Computer Vision

HOMEWORK I REPORT

İpek Erdoğan

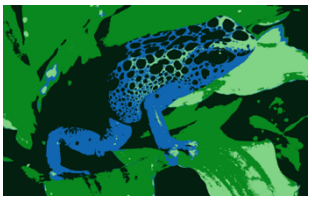
2019700174

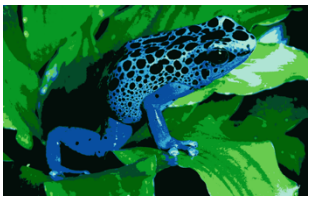
Chapter 1: Color Quantization

Results with randomly selected color centers


Original


k=2


k=4


k=8


k=16


Original


k=2


k=4


k=8


k=16


k=32


Original


k=2

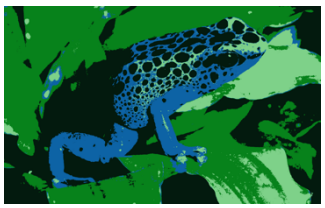

k=4

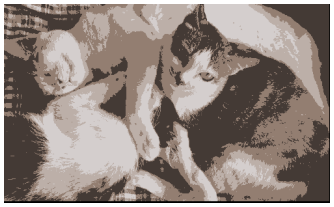

k=8


k=16


k=32

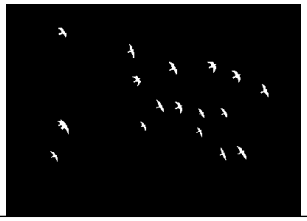
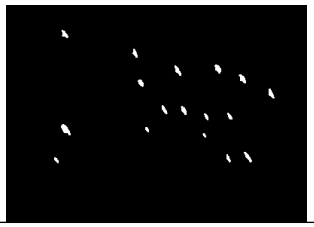
Results with color centers selected with clicking





Chapter 2: Connected Component Analysis

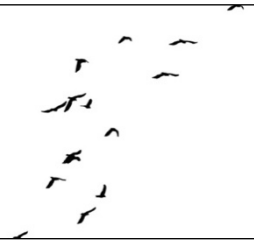
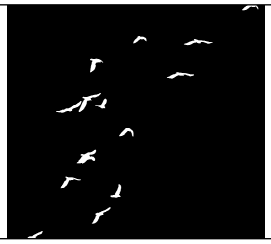
Original | Binary Image | Input of the Connected Component Analysis function

Birds1: Counted: 18 Blur + Closing with 4,4 kernel have been used.



Birds2: Counted: 14 Blur+Opening with 10,9 kernel have been used.


Birds3: Counted: 14 Blur+Erosion with 9,9 kernel have been used.



Dice5: Counted: 6 Blur + Dilation + Erosion + Opening with kernel 10,10 have been used.



Dice6: Counted: 6 Dilation + Erosion + Opening with kernel (9,10) have been used.
Threshold was zero to get rid of the shadowed background.



Dice6-2: Counted: 6 Dilation + Erosion + Opening with kernel (9,10) have been used.
Threshold was zero to get rid of the shadowed background.

By using K-means algorithm, I basically determined new color centers iteratively and at the end I equalized the points' color values to the center values of the clusters they assigned to. Color quantization is important if we have limited number of colors to show our image, or if we aim to do compression.

Fort Part 2, I implemented a connected component analysis algorithm. It traverses the image pixels and checks 8 of it's neighbors each time. If there is a neighbor which is assigned to a label, the element takes this label too. If not, it creates a new class label and assign this label to it's foreground neighbors. You can see more detailed explanations in my code comments. Also, I made some preprocessing before I give the image to the connected component analysis function. I turned my image into a binary image (with different thresholds for different examples). And then I implemented some of the morphological operations to the images. These operations are:

Erosion, Dilation, Opening, Closing, Top Hat, Black Hat and Blurring.

Note: Please see my code and code comments for more details about my K-means and Connected Component Analysis implementations.