Introduction to Image Processing

1 Image Representation

1.1 Continuous-domain, discrete-domain, and finite-size images

An *image* is a spatially varying signal $s(x, y)$ where x and y are two *spatial* coordinates. The signal value $s(x, y)$ at each spatial location (x, y) can be either a scalar (e.g. light intensity for gray scale images) or a vector (e.g. 3 dimensional vector for RGB color images, or more general P-dimensional vector for multispectral images). In the latter case, we could treat each vector component separately as a scalar image (referred to as a channel).

In digital image processing, images are discretized into samples at discrete spatial locations that are indexed by integer coordinates $[m, n]$. Typically, a *discrete-domain* image $s[m, n]$ is related to a *continuous-domain* image $s(x, y)$ through the *sampling* operation

$$s[m, n] = s(m\Delta_x, n\Delta_y), \tag{1}$$

where Δ_x and Δ_y are sampling intervals in x and y dimensions, respectively. More general, a discrete-space image is obtained through the generalized-sampling operation

$$w[m, n] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} s(x, y)\phi_{m,n}(x, y)\mathrm{d}x\mathrm{d}y, \tag{2}$$

where $\phi_{m,n}(x, y)$ is the *point-spread function* of the image sensor (e.g. a photometric sensor in a digital camera) at the location indexed by (m, n). Typically, point-spread functions at different locations are simply shifted versions of a single function as

$$\phi_{m,n}(x, y) = \phi(x - m\Delta_x, y - n\Delta_y), \tag{3}$$

and $\phi_{m,n}(x, y)$ is called the sampling kernel.

Furthermore, a discrete image $s[m, n]$ is often of *finite* size; for example $0 \leq m \leq M - 1,\ 0 \leq n \leq N - 1$. Then $s[m, n]$ can also be treated as an $M \times N$ matrix. The image sample $s[m, n]$ and the corresponding location $[m, n]$ is often called a *pixel*, or picture element.

1.2 Fourier transforms and sampling theorem

It is often very effective, conceptually and computationally, to represent images in the *frequency domain* using the *Fourier transform*. For a continuous-domain image $s(x, y)$, its Fourier transform is defined as

$$S(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} s(x, y)e^{-j2\pi(xu+yv)}\mathrm{d}x\mathrm{d}y. \tag{4}$$

Here, u and v denote frequency variables and they have reciprocal unit with x and y. For example, if the spatial coordinate x has unit in mm, then the corresponding frequency variable u has unit in mm^{-1}. Under certain conditions, the image $s(x, y)$ can be exactly recovered from its frequency-domain $S(u, v)$ by the *inverse Fourier transform*

$$s(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} S(u, v) e^{j2\pi(xu+yv)} \mathrm{d}u \mathrm{d}v. \tag{5}$$

We denote this pair of signals related by the Fourier transform (FT) as

$$s(x, y) \overset{\text{FT}}{\longleftrightarrow} S(u, v). \tag{6}$$

For a discrete image $s[m, n]$ the *discrete-space Fourier transform* (DSFT) relation

$$s[m, n] \overset{\text{DSFT}}{\longleftrightarrow} S(u, v). \tag{7}$$

is defined as

$$S_d(u, v) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} s[m, n] e^{-j2\pi(mu+nv)}, \tag{8}$$

$$s[m, n] = \int_{-1/2}^{1/2} \int_{-1/2}^{1/2} S_d(u, v) e^{j2\pi(mu+nv)} \mathrm{d}u \mathrm{d}v. \tag{9}$$

It is easy to see that $S_d(u, v)$ is a periodic function

$$S_d(u + k, v + l) = S_d(u, v), \quad \text{for all } k, l \in \mathbb{Z},$$

and thus we only need to consider the function in one period; e.g. $S_d(u, v)$ with $|u| \leq 1/2,\ |v| \leq 1/2$.

