

C.J. Fulciniti

Completed



Original



```
#C.J. Fulciniti
#Final Edit Date: 18 October 2023
def collage():
    picture = makePicture(getMediaPath("starting_picture.jfif"))#starting picture
    #First Technique: Scaling, this time by 0.75 the original size to fit the project requirements.
    scaledDownPicture = allScale(picture,0.75)
    canvas = makeEmptyPicture(getWidth(scaledDownPicture)*2,getHeight(scaledDownPicture)*2,gray)
    #Second technique: Cropping, this time cropping the eyes of the owl and pasting them in the center of the canvas
    owlsEyes = cropOwlsEyes(scaledDownPicture)
    #Third technique: Grayscale, this is only applied to our new cropped Eyes for the owl.
    #These will be pasted later.
    grayOwlsEyes = grayscale(owlsEyes)
    #Fourth Technique: Mirroring, this time using the edge of the picture.
    firstMirrorLU = mirrorPartLeft(scaledDownPicture)
    firstPictureComplete = mirrorPartTop(firstMirrorLU)
    mirroredTemplateRL = duplicatePicture(firstPictureComplete)
    copyOG(firstPictureComplete,canvas,0,0)
    #Fifth Custom Technique: These functions remove red, green, or blue from different
    #sections of the photo to create an interesting effect.
    removeRedH(scaledDownPicture)
    #these functions were created to satisfy project requirements of writing a function
    #that is original and not from another source
    removeBlueH(scaledDownPicture)
    secondPictureComplete = removeGreenH(scaledDownPicture)
    secondPhotoCopy = copyOG(secondPictureComplete,canvas,getWidth(canvas)/2,0)
    removeRedV(mirroredTemplateRL)
    #these will do the same as the previous functions, but in vertical stripes
    removeBlueV(mirroredTemplateRL)
    thirdPictureComplete = removeGreenV(mirroredTemplateRL)
    copyOG(thirdPictureComplete,canvas,0,getHeight(canvas)/2)
    removeRedV(secondPictureComplete)
    #final photo will be removing r,g,b in a horizontal and vertical pattern.
    removeBlueV(secondPictureComplete)
    fourthPictureComplete = removeGreenV(secondPictureComplete)#
```

```

copyOG(fourthPictureComplete, canvas, getWidth(canvas)/2, getHeight(canvas)/2)
#copying our cropped post grayscale eyes
finalPictureAdded = copyOG(grayOwlsEyes, canvas, (getWidth(canvas)/2)-120, (getHeight(canvas)/2-45))
readyForSignature = duplicatePicture(finalPictureAdded)
#scaling used again, this time for the signature so it will fit
signature = makePicture(getMediaPath("collage_signature.jpg")) #signature file
scaledSignature = allScale(signature, 0.09)
#placing signature on the final collage
finalEdit = chromakeySignature(scaledSignature, readyForSignature)
explore(finalEdit)#check it out!

def chromakeySignature(signature, target):
    for pixel in getPixels(signature):
        x = getX(pixel)
        y = getY(pixel)
        #targeting exactly where to put the signature took 40+ tries to get right!
        targetPixel = getPixel(target, x + 573, y + 485)
        luminance = getRed(pixel) + getGreen(pixel) + getBlue(pixel)
        if (luminance <= 300): #call back to luminance from earlier chapters
            signaturePixels = getPixel(signature, x, y)
            #I used orange because I love orange, but it also will show up better in the
            #collage.
            setColor(targetPixel, orange)
    return(target)

def grayscale(picture):
    for px in getPixels(picture):
        newRed = getRed(px) * 0.299
        newBlue = getBlue(px) * 0.114
        newGreen = getGreen(px) * 0.587
        lum = newRed + newGreen + newBlue
        setColor(px, makeColor(lum, lum, lum))
    return(picture)

def cropOwlsEyes(picture):
    canvas = makeEmptyPicture(230, 90)
    targetX = 0
    for sourceX in range(40, 270):
        targetY = 0
        for sourceY in range(140, 230):
            sourceColor = getColor(getPixel(picture, sourceX, sourceY))
            setColor(getPixel(canvas, targetX, targetY), sourceColor)
            targetY = targetY + 1
        targetX = targetX + 1
    return(canvas)

```

```

def mirrorPartLeft (picture):
    mirrorPoint = getWidth (picture) / 10
    for x in range (0, mirrorPoint):
        for y in range (0, getHeight (picture)):
            leftQ = getPixel (picture, x, y)
            targetQ = getPixel (picture, mirrorPoint - 1 - x, y)
            setColor (targetQ, getColor (leftQ))
    return (picture)

def mirrorPartTop (picture):
    mirrorPoint = getHeight (picture) / 10
    for x in range (0, getWidth (picture)):
        for y in range (0, mirrorPoint):
            topQ = getPixel (picture, x, y)
            targetQ = getPixel (picture, x, mirrorPoint - 1 - y)
            setColor (targetQ, getColor (topQ))
    return (picture)

def removeBlueH (picture):
    for x in range (0, getWidth (picture) / 3):
        for y in range (0, getHeight (picture)):
            pixel = getPixel (picture, x, y)
            setBlue (pixel, 0)
    return (picture)

def removeGreenH (picture):
    for x in range (getWidth (picture) / 3, getWidth (picture) / 3 + getWidth (picture) / 3):
        for y in range (0, getHeight (picture)):
            pixel = getPixel (picture, x, y)
            setGreen (pixel, 0)
    return (picture)

def removeRedH (picture):
    for x in range (getWidth (picture) / 3 + getWidth (picture) / 3, getWidth (picture)):
        for y in range (0, getHeight (picture)):
            pixel = getPixel (picture, x, y)
            setRed (pixel, 0)
    return (picture)

```

```

def removeBlueV(picture):
    for x in range(0,getWidth(picture)):
        for y in range(getHeight(picture)/3 + getHeight(picture)/3,getHeight(picture)):
            pixel = getPixel(picture,x,y)
            setBlue(pixel,0)
    return(picture)

def removeGreenV(picture):
    for x in range(0,getWidth(picture)):
        for y in range(getHeight(picture)/3,getHeight(picture)/3 + getHeight(picture)/3):
            pixel = getPixel(picture,x,y)
            setGreen(pixel,0)
    return(picture)

def removeRedV(picture):
    for x in range(0,getWidth(picture)):
        for y in range(0,getHeight(picture)/3):
            pixel = getPixel(picture,x,y)
            setRed(pixel,0)
    return(picture)

def allScale(picture,uNumber):
    frameForScaling = makeEmptyPicture(int(getWidth(picture)*uNumber),int(getHeight(picture)*uNumber),green)
    sourceX = 0
    for targetX in range(0,int(getWidth(picture) * uNumber)):
        sourceY = 0
        for targetY in range(0,int(getHeight(picture) * uNumber)):
            color = getColor(getPixel(picture,int(sourceX),int(sourceY)))
            setColor(getPixel(frameForScaling,targetX,targetY),color)
            sourceY = sourceY + 1.0/uNumber
            sourceX = sourceX + 1.0/uNumber
    return(frameForScaling)

def copyOG(picture,canvas,startPixelX,startPixelY):
    targetX = startPixelX
    for x in range(0,getWidth(picture)):
        targetY = startPixelY
        for y in range(0,getHeight(picture)):
            pixelsToCopy = getPixel(picture,x,y)
            copyColor = getColor(pixelsToCopy)
            setColor(getPixel(canvas,targetX,targetY),copyColor)
            targetY = targetY + 1
            targetX = targetX + 1
    return(canvas)

```