**Image – Worksheet 1**

# CDS 230 Spring 2018

## 1 Problem

Using functions from the *scipy* package, load the image bird.jpg from the data section of the course. In Chapter 15 of the textbook, code was presented that enabled you to isolate the elements of a matrix of distance 10 from a central point. Since an image is just a matrix, create a filter that isolates the pixels around the birds head in the image. You will have to provide a best guess as to where the center of the birds head is. Save this as a new image, where only the birds head appears, perhaps in a circle or square, and everything else in the image is black.

import scipy.misc as sm

import numpy as np

from PIL import Image, ImageOps, ImageDraw

def parrot\_head():

# mask = Image.open("bird.jpg").convert("L")

im = Image.open("bird.jpg")

im = im.crop((200,60,320,180))

bigsize = (im.size[0] \* 3, im.size[1] \* 3)

mask = Image.new('L', bigsize, 0)

draw = ImageDraw.Draw(mask)

draw.ellipse((0, 0) + bigsize, fill=255)

mask = mask.resize(im.size, Image.ANTIALIAS)

im.putalpha(mask)

output = ImageOps.fit(im, mask.size, centering=(0.5, 0.5))

output.putalpha(mask)

output.save('output.png')



## 2 Problem

Create a new image from bird.jpg by swapping the color channels. To accomplish this, the pixel values for the green channel were placed in the red channel, the pixel values for the blue channel were placed in the green channel, and the original pixel values for the red channel were placed in the blue channel. Things that were red in the original are blue in the output.

def colorswap():

import scipy.misc as sm

img = sm.imread("bird.jpg")

red = img[:,:,0].copy()

green = img[:,:,1].copy()

blue = img[:,:,2].copy()

img[:,:,0]=green

img[:,:,1]=blue

img[:,:,2]=red

sm.imsave("colorswap.png", img)

colorswap()

