UNIVERSITY OF ILLINOIS
Department of Electrical and Computer Engineering
ECE 417 Multimedia Signal Processing

Lecture 11 Sample Problem Solutions

Problem 11.1

1. The standard way to implement 2D convolution is

$$f[n_1,n_2] = \sum_{m_1=\infty}^{\infty}\sum_{m_2=-\infty}^{\infty} x[m_1,m_2]h[n_1-m_1,n_2-m_2] \tag{11.1-1}$$

   Eq. 11.1-1 requires a double-summation, with up to M_1M_2 non-zero terms in the summation, in order to compute each output pixel. There are $(N_1+M_1-1)\times(N_2+M_2-1) = \mathcal{O}\left\{N_1N_2\right\}$ output pixels. So the total complexity is $\mathcal{O}\left\{N_1N_2M_1M_2\right\}$.

   More loosely, if $N_1 \approx N_2 \approx M_1 \approx M_2 \approx N$, then we can say that standard convolution has a complexity of $\mathcal{O}\left\{N^4\right\}$.

2. The integral image is

$$i[n_1,n_2] = \sum_{m_1=0}^{n_1}\sum_{m_2=0}^{n_2} x[m_1,m_2]$$

   It can be computed in three operations per output pixel:

$$i[n_1,n_2] = x[n_1,n_2] + i[n_1-1,n_2] + i[n_1,n_2-1] - i[n_1-1,n_2-1]$$

   The feature we are trying to compute is

$$\begin{aligned} f[n_1,n_2] \quad &= \textstyle\sum_{m_1=n_1-M_1/2+1}^{n_1}\sum_{m_2=n_2-M_2/2+1}^{n_2} x[n_1,n_2] \\ &- \textstyle\sum_{m_1=n_1-M_1/2+1}^{n_1}\sum_{m_2=n_2-M_2+1}^{n_2-M_2/2} x[n_1,n_2] \\ &- \textstyle\sum_{m_1=n_1-M_1+1}^{n_1-M_1/2}\sum_{m_2=n_2-M_2/2+1}^{n_2} x[n_1,n_2] \\ &- \textstyle\sum_{m_1=n_1-M_1+1}^{n_1-M_1/2}\sum_{m_2=n_2-M_2+1}^{n_2-M_2/2} x[n_1,n_2] \end{aligned}$$

   Which can be computed in just eight operations per output pixel, as

$$\begin{aligned} f[n_1,n_2] \quad &= i[n_1,n_2] - 2i\left[n_1-\tfrac{M_1}{2},n_2\right] + i\left[n_1-M_1,n_2\right] \\ &-2i\left[n_1,n_2-\tfrac{M_2}{2}\right] + 4i\left[n_1-\tfrac{M_1}{2},n_2-\tfrac{M_2}{2}\right] - 2i\left[n_1-M_1,n_2-\tfrac{M_2}{2}\right] \\ &+i\left[n_1,n_2-M_2\right] - 2i\left[n_1-\tfrac{M_1}{2},n_2-M_2\right] + i\left[n_1-M_1,n_2-M_2\right] \end{aligned}$$

   So the total complexity is just 8+3=11 additions per output pixel, for a total of $11N_1N_2 = \mathcal{O}\left\{N_1N_2\right\}$ additions.