

You can talk with each other, but submit your own independent work.

1. Read the complete HW, and estimate how many hours you think it will take to complete.
 Once you get the first one working, you should be able to re-use many of the concepts in the next problems. (1/2)

2. Exploring Spatial Quantization: (3)

Write a function called `Check_Resizing_Nearest(fn)`.

The objective here is to see what happens when you reduce the pixel resolution of an image.

It takes one parameter, the filename of an image.

The function performs the following tasks:

- a. Read in the given image.

Sub-samples the input image by factors of:

[1 2 4 8 16 24 32 48 50 52 56 64 128]

- b. For each of those sub-sampling factors:

- i. Compute the new image size as
`round(original_image_size / sub-sampling)`

- ii. Resize the image down to this new height and width.

Use nearest neighbor option for `imresize`:

`im_small = imresize(im, newSize, 'nearest')`

- iii. Then up-sample the image back to the original size, again use 'nearest':

- iv. `im_restored = imresize(im_small, originalSize, 'nearest')`

- v. Display the restored image, using `imshow`.

- vi. Put a title over it saying 'Sub-Sampling is %d '.

- vii. Pause for 1.5 seconds

- c. Run the program on the following supplied images:

TBK_Kite.jpg, TBK_BRICKS.jpg, football.jpg, peppers.png, finger_prints.png, kod_kid.png, kod_parrots.png, and Might_or_Might_NOT.jpg.

- d. Fill in the following table:

| Image Name | Sub-Sampling Where Image Details Lost | Sub-Sampling at which Aliasing (Jagged Edges) Occurred | What details were most important that were lost? |
|-----------------------------|---------------------------------------|--|--|
| TBK_Kite | | | |
| TBK_BRICKS | | | |
| football.png (matlab image) | | | |
| peppers.png (matlab image) | | | |
| kod_kid.png | | | |
| kod_parrots.png | | | |
| Might_or_Might_NOT.jpg | | | |

- e. Write a modified version of your program, called `Check_Resizing_Default(fn)`.

It is the same as the previous program, but remove both uses of 'nearest' in `imresize`.

What changes to you observe?

3. Exploring Intensity Quantization: (3)

Write a function called CHECK_QUANTIZATION_HW(fn).

The objective here is to see what happens when you reduce the number of levels of pixel values for an image.

It takes one parameter, the filename of an image.

The function performs the following tasks:

- a. Read in the given image. Leave it as an uint8.

Quantize the input image by factors of:

[1 2 4 8 16 32 40 50 60 70 80 90 100 128]

- b. For each of those quantization factors:

- Compute the new image as:
 $\text{new_im} = \text{round}(\text{original_image} / \text{quantization})$;
- Restore the dynamic range:
 $\text{im_restored} = \text{new_im} * \text{quantization}$;
- Display the restored image, using imshow.
- Put a title under it saying 'Quantization Factor = %d '.
- Pause for 1.5 seconds

- c. Run the program on the following supplied images:

TBK_Kite.jpg, TBK_BRICKS.jpg, football.jpg, peppers.png, finger_prints.png, kod_kid.png, kod_parrots.png, and Might_or_Might_NOT.jpg.

- d. Fill in the following table:

| Image Name | Quantization Where Image Details Lost | What else weird did you notice? |
|--------------------------------|---|---------------------------------|
| TBK_Kite | | |
| TBK_BRICKS | | |
| football.png (matlab image) | | |
| peppers.png (matlab image) | | |
| kod_kid.png | | |
| kod_parrots.png | | |
| Might_or_Might_NOT.jpg | | |

- e. Which type of image suffered the most?

4. Contrast Control: (3)

You are provided with an image called 'Might_or_Might_NOT.jpg'.

When the image was taken, you could not see into the shadows to see the backs of the boats.

a. For each of the following techniques:

imadjust, histeq, and adaptive histogram equalization

i. Read in the image and display the image histogram:

`figure;`

`subplot(2,2,1);`

display the original image.

ii. `subplot(2,2,3);`

display the original histogram.

iii. fix the image tone.

iv. `subplot(2,2,2);`

display the fixed image.

v. `subplot(2,2,4);`

display the new histogram.

b. What problem was there in the original image?

c. Which contrast modification method do you think works best, and why?

5. Report how many hours this actually took you. (1/2)