Engineering Tripos Part IIB FOURTH YEAR

Module 4F12: Computer Vision

Solutions to Examples Paper 1

1. *Images*

   Each frame requires $1920 \times 1080 \times 1 = 2.07 \times 10^6$ Bytes. A 25Hz stereo image stream requires $2.07 \times 10^6 \times 25 \times 2 = 1.04 \times 10^8$ Bytes/s. Assuming an average A4 page of text contains 50 lines, with about 80 characters on each line, and that a character is represented (using an ASCII code) as a single byte, a page of text requires $80 \times 50 \times 1 = 4000$ Bytes. So, instead of one second of stereo video, we could alternatively store $1.04 \times 10^8/4000 \approx 26000$ pages of text — enough for a large encyclopaedia!

2. *Smoothing by convolution with a Gaussian*

   Consider smoothing an image, first with a Gaussian of standard deviation σ_1, then with a Gaussian of standard deviation σ_2:

$$s(x) = g_{\sigma 2}(x) * (g_{\sigma 1}(x) * I(x))$$

   Since convolution is associative, we can write this as the convolution of the image with the kernel $g_{\sigma 2}(x) * g_{\sigma 1}(x)$:

$$s(x) = (g_{\sigma 2}(x) * g_{\sigma 1}(x)) * I(x)$$

   The easiest way to evaluate the convolution of two Gaussians is to find their Fourier transforms and then multiply the transforms in the frequency domain. If $g_\sigma(x) \leftrightarrow G_\sigma(\omega)$, then:

$$\begin{aligned}
G_\sigma(\omega) &= \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} \exp\left(-\frac{x^2}{2\sigma^2}\right) e^{-j\omega x} dx \\
&= \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} \exp\left[-\left(\frac{x^2}{2\sigma^2} + j\omega x\right)\right] dx \\
&= \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} \exp\left[-\frac{1}{2\sigma^2}\left(x^2 + 2j\omega\sigma^2 x\right)\right] dx \\
&= \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} \exp\left[-\frac{1}{2\sigma^2}\left((x + j\omega\sigma^2)^2 - j^2\omega^2\sigma^4\right)\right] dx \\
&= \exp\left(-\frac{\omega^2\sigma^2}{2}\right) \times \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} \exp\left(-\frac{(x + j\omega\sigma^2)^2}{2\sigma^2}\right) dx \\
&= \exp\left(-\frac{\omega^2\sigma^2}{2}\right) \quad \text{(since the integral is a standard Gaussian)}
\end{aligned}$$

Hence

$$\begin{aligned}
g_{\sigma 2}(x) * g_{\sigma 1}(x) &\leftrightarrow G_{\sigma 2}(\omega) \times G_{\sigma 1}(\omega) = \exp\left(-\frac{\omega^2\sigma_2^2}{2}\right) \times \exp\left(-\frac{\omega^2\sigma_1^2}{2}\right) \\
\Leftrightarrow g_{\sigma 2}(x) * g_{\sigma 1}(x) &\leftrightarrow \exp\left(-\frac{\omega^2(\sigma_2^2+\sigma_1^2)}{2}\right)
\end{aligned}$$

The expression on the right is the Fourier transforms of a Gaussian with standard deviation $\sqrt{\sigma_2^2+\sigma_1^2}$. So the convolution of two Gaussians with variances σ_1^2 and σ_2^2 is a Gaussian with variance $\sigma_1^2+\sigma_2^2$. It follows that consecutive smoothing with a series of 1D Gaussians, each with a particular standard deviation σ_i, is equivalent to a single convolution with a Gaussian of variance $\sum_i \sigma_i^2$.

Spatial domain convolution

Alternatively, we can convolve in the spatial domain. The trick, once again, is to complete the square:

$$g_{\sigma 2}(x) * g_{\sigma 1}(x) = \frac{1}{2\pi\sigma_1\sigma_2}\int_{-\infty}^{\infty} \exp\left(-\frac{u^2}{2\sigma_2^2}\right)\exp\left(-\frac{(x-u)^2}{2\sigma_1^2}\right) du$$

