Supplementary Material for: Loglet SIFT for Part Description in Deformable Part Models: Application to Face Alignment

Qiang Zhang
q.zhang.13@warwick.ac.uk
Abhir Bhalerao
abhir.bhalerao@warwick.ac.uk

Department of Computer Science
University of Warwick
Coventry, UK

Appendix A: Scale Pooling in Spatial Domain and Filter Accumulation in Fourier Domain

One of the key features of the SIFT is that it performs the pooling in a 2D neighbourhood in the spatial domain. In a recent study [1] Dong proposed to extend the pooling to spatial-scale space, by performing an additional pooling across adjacent scales in order to enhance the invariance to minor scale changes. We prove that the filter accumulation in Fourier domain is equivalent to scale pooling, under the approximation that gradients at adjacent scales have similar orientations, which is reasonable when low orientation resolution such as $\pi/4$ (8 orientation bins) are used.

An un-normalised gradient histogram of a SIFT cell in a region centred at point $\boldsymbol{x}$ can be written compactly as [1],

$$h(\theta|I) = \sum_{\boldsymbol{x}'} \kappa_\varepsilon(\theta - \angle\nabla I(\boldsymbol{x}'))\kappa_\sigma(\boldsymbol{x}-\boldsymbol{x}')||\nabla I(\boldsymbol{x}')|| \tag{1}$$

where θ is a variable corresponding to an orientation histogram bin. Discrete bins are computed using a bilinear interpolation kernel κ_ε with $\varepsilon = 2\pi/n$ where n is the number of bins, and linear spatial weighting kernel κ_σ with σ controls the size of a cell. Note that unlike in [1] we use a discrete form in (1) as is the case in the practical implementation.

Now consider the filter accumulation in Fourier domain used in the first scale of our L-SIFT descriptor. The gradient can be represented by $\nabla I = [I_x, I_y]$, with each direction calculated with the first scale filter (a bundle of loglets),

$$\begin{aligned} I_x =& \mathcal{F}^{-1}(\mathcal{F}(I)\cdot\mathcal{W}_x^{(1)}) \\ =& \mathcal{F}^{-1}\left(\mathcal{F}(I)\cdot\sum_s \mathcal{W}_x(\boldsymbol{u},s)\right) \\ =& \sum_s \mathcal{F}^{-1}\left(\mathcal{F}(I)\cdot\mathcal{W}_x(\boldsymbol{u},s)\right), \quad s\in\{0,-1,...\} \end{aligned} \tag{2}$$

The gradient can therefore be written as

$$\nabla I = [I_x, I_y] = \sum_s \nabla^{(s)} I \tag{3}$$

with $\nabla^{(s)}$ represents a gradient computation with a single loglet filter at scale s. Substituting (3) into (1) we obtain the SIFT with gradient computed by accumulated filters,

$$h(\theta|I) = \sum_{\boldsymbol{x}'} \kappa_\varepsilon(\theta - \angle\nabla I(\boldsymbol{x}'))\kappa_\sigma(\boldsymbol{x} - \boldsymbol{x}')||\sum_s \nabla^{(s)} I(\boldsymbol{x}')|| \tag{4}$$

Considering that feature gradients at adjacent scales have similar orientations, and SIFT uses discrete orientation bins with a low angular resolution, the following approximation can be made,

$$\begin{aligned} h(\theta|I) &\approx \sum_s \sum_{\boldsymbol{x}'} \kappa_\varepsilon(\theta - \angle\nabla_s I(\boldsymbol{x}'))\kappa_\sigma(\boldsymbol{x} - \boldsymbol{x}')||\nabla_s I(\boldsymbol{x}')|| \\ &= \sum_s h_s(\theta|I) \end{aligned} \tag{5}$$

Where $h_s(\theta|I)$ represents a standard SIFT with the gradient computed by a single loglet at scale s. Therefore we proved that *a SIFT computed on the gradient by accumulated filters in Fourier domain is equivalent to accumulating a group of SIFTs cumputed on multi-scale gradients in spatial domain.*

The rationale behind the significant improvement by scale pooling is that it gives invariance to minor scale changes as well as sample shifts of digital images. In our strategy, the first invariance is achieved by expanding the bandwidth by filter accumulation, the second invariance comes from the natural insensitivity of loglets function to sample shift [2].