Theorem 1 (Sampling)**.** *Suppose that the discrete-domain image $s[m, n]$ is related to the continuous-domain image $s(x, y)$ through the sampling operation* (1)*. Then their Fourier transforms are related by*

$$S_d(u, v) = \frac{1}{\Delta_x \Delta_y} \sum_{k \in \mathbb{Z}} \sum_{l \in \mathbb{Z}} S\left(\frac{u + k}{\Delta_x}, \frac{v + l}{\Delta_y}\right). \tag{10}$$

Proof. (Sketch) One way to prove this is to express $s[m, n]$ using (5) by substituting $x = m\Delta_x,\ y = n\Delta_y$ and then "match" with the right-hand side of (9). □

The summation on the right-hand side of (10) consists of $S(u/\Delta_x, v/\Delta_y)$ and its translated copies in frequency by (k, l). These copies with $(k, l) \neq (0, 0)$ are called *alias* terms. If $s(x, y)$ is *bandlimited* such that

$$S(u, v) = 0, \quad \text{for } |u| \geq 1/(2\Delta_x),\ |v| \geq 1/(2\Delta_v), \tag{11}$$

then these alias terms do not overlap with $S(u/\Delta_x, v/\Delta_y)$, and thus $S(u, v)$ can be exactly recovered from $S_d(u, v)$ simply by

$$S(u, v) = \Delta_x \Delta_y \ \mathrm{rect}(\Delta_x u)\ \mathrm{rect}(\Delta_y v)\ S_d(\Delta_x u, \Delta_y v). \tag{12}$$

Here the rectangular function is defined as

$$\mathrm{rect}(x) = \begin{cases} 1 & \text{if } |x| \le 1/2 \\ 0 & \text{else.} \end{cases}$$

We can show that (12) in the spatial domain is equivalent to the following interpolation formula in the spatial domain:

$$s(x,y) = \sum_{k\in\mathbb{Z}} \sum_{l\in\mathbb{Z}} s[m,n] \,\mathrm{sinc}(t/\Delta_x - m)\,\mathrm{sinc}(t/\Delta_y - n), \tag{13}$$

where the sinc function is defined as

$$\mathrm{sinc}(x) = \frac{\sin(\pi x)}{\pi x}.$$

For the discrete image $s[m,n]$ of finite size $M \times N$ with $0 \le m \le M-1,\ 0 \le n \le N-1$, we have the *discrete Fourier transform* (DFT) relation

$$s[m,n] \overset{\text{DFT}}{\longleftrightarrow} S[k,l], \tag{14}$$

which is defined as

$$S[k,l] = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} s[m,n] e^{-j2\pi(mk/M + nl/N)}, \tag{15}$$

$$s[m,n] = \frac{1}{MN} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} S[k,l] e^{j2\pi(mk/M + nl/N)}. \tag{16}$$

Therefore the DFT maps an $M \times N$ image in the spatial domain into an $M \times N$ image in the frequency domain; both images can have complex values.

Relating (15) to (8), we see that if the $M \times N$ image $s[m,n]$ is zero padded outside its support $[0, M-1] \times [0, N-1]$ then $S[k,l]$ is a sampled image of $S_d(u,v)$,

$$S[k,l] = S_d(k/M, l/N). \tag{17}$$

In summary, we have seen the following three Fourier transforms

$$\begin{aligned} \text{continuous-domain} &\overset{\text{FT}}{\longleftrightarrow} \text{continuous-domain} \\ \text{discrete-domain} &\overset{\text{DSFT}}{\longleftrightarrow} \text{continuous-domain} \\ \text{discrete-domain} &\overset{\text{DFT}}{\longleftrightarrow} \text{discrete-domain} \end{aligned}$$

Among these transforms, only the last one, the DFT, is computationally feasible (i.e. with summations of finite terms). Moreover, the DFT can be implemented efficiently with fast Fourier transform algorithms. In moving from the FT to the DSFT and then to the DFT, we first discretize the spatial domain and then the frequency domain. Therefore, it is important to understand (12) and (17) so that we can relate the computational results and images by the DFT to the frequency representation of the original image in the real world.