$$\begin{aligned}
&= \frac{1}{2\pi\sigma_1\sigma_2}\int_{-\infty}^{\infty} \exp\left(\frac{-u^2\sigma_1^2 - x^2\sigma_2^2 - u^2\sigma_2^2 + 2ux\sigma_2^2}{2\sigma_1^2\sigma_2^2}\right) du \\
&= \frac{1}{2\pi\sigma_1\sigma_2}\int_{-\infty}^{\infty} \exp\left(\frac{-(\sigma_1^2+\sigma_2^2)\left(u - \frac{x\sigma_2^2}{\sigma_1^2+\sigma_2^2}\right)^2 + \frac{x^2\sigma_2^4}{\sigma_1^2+\sigma_2^2} - x^2\sigma_2^2}{2\sigma_1^2\sigma_2^2}\right) du \\
&= \frac{1}{2\pi\sigma_1\sigma_2}\int_{-\infty}^{\infty} \exp\left(\frac{-\left(u - \frac{x\sigma_2^2}{\sigma_1^2+\sigma_2^2}\right)^2}{\frac{2\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}}\right) \exp\left(\frac{-x^2\sigma_1^2\sigma_2^2}{2(\sigma_1^2+\sigma_2^2)\sigma_1^2\sigma_2^2}\right) du \\
&= \frac{1}{2\pi\sigma_1\sigma_2}\exp\left(\frac{-x^2}{2(\sigma_1^2+\sigma_2^2)}\right)\int_{-\infty}^{\infty} \exp\left(\frac{-\left(u - \frac{x\sigma_2^2}{\sigma_1^2+\sigma_2^2}\right)^2}{2\left(\frac{\sigma_1\sigma_2}{\sqrt{\sigma_1^2+\sigma_2^2}}\right)^2}\right) du \\
&= \frac{1}{\sqrt{2\pi}\sqrt{\sigma_1^2+\sigma_2^2}}\exp\left(\frac{-x^2}{2(\sigma_1^2+\sigma_2^2)}\right)\frac{1}{\sqrt{2\pi}\left(\frac{\sigma_1\sigma_2}{\sqrt{\sigma_1^2+\sigma_2^2}}\right)}\int_{-\infty}^{\infty} \exp\left(\frac{-\left(u - \frac{x\sigma_2^2}{\sigma_1^2+\sigma_2^2}\right)^2}{2\left(\frac{\sigma_1\sigma_2}{\sqrt{\sigma_1^2+\sigma_2^2}}\right)^2}\right) du \\
&= \frac{1}{\sqrt{2\pi}\sqrt{\sigma_1^2+\sigma_2^2}}\exp\left(\frac{-x^2}{2(\sigma_1^2+\sigma_2^2)}\right) \quad \text{(since the integral is a standard Gaussian)}
\end{aligned}$$

This expression is a Gaussian with standard deviation $\sqrt{\sigma_2^2+\sigma_1^2}$.

3. *Generating the Gaussian filter kernel*

   In general, if we discard the sample $(n+1)$ pixels from the center of the kernel, the size of the kernel will be $2n+1$ pixels. We can find n by solving:

$$
\begin{aligned}
\exp\left[-\frac{(n+1)^2}{2\sigma^2}\right] &< \frac{1}{1000} \\
\Leftrightarrow n &> 3.7\sigma - 1
\end{aligned}
$$

   So n must be the nearest integer to $3.7\sigma - 0.5$.

   (a) Applying this formula for $\sigma = 1$ gives $n = 3$ and a kernel size of $2n+1 = 7$ pixels. The filter coefficients can be found by sampling the 1D Gaussian $g_1(x)$ at the points $x = \{-3, -2, -1, 0, 1, 2, 3\}$. The sum of the coefficients is one, so regions of uniform intensity are unaffected by smoothing.

   (b) For $\sigma = 5$ we get $n = 18$ and a kernel size of 37 pixels.

   (c) The choice of σ depends on the *scale* at which the image is to be analysed. Modest smoothing (a Gaussian kernel with small σ) brings out edges at a fine scale. More smoothing (larger σ) identifies edges at larger scales, suppressing the finer detail. There is no right or wrong size for the kernel: it all depends on the scale we're interested in. Another factor is image noise: the smoothing suppresses noise. It may be difficult to detect fine scale edges, since a kernel large enough to suppress the noise may also suppress the fine detail. Finally, computation time may be an issue: large σ means a large kernel and computationally expensive convolutions.