Appendix B: Spectrum Cropping as Image Downsampling

A digital image is a discrete (i.e., band limited) sampling of the continuous (i.e., not band limited) *true* signal. In Fourier domain, the spectrum of the image therefore covers only the low frequency components of the true signal spectrum which spreads infinitely, and cuts off at the Nyquist boundary, see Fig. 1. Images representing the same scene with lower resolution is presented in Fourier domain as a spectrum covering a smaller range centred at the zero frequency. As such, image downsampling can be done by cutting the spectrum keeping only the lower frequency. In practice, due to the discrete form, both the image and the spectrum are periodic signals. To avoid the aliasing caused by the periodic discontinuity, a window function such as Gaussian is applied to attenuate the magnitude near Nyquist frequency, which appears in spatial domain as a Gaussian smoothing. Below we describe the process mathematically.

Denote $\mathcal{I}$ as the spectrum of a digital image I, the image can be recovered by inverse Fourier transform,

$$I(\boldsymbol{x}) = \iint_{-\pi}^{\pi} \mathcal{I}(\boldsymbol{u}) e^{j\boldsymbol{x}\cdot\boldsymbol{u}} \mathrm{d}\boldsymbol{u} \tag{6}$$

where $\boldsymbol{x} = (x, y)$ and $\boldsymbol{u} = (u, v)$ are the index vectors in spatial and Fourier domain respectively. Suppose we want to downsample the image at ratio $\beta \in (0, 1)$. We first apply a

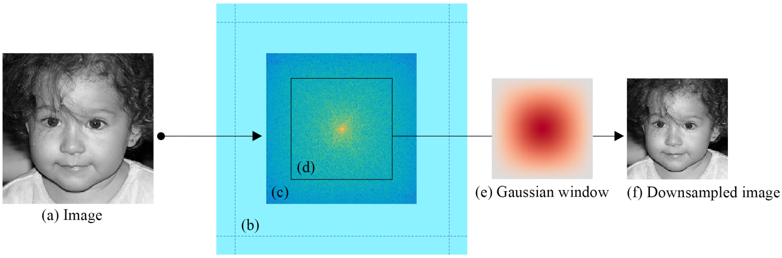

Figure 1: (a) A digital image is a discrete sampling and approximation of the true continuous signal at certain resolution. (b) Spectrum of the true continuous signal is not band-limited therefore spreads infinitely. (c) Spectrum of the digital image is band-limited therefore covers only a low frequency area and is truncated at certain range. (d) Image downsampling can be performed directly by cropping the spectrum in Fourier domain. In theory the new spectrum (d) should be a subregion of the true spectrum (b), which is however not available. Therefore the cropping can be performed on the spectrum (c) instead. In practice cropping the spectrum can not ensure the periodic contineous of the new spectrum therefore causes aliasing effect. A standard way is to attenuate the spectrum to zero at the boundary by applying a Gaussian window to it (e), which corresponds to a smoothing in the spatial domain.

window function to attenuate the components beyond the boundary $\pm\beta\pi$ to zero. With the windowing (6) equals to,

$$I(\boldsymbol{x}) = \iint_{-\beta\pi}^{\beta\pi} \mathcal{I}(\boldsymbol{u}) e^{j\boldsymbol{x}\cdot\boldsymbol{u}} \mathrm{d}\boldsymbol{u} \tag{7}$$