2 Image Filtering

2.1 Convolution operations

The image (linear) *filtering* operation in the continuous-domain is defined by *convolution*

$$r(x,y) = (s*h)(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(x',y')\, s(x-x',y-y')\mathrm{d}x'\mathrm{d}y'. \tag{18}$$

And similarly in the discrete-domain:

$$r[m,n] = (s*h)[m,n] = \sum_{m'=-\infty}^{\infty}\sum_{n'=-\infty}^{\infty} h[m',n']\, s[m-m',n-n']. \tag{19}$$

The two-dimensional signal $h(x,y)$ or $h[m,n]$ is called *filter*, *mask*, or *point-spread function.*

2.2 Examples

Example 1 (First-order derivatives)**.** *The first-order derivatives in the x and y directions of a discrete image $s[m,n]$ can be approximated by finite differences*

$$\frac{\partial s}{\partial x} = s[m+1,n] - s[m,n] = (s*h_x)[m,n] \tag{20}$$

$$\frac{\partial s}{\partial y} = s[m,n+1] - s[m,n] = (s*h_y)[m,n], \tag{21}$$

which are convolutions with the following filters

$$h_x^{(1)} = \begin{pmatrix} 1 \\ \boxed{-1} \end{pmatrix}, \qquad h_y^{(1)} = \begin{pmatrix} 1 & \boxed{-1} \end{pmatrix}.$$

Here in the matrix form, row and column indexes correspond to x (first) and y (second) dimensions, respectively; and the sample in the box corresponds to the original (i.e. $(m,n)=(0,0)$).

Example 2 (Gausian smoothing filter)**.** *The two-dimensional Gaussian filter, which is often used for image smoothing, is defined as*

$$h_{Gauss}^{(2D)}(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}. \tag{22}$$

The 2D Gaussian filter is separable, which means it is a product of 1D filters in each dimension

$$h_{Gauss}^{(2D)}(x,y) = h_{Gauss}^{(1D)}(x)\, h_{Gauss}^{(1D)}(y), \quad \text{where} \quad h_{Gauss}^{(1D)}(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}.$$

Discrete-domain Gaussian filters used in practice are sampled and truncated versions of the above continuous-domain filters.

Example 3 (Edge detector). *The Sobel edge detector is obtained by smoothing the image in the perpendicular direction before computing the directional derivatives. The associate Sobel edge detector filters are given by*

$$h_x^{(Sobel)} = \begin{pmatrix} 1 & 2 & 1 \\ 0 & \boxed{0} & 0 \\ -1 & -2 & -1 \end{pmatrix}, \qquad h_y^{(Sobel)} = \begin{pmatrix} 1 & 0 & -1 \\ 2 & \boxed{0} & -2 \\ 1 & 0 & -1 \end{pmatrix}.$$

Edges are detected as pixels $[m, n]$ where the magnitude of the gradient is above a certain threshold T; i.e.

$$|(s * h_x^{(Sobel)})[m, n]| + |(s * h_y^{(Sobel)})[m, n]| \geq T.$$

2.3 Frequency response of a filter

A key result in signal and image processing is that convolution in the space domain becomes multiplication in the frequency domain

$$r(x, y) = (s * h)(x, y) \quad \overset{\text{FT}}{\longleftrightarrow} \quad R(u, v) = S(u, v)\ H(u, v), \tag{23}$$

$$r[m, n] = (s * h)[m, n] \quad \overset{\text{DSFT}}{\longleftrightarrow} \quad R_d(u, v) = S_d(u, v)\ H_d(u, v). \tag{24}$$

Therefore, the Fourier transform $H(u, v)$ of the filter, called *frequency response*, indicates how certain frequency components of the input image $s(x, y)$ are amplified or attenuated in the resulting filtered image $r(x, y)$.

However, multiplication in the DFT domain corresponds to circular convolution in the space domain

$$r[m, n] = (s \circledast_{M,N} h)[m, n] \quad \overset{\text{DFT}}{\longleftrightarrow} \quad R[k, l] = S[k, l]\ H[k, l]. \tag{25}$$

The circular convolution operation for images of size $M \times N$ is defined as

$$(s \circledast_{M,N} h)[m, n] = \sum_{m'=0}^{M-1} \sum_{n'=0}^{N-1} h[m', n']\ s[\langle m - m' \rangle_M, \langle n - n' \rangle_N], \tag{26}$$

where $\langle n \rangle_N$ denotes modulo N of n.