4. *Discrete convolution*

   The image and filter kernels are discrete quantities and convolutions are performed as truncated summations:

$$
s(x) = \sum_{u=-n}^{n} g_\sigma(u) I(x-u)
$$

   Applying this to the pixel with intensity 118, which is the 11th pixel in the row, we obtain

$$
\begin{aligned}
s(x) &= \sum_{u=-3}^{3} g_\sigma(u) I(11-u) \\
&= 0.004 \times 57 + 0.054 \times 77 + 0.242 \times 99 + 0.399 \times 118 \ldots \\
&\quad + 0.242 \times 130 + 0.054 \times 133 + 0.004 \times 134 \\
&= 115 \qquad \text{(to the nearest integer)}
\end{aligned}
$$

5. *Derivative of convolution theorem*

(a) This is easily proved by interchanging the order of differentiation and integration:

$$\begin{aligned}
s'(x) &= \frac{d}{dx}\left[g_\sigma(x) * I(x)\right] = \frac{d}{dx}\left[\int_{-\infty}^{\infty} g(x-u)I(u)\,du\right] \\
&= \int_{-\infty}^{\infty} \frac{d}{dx}\left[g(x-u)\right] I(u)\,du \\
&= \int_{-\infty}^{\infty} g'(x-u)I(u)\,du = g'_\sigma(x) * I(x)
\end{aligned}$$

(b) Edges are localised at the maxima and minima of $\frac{d}{dx}[g_\sigma(x) * I(x)]$. These occur when

$$\frac{d^2}{dx^2}[g_\sigma(x) * I(x)] = 0$$

The derivate of convolution theorem tells us that

$$\frac{d^2}{dx^2}[g_\sigma(x) * I(x)] = g''_\sigma(x) * I(x)$$

Hence edges can be localised at the zero-crossings of $g''_\sigma(x) * I(x)$.

6. *Differentiation and 1D edge detection*

   An approximation to the first-order spatial derivative of $I(x)$ mid-way between the nth and $(n+1)$th sample is $I(n+1) - I(n)$. This can be computed by convolving with the kernel

1/2	0	-1/2

   (remember that the kernel is flipped before the multiply and accumulate operation).

   Applying this kernel to the smoothed row of pixels gives the approximation to the first-order spatial derivative:

x	x	x	x	2.5	3	5.5	11.5	17	18	14	8.5	3.5	0.5	-.05	x	x	x	x

   The intensity discontinuity is at the maximum of the first-order spatial derivative. The maximum derivative (18) occurs at the tenth pixel - between the pixel with smoothed intensity 79 and the pixel with intensity 98[1].

7. *Decomposition of 2D convolution*

   The 2D convolution can be decomposed into two 1D convolutions as follows:

$$G_\sigma(x,y) * I(x,y) = \frac{1}{2\pi\sigma^2}\int\int I(x-u, y-v)\ \exp - \left(\frac{u^2+v^2}{2\sigma^2}\right)\ du\ dv$$

[1] If you want to be more precise, you can localise the discontinuity to sub-pixel accuracy by calculating the second order derivatives and then interpolating to find the zero-crossing.

$$
\begin{aligned}
&= \frac{1}{\sqrt{2\pi}\sigma} \int \exp - \left(\frac{u^2}{2\sigma^2}\right) \left[\frac{1}{\sqrt{2\pi}\sigma} \int I(x-u, y-v) \exp - \left(\frac{v^2}{2\sigma^2}\right) dv \right] du \\
&= \frac{1}{\sqrt{2\pi}\sigma} \int \exp - \left(\frac{u^2}{2\sigma^2}\right) [g_\sigma(y) * I(x-u, y)] \, du \\
&= g_\sigma(x) * [g_\sigma(y) * I(x, y)]
\end{aligned}
$$

Performing two 1D convolutions is much more efficient and quicker than performing a single 2D convolution. A discrete 1D convolution with a kernel of size $N = 2n + 1$ requires N multiply and add operations. A discrete 2D convolution with a kernel of size $N \times N$ requires N^2 multiply and add operations. The speed-up offered by decomposing the 2D convolution is $N^2/2N = N/2$.

8. *Corner detection*

   This is taken from a previous tripos examination. See solutions for 4F12 examination Q1 in 2012 (or 2018).

9. *Band-pass filtering using Image Pyramids*

   This question was taken from 4F12 examination Q1 (2021). See the examination solution and marking scheme (Crib).

10. *Feature description and matching*

    This question was taken from a previous tripos examination. See solution and marking scheme for 4F12 examination Q1 (2021).

Roberto Cipolla
October 2022