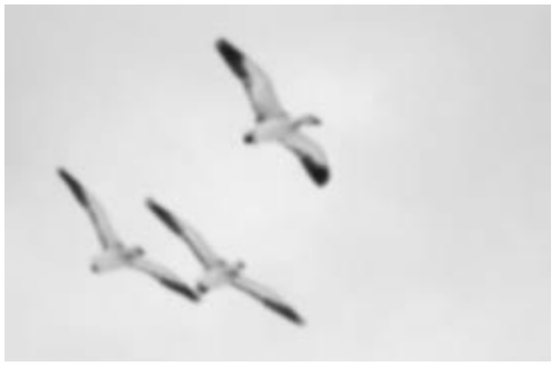


Figure 9: *Gaussian blurred image 2*

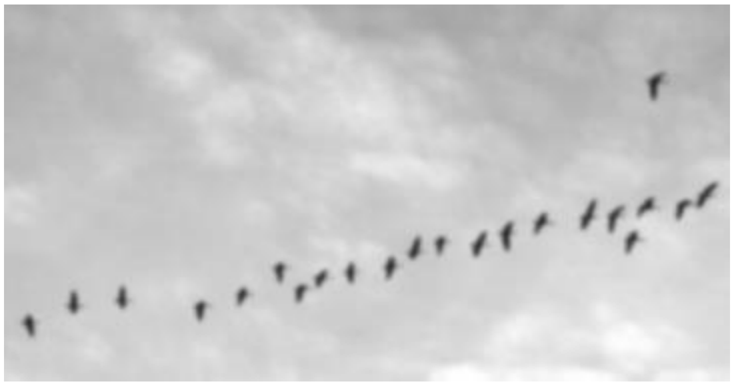


Figure 9: *Gaussian blurred image 3*

STEP 4: Binary Image Conversion

In this step, blurred images are converted to a binary image based on a threshold value. To find the threshold, I developed an algorithm. If the number of dark pixels (i.e. 200) considerably high (10% in this case), the threshold for the binary image should be close to black. If the image has mostly light pixels (objects & background are light), the threshold for binary conversion should be close to the white. The threshold is selected according to this strategy.

Then images are converted to a binary image. Hence, all pixels in the image traversed and converted to black or white pixel based on the comparison with the threshold value.