We define new variables $\boldsymbol{u}_1 = \boldsymbol{u}/\beta$, $\boldsymbol{x}_1 = \beta\boldsymbol{x}$, and a coordinate transform of the spectrum $\mathcal{I}_1(\boldsymbol{u}_1) = \mathcal{I}(\boldsymbol{u})$. Substituting them in to (7) we have,

$$\begin{aligned} I(\boldsymbol{x}) &= \iint_{-\beta\pi}^{\beta\pi} \mathcal{I}_1(\boldsymbol{u}_1) e^{j\boldsymbol{x}_1\cdot\boldsymbol{u}_1} \mathrm{d}(\beta\boldsymbol{u}_1) \\ &= \beta \iint_{-\pi}^{\pi} \mathcal{I}_1(\boldsymbol{u}_1) e^{j\boldsymbol{x}_1\cdot\boldsymbol{u}_1} \mathrm{d}\boldsymbol{u}_1 \\ &= \beta I_1(\boldsymbol{x}_1), \end{aligned} \tag{8}$$

where I_1 is the downsampled image, i.e.,

$$I_1(\boldsymbol{x}_1) = I(\boldsymbol{x})/\beta, \quad \boldsymbol{x}_1 = \beta\boldsymbol{x}. \tag{9}$$

In our case, with larger scale filters $\mathcal{W}^{(s)}, s \in \{2, 3, \ldots\}$, the high frequency components beyond the boundary $\pm\pi/2^{(s-1)}$ are eliminated and a direct crop of the spectrum gives an efficient downsampling at ratio $1/2^{(s-1)}$ without aliasing.

Appendix C

Theorem: Derivative of an image $I(x, y)$ results in an imaginary anti-symmetrical transform of the spectrum $\mathcal{F}(I)$.

Derivation: Taking the derivative with respect to x as an example. The Fourier transform of I is,

$$\mathcal{F}(I) = \iint_{-\infty}^{+\infty} I\, e^{-2\pi i(ux+vy)} dxdy, \tag{10}$$

where $(u,v) \in [-\pi, \pi]$ are the coordinates in the Fourier domain. The Fourier transform of the derivative $\mathrm{d}I/\mathrm{d}x$ is,

$$\mathcal{F}(\frac{\mathrm{d}I}{\mathrm{d}x}) = \iint_{-\infty}^{+\infty} \frac{\mathrm{d}I}{\mathrm{d}x} e^{-2\pi i(ux+vy)} dxdy, \tag{11}$$

Consider the following equation,

$$\begin{aligned} \frac{\mathrm{d}(Ie^{-2\pi i(ux+vy)})}{\mathrm{d}x} &= \frac{\mathrm{d}I}{\mathrm{d}x} e^{-2\pi i(ux+vy)} + \frac{\mathrm{d}(e^{-2\pi i(ux+vy)})}{\mathrm{d}x} I \\ &= \frac{\mathrm{d}I}{\mathrm{d}x} e^{-2\pi i(ux+vy)} - 2\pi iu\, e^{-2\pi i(ux+vy)} I, \end{aligned} \tag{12}$$

therefore,

$$\frac{\mathrm{d}I}{\mathrm{d}x} e^{-2\pi i(ux+vy)} = \frac{\mathrm{d}(Ie^{-2\pi i(ux+vy)})}{\mathrm{d}x} + 2\pi iu\, e^{-2\pi i(ux+vy)} I, \tag{13}$$

Substituting (13) into (11) we have,

$$\begin{aligned} \mathcal{F}(\frac{\mathrm{d}I}{\mathrm{d}x}) &= \iint_{-\infty}^{+\infty} \frac{\mathrm{d}(Ie^{-2\pi i(ux+vy)})}{\mathrm{d}x} dxdy + \iint_{-\infty}^{+\infty} 2\pi iu\, e^{-2\pi i(ux+vy)} I dxdy \\ &= 2\pi iu \iint_{-\infty}^{+\infty} I\, e^{-2\pi i(ux+vy)} dxdy \\ &= 2\pi iu\, \mathcal{F}(I), \end{aligned} \tag{14}$$

which indicates that the derivative of image I results in the multiplication of the Fourier spectrum by an imaginary anti-symmetrical term $2\pi iu$.

References

[1] Jingming Dong and Stefano Soatto. Domain-size pooling in local descriptors: DSP-SIFT. In *Computer Vision and Pattern Recognition, 2005. IEEE Conference on*, 2015.

[2] Hans Knutsson and Mats Andersson. Loglets: Generalized quadrature and phase for local spatio-temporal structure estimation. In *Image Analysis*, pages 741–748. Springer, 2003.