

Tutorial 1: Image Formation, Image Filtering

- Let (x, y) be the coordinate of a point in an image. Its homogeneous coordinate $(x, y, 1)$ is often used in computer vision or graphics, for example, transforming an image by scaling, translation, rotation etc. The transformation can be described by the following equation,

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = A \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

where $(x, y, 1)$ and $(x', y', 1)$ denote the coordinate before and after transformation, A is a 3×3 transformation matrix. For the following examples of A , describe what kind of transformation it performs.

1.1 Example 1

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 10 \\ 0 & 1 & 5 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

1.2 Example 2

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} 5 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

- For a RGB image, at each pixel, there are intensity values for three channels (R: red; G: green; B: blue). For example, $[255, 0, 0]$ represents pure red. $[0, 255, 0]$ represents pure green. $[0, 0, 255]$ represents pure blue. Each channel is represented by an integer between 0 and 255, i.e. an 8-bit unsigned char.

2.1 For a RGB image of size 1280x960 pixels, without image compression, how many bytes are needed for storing the image?

2.2 There are algorithms to convert a RGB image into a grayscale image, where each pixel only has one intensity value, which represents the brightness. For example, one algorithm is recommended by ITU-R Recommendation BT.601, which is formulated as the following equation,

$$Y = 0.299R + 0.587G + 0.114B$$

Could you work out what grayscale values pure red, pure green and pure blue respectively convert to?

- Suppose that we have an image that is corrupted by Gaussian white noise $Y = I + n$, where I denotes the clean image, $n \sim N(0, \sigma^2)$ denotes the Gaussian white noise and Y denotes the corrupted image. Perform image filtering using a low-pass filter is one approach for denoising, but it may also result in loss of fine details. Another approach is to denoise to take a lot of images of

the same object and combine them. For example, you can take a lot of pictures of the moon in the sky from the same angle. Each image is described by $Y_i = I + n_i$ ($i = 1, 2, \dots, N$), where n_i is one independent realisation of the noise. Then you can take the average, $Y = \frac{1}{N} \sum_{i=1}^N Y_i$. Please derive the mean and variance of the noise in the combined image Y .

4. Consider a 2D Gaussian filter

$$h(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

4.1 Derive its first derivative $\frac{\partial h}{\partial x}$ and $\frac{\partial h}{\partial y}$.

4.2 The Gaussian filter has an infinitive support. People often truncate and approximate the filter. For example, the following 3x3 filter approximates the 2D Gaussian filter,

1/36	1/9	1/36
1/9	4/9	1/9
1/36	1/9	1/36

The following filter approximates the 1D Gaussian filter,

1/6	2/3	1/6
-----	-----	-----

Please check that whether the 2D Gaussian filter is equivalent to the convolution of a 1D Gaussian filter along x-axis and a 1D Gaussian filter along y-axis.

4.3 What is the computational cost when we convolve an NxN image with a 3x3 2D Gaussian filter?

4.4 What is the computational cost when we convolve an NxN image with two 1D Gaussian filters, respectively with size 1x3 and 3x1?

4.5 In general, what is the computational cost when we convolve an NxN image with a KxK 2D Gaussian filter? Use the big O notation.

4.6 What is the computational cost when we convolve an NxN image with two 1D Gaussian filters, respectively with size 1xK and Kx1?