CS 242 Discrete Structures for Comp Science Spring 2014

Programming project 1

(To be done in teams of two. Due: March 5, 2014)

In this project you will be using the tools that you learned in chapter 2 to track the movement of a small crab in a video. You will find the set of pictures under the recources page of the course web site. Also I have included some of the python code I used to generate the images in this document. If you use the scipy and numpy modules, most of the work is already done for you.

First we begin by defining the sets we will be working with for this project and understand their relations to each other. Define the set:

- A the set of all images,
- B the set of all gray scale images, and
- C the set of binary images.

. In Figure 1 there are examples of images from our videos in sets A, B, and C

```
import numpy as np
from scipy import ndimage, misc
from copy import copy
import matplotlib.pyplot as plt

first = misc.imread('../crab1Frames/crab1001.jpg') #read the image
firstGray = copy(first[:,:,1]) #project the image
firstBlackWhite = firstGray.copy()
firstBlackWhite = firstBlackWhite<=80  #threshold the image

#save the images.
misc.imsave('first.png',first)
misc.imsave('firstGray.png',firstGray)
misc.imsave('firstBlackWhite.png',firstBlackWhite)
```

Figure1: Examples of elements from set A, B, and C.

A color image that has $m \times n$ pixels can be thought of as a set of three $m \times n$ matrices. The three matrices represent the red-color-plan, green-color-plan, and blue-color-plan. Elements in ith row and jth column in each matrix representing the intensity of that color at pixel ij. We use the notation for $a \in A$, $a[i,j,k]$ is either an integer from 0 to 255 or a real number for 0 to 1 representing the intensity of the pixel located in the row i and column j and color plain k. Similarly for images $b \in B$ we denote them $b[i,j]$ to be the gray scale intensity at pixel $[i,j]$.

deskPDF Studio Trial

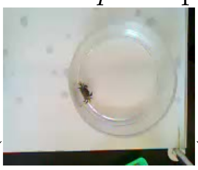
Sets

1. Use set builder notation to represent the set A, B, and C.
2. Show that $B \subseteq C \subseteq A$.

Functions

Before we start working with the entire set of images it is easier to get all our functions to work with one first. In this section we will develop the functions that we will have to use.

Projection $p()$ Converts a color image to a gray scale image. One can think of this in the sense of a projection as defined in section 9.2 of the text as follows. Let M be the set of all $m \times n$ matrices. We can then think of the set of color image A as a 3-array relation on M (i.e $A \subset M \times M \times M$). Also the set of gray scale images is a 1-relation on M. Thus the function p is a projection that maps A to B by restricting to one of the color plains.

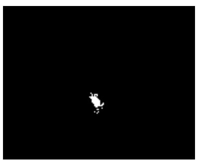
$p($  $) =$ 

Threshold $T_t()$ Converts a gray scale image to a binary image by setting all pixel values below a threshold value equal to 0 and those above equal to 1. We can think of this as making a matrix $cinC$ by $c = T_t(b) = B \leq t$ were the $\leq$ means to return true or false for each element. We choose t so we end up with the subject of interest being true.

$T_t($  $) =$ 

Crop $m()$ Forms a new image out of a subset of pixels from the original image. This simply takes a sub set of the pixels. As you have seen in our binary image there are other true regions than just the crab we can use cropping to get rid of them (i.e remove rows an columns.

```
firstCroped = copy(firstBlackWhite[:-20:,:-20])
misc.imsave('firstCroped.png',firstCroped)
```

$m($  $) =$ 

Dilation $d_B()$ This function operates on each point in a binary image by setting it to true if any pixels within an given radius of it are true. See the section below on matrices to see how to implement this function.

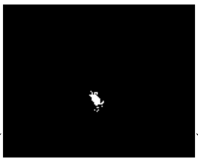
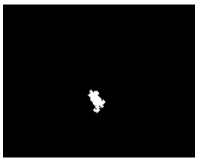
```
firstDilation = firstCroped.copy()

#This is the B matrix described in the matrix section of the HW.
struct1 = ndimage.generate_binary_structure(2,1)
```

deskPDF Studio Trial

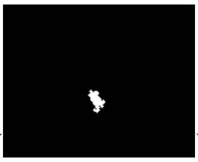
```
firstDilation = ndimage.binary_dilation(firstDilation, structure=struct1)

misc.imsave('firstDilation.png',firstDilation)
```

$m($  $) =$ 

CenterFinder $c()$ Finds the center of the connected component (i.e. the crab). Since we only have one connected component we can find the center by averaging the location of all the true pixels and dividing by the number of true pixels. That is if S is the set of all location (i, j) of true pixel values this function returns

$$\left(\frac{1}{|S|} \sum_{(i,j)\in S} i, \frac{1}{|S|} \sum_{(i,j)\in S} j \right).$$

$p($  $) = (23, 155)$

3. Define each of the functions above using function notation including its domain and co-domain.

4. What is the difference between co-domain and range?

5. Are any of the above functions one to one, onto, or bijective? Explain your reasoning.

6. What does the function $m \circ T_t$ do? What is the domain and range of $m \circ T_t$?

7. Write a program capable of implementing each of the above functions and test it with the first image in the series of crab images. Show your results of each. You may use the dilation function from the scipy module to make things faster. Your submission should include a screen capture of the output for each of the functions.

Sequences

Now we are going to be working with the entire sequence of crab images. Given time series of pictures of the crab we define the sequence a_n to be the series of ordered images where n is the order in which the picture was taken. Then we can use a function f to define a new sequence $b_n = f(a_n)$. We will be applying all the the functions above to each image in the sequence so you may want to form your sequence only with every fifth image to make your code run faster.

8. Use a Venn diagram to represent the transformation from a sequence in the set A to a sequence into the set B using the function $p \cdot t$.

9. What (qualitatively) is the sequence $c_n = c(d_B((T_t(p(a_n)))))$?

10. Write a program to calculate c_n and plot c_n with lines connecting c_n and $c_n - 1$ for all n.

11. Use the distance formula to write a function (s) to calculate the distance between c_n and c_{n-1} and use it to create a new sequence d_n = s(c_n)

12. Write a program to calculate d_n.

deskPDF Studio Trial

13. Use summation notation, d_n, and the work from the previous question to write the distance the crab moved in the time series. Write a program to calculate the total distance, in pixels, that the crab moved. If you know the bowl that the crab is in is 6 inches could you determine how far the crab moved?

Matrices

14. Define the shift function $S_{ij} : C \to C$ such that $S_{ij}(c)$ shifts all the rows of c down by i rows and every column to the right by j columns but keeps the matrix the same size by adding the proper number of 0 rows and columns to replace those shifted away. For example: If $c = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$ Then

$$S_{-1,2}(c) = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \text{ and } S_{1,1}(c) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

Now for a binary matrix B we define the set of pixels of B to be the set of true pixels (i.e $(i, j) \in B$ means $b_{ij} = 1$). Now given a $2k + 1 \times 2l + 1$ matrix B We can then define a dilation $d_B()$ in terms of unions of matrices as

$$d(A) = \cup_{(i,j)\in B} S_{i-(k+1),j-(l+1)}(c)$$

For this project we will use $B = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$. Using c and B as defined above calculate $d_B(c)$

deskPDF Studio Trial