影像處理 hw1

資科碩一 111753147 羅揚

Q1.

Output: “Q1_answer.png”, the horizon combination picture of “laptop_right.png” and “laptop_left.png”

Explain:

(1) Import “cv2” package.

(2) Use the function ‘imread’ in cv2 to read the picture ‘laptop_left.png’ and ‘laptop_right.png’

(3) Combine two picture above by function ‘hconcat’ to obtain new picture and save it as "Q1_answer.png".

Q2.

Input: (Image, degree)

Output: “Q2_answer.png”, the picture rotates 15 degrees

Explain:

1. Set a function “image_rotate (image,degree)”
2. Take the high and width of the image by “shape” function
3. Create an zeros pixel value image with ($\sqrt{high^2 + width^2}$, $\sqrt{high^2 + width^2}$) as. high and width

4.Rotate the image from the center of the picture, by compute $\begin{bmatrix} cos\theta & sin\theta & 0 \\ -sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} high \\ width \\ layer \end{bmatrix}$,however, since the rotated picture will get out of the image so we set the rotating center as parameter (middle x // $\sqrt{2}$, middle y // $\sqrt{2}$).

5.Output the rotating image.

Q3.

Input: (image, new_high, new_width)
Output: “Q3_answer.png”, new image with new_high and new_width

Explain:

(1) Define a function “resize (image, new_high, new_width)” which input the target image and the size you wonder it become.

(2) In “resize” function, set a new image which pixel values are zeros, new_high and new_width as high and width.

(3) Set the x_scale, y_scale be the ratio between new_width and old_width, new_high and old_scale.

(4) If the pixel is on the four corner, then new_image[I,j,k] = image[i,j,k]
Else we use the Bilinear interpolation method to get the pixel value.
(Set the pixel value of the corner as the values using in Bilinear interpolation)

(5) Then we input “lena.bmp” as image to function resize(image,1024 1024) to get the 1024*1024 size lena image.

Q4.

Input: (under_image, top_image)
Output: “Q4_image.png”, an image which top_image is overlay on ther under_image

Explain:

1. First, set a function “remove_bg (image)”, to remove the background of the image, by the “remove” function and save it.
2. Second, set a function ”overlay (under_image, top_image)” function to the the top_image to overlay on the under_image.
3. In function ”overlay (under_image, top_image)”, I create a pixel values zeros image with same size as the under_image.
4. Let the t_h be the high of top_image, t_w be the width of top_image
5. Let the new_image[I,j,:] == top_image[I,j,:] if i<t_h && j<t_w && k<t_c && !(all(top_image[I,j,:]==0))
Else new_image[I,j,:] == under_image[I,j,:]
6. Then we got the new_image which top_image is on the under_image.

Q5.(1)

Watermark part:

Input: (original image, watermark)

Output: “Q5_1_water.png” image with watermark

Explain:

1. Create an zeros pixel values image with same size as original image denote the image as **new_img**
2. Copy the watermark, divide the pixel values by 32(We only take 3 bits of watermark) denote the image as **wm**
3. Copy the original image, divide the pixel values by 8 and multiple the pixel value by 8 (Set the last 3 bits of original values as 000) denote the image as **ori_img**
4. Let **new_img**[I,j,k] = **wm**[I,j,k] + **ori_img**[I,j,k] if I,j,k is in the size of **wm (**embed the watermark), else **new_img**[I,j,k] = **ori_img**[I,j,k]
5. Obtain the watermark image.

Extract part:

Input: watermark image

Output: watermark

1. Take the watermark image denote as **wm_img**
2. For I,j,k in the size of **wm_img** , **wm_img**[I,j,k]%8*32 (Get the last 3 bits of watermark image and recover it)

3.Obtain the similar image as watermark.

Q5.(2)

Input: watermark image(denote as wim)

Output:

PSNR between original image and decode of wim with compressed ratio 10,

PSNR between original image and decode of wim with compressed ratio 50,

PSNR between original image and decode of wim with compressed ratio 100,

Extract image of decode of wim with compressed ratio 10,

Extract image of decode of wim with compressed ratio 50,

Extract image of decode of wim with compressed ratio 100,

Explain:

1. Define the function “PSNR”, if inputs two image, then can obtain the psnr value of two image.
2. Get the array of watermark image save it as compression ratio of 10,50,100 by function “imencode” in “cv2” package
3. Decode the encoding image with compression ratio 10,50,100 by function “imdecode” in “cv2” package
4. Use “PSNR” function to obtain the psnr values between decoding images and original image.
5. Use the “extract” function in Q5(1) to retrieve watermark, however, we can’t retrieve watermark both after encode and decode.