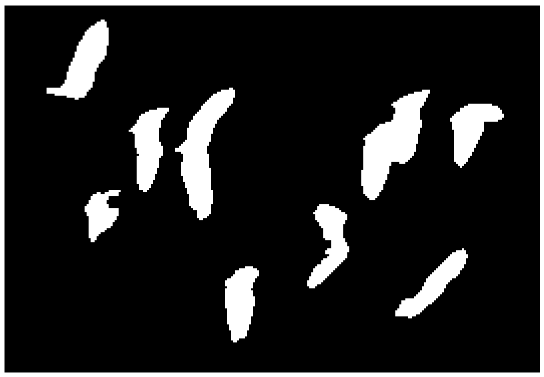


Figure 10: *Binary image 1*

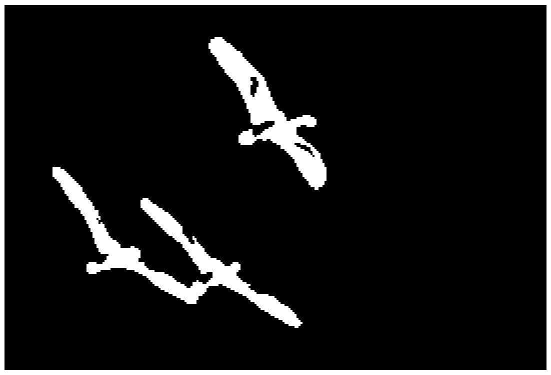


Figure 11: *Binary image 2*

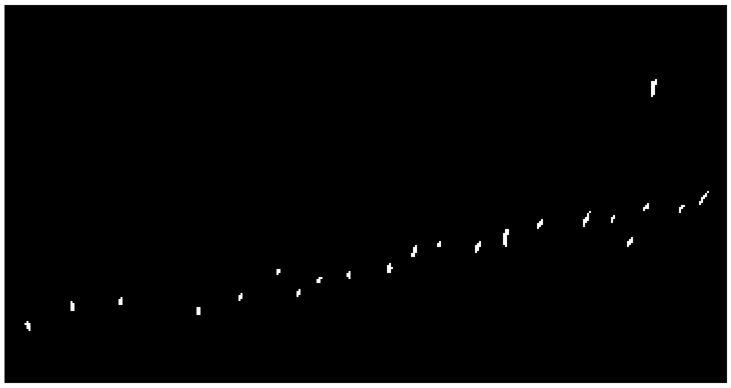


Figure 11: Binary image 3

STEP 5: Find Connected Region & Number of Birds

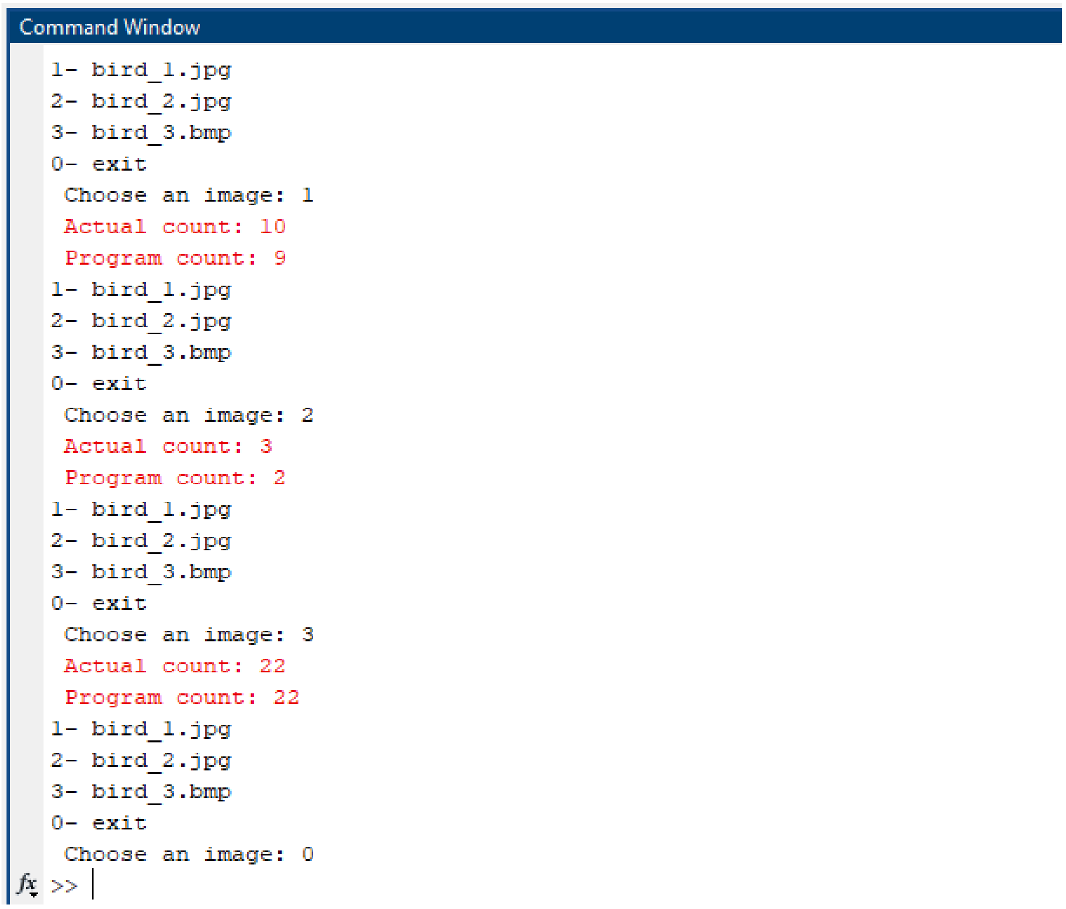
In this step, all connected components are found in the binary image. 8-neighbourhood of pixels where there exists a neighbour pixel (SE, NE, NW, SW are considered as neighbour as well) are used. In this algorithm, the image is traversed and whenever a nonzero pixel is detected, an id value is assigned to that location. The same id value is continuously assigned with the help of a depth-first search algorithm. When we are done with the segment block, the same strategy is applied for the next segment block, but this time id is incremented.

With this algorithm, each pixel in the image is grouped with its corresponding id, and the id shows in which segment block the pixel is. The pixels having zero value do not belong to any connected region. For that algorithm, a stack and two lists (same size of image) are needed. The first list contains the information of whether the current pixel is visited or not. The second list keeps connected component information for each pixel.

During the traversal of the connected region algorithm, the id value is incremented by one for each segment block. Thus, the number of birds is the current id value subtracted by one. Since the algorithm starts assigning id values beginning from one, subtraction is necessary.

STEP 6: Output & Comparison

Finally, the program prints the actual number of birds and the calculated number of birds by the algorithm with a red marker for the selected image.



```
1- bird_1.jpg
2- bird_2.jpg
3- bird_3.bmp
0- exit
 Choose an image: 1
 Actual count: 10
 Program count: 9
1- bird_1.jpg
2- bird_2.jpg
3- bird_3.bmp
0- exit
 Choose an image: 2
 Actual count: 3
 Program count: 2
1- bird_1.jpg
2- bird_2.jpg
3- bird_3.bmp
0- exit
 Choose an image: 3
 Actual count: 22
 Program count: 22
1- bird_1.jpg
2- bird_2.jpg
3- bird_3.bmp
0- exit
 Choose an image: 0
>> |
```

Figure 12: Output of the program

3-) RESULTS & DISCUSSION

The experimental result for the 3 input images is shown in Table 1.

Image	bird_1.jpg	bird_2.jpg	bird_3.bmp	Average
Actual count	10	3	22	-
Program count	9	2	22	-
TP	9	2	22	-
TN	0	0	0	-
FP	0	0	0	-
FN	1	1	0	-
Accuracy	0.9	0.67	1	0.86
Weighted Accuracy	10 * 0.19	3 * 0.67	22 * 1	0.94

Table 1: Obtained results from the program

True Positive (TP): The program detects the real bird correctly.

False Negative (FN): The program does not detect the real bird.

Accuracy: (TP + TN) / (TP + TN + FP + FN)

The program can segment the birds successfully in many cases and the weighted accuracy is 94% which can be acceptable. In the table, for images 1 and 2, there are FN values. It is because of overlapping birds. The program counts two overlapping birds as one. The overlapping birds could be separated using erosion & dilution however, selecting a structural element is tough since using generic values are mandatory in this assignment. In the second image there exist two birds very close to each other and when blurring applied to this image they overlapped. The use of the generic approach affects negatively in this case as the same sigma value is used in the Gaussian filter part. Finally, for the third image, everything goes very well. Occluded birds are separated using the filter and they are counted successfully using the connected component algorithm.