Approximate Two-Dimensional Separable Digital Filtering

Konstantinos G. Derpanis
Department of Computer Science and Engineering
York University
kosta@cs.yorku.ca

March 5, 2005

A two-dimensional linear digital filter may be characterized in terms of its impulse response which takes the form of a matrix. Unfortunately, as the dimensions of the filter increase its computational efficiency in convolutional implementations rapidly diminish. In this note, we consider the *Singular Value Decomposition* (SVD) approach for expanding an arbitrary 2D filter into a finite and converging sum of separable filtering stages (Treitel & Shanks, 1971). Computational savings are realized by truncating this representation with the consequence of introducing a *tolerable* error. For a brief review of the SVD, see Appendix A.

A two-dimensional $m \times n$ filter, $\mathbf{F}$, is said to be separable if it can be expressed as the product of a column and a row vector:

$$\mathbf{F} = \mathbf{ab} \tag{1}$$

$$= \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix} \begin{bmatrix} b_1 & b_2 & \cdots & b_m \end{bmatrix} \tag{2}$$

$$= \begin{bmatrix} a_1b_1 & a_1b_2 & \cdots & a_1b_n \\ a_2b_1 & a_2b_2 & \cdots & a_2b_n \\ \vdots & & & \\ a_mb_1 & a_mb_2 & \cdots & a_mb_n \end{bmatrix}. \tag{3}$$

Given this filter structure, it can be easily shown that the two-dimensional convolution can be reexpressed as a sequence of one-dimensional filtering operations: 1D convolution along the rows (columns) followed by 1D convolution along the columns (rows).

The SVD of the 2D filter, $\mathbf{F}[x, y]$, results in the finite and converging sum of separable convolutions,

$$\mathbf{F}[x, y] = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top \tag{4}$$

$$= \sum_{i=1}^{k} \sigma_i g_i(x) h_i(y), \tag{5}$$

where $g_i(\cdot)$ and $h_i(\cdot)$ denote the columns of $\mathbf{U}$ and $\mathbf{V}$, respectively, σ_i denotes the ith element along the diagonal of $\mathbf{\Sigma}$ and $k = \min\{m, n\}$. This representation of the original filter is *exact* and no direct efficiencies[1] are gained by applying this separable multistage formulation. The computational efficiencies are realized by truncating the representation to the l dominant terms where $l < k$ (i.e., setting $\sigma_{l+1} = \sigma_{l+2} = \cdots = \sigma_k = 0$). It can be easily shown that retaining the l greatest singular values σ_i yields the optimal reconstruction of $\mathbf{F}$ in the least-squares sense.

References

Golub, G. & Van Loan, C. (1989). *Matrix Computations* (second Ed.). Baltimore, MD, USA: Johns Hopkins Press.

Treitel, S. & Shanks, J. (1971). The design of multistage separable planar filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *9(1)*, 10–27.

[1] Efficiencies in the exact representation may be realized by virtue of the structure of the exact sum of separable formulation being more amenable to hardware implementation.

A Singular Value Decomposition

The *Singular Value Decomposition* (SVD) (Golub & Van Loan, 1989) of an arbitrary $m \times n$ real matrix $\mathbf{F}$ is given by,

$$\mathbf{F} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top, \tag{6}$$

where $\mathbf{U}$ and $\mathbf{V}$ are orthonormal matrices of dimensions $m \times m$ and $n \times n$, respectively, and $\mathbf{\Sigma}$ is a diagonal matrix of dimensions $m \times n$. $\mathbf{\Sigma}$ is termed the singular value matrix while the elements along the diagonal are termed the singular values. The columns of $\mathbf{U}$ and $\mathbf{V}$ correspond to the eigenvectors of $\mathbf{F}\mathbf{F}^\top$ and $\mathbf{F}^\top\mathbf{F}$, respectively, whereas the singular values correspond to the square roots of the eigenvalues from $\mathbf{F}\mathbf{F}^\top$ or $\mathbf{F}^\top\mathbf{F}$.

Equivalently, the SVD decomposition can be expressed as:

$$\mathbf{F}[x, y] = \sum_{i=1}^{k} \sigma_i g_i(x) h_i(y) \tag{7}$$

where $g_i(\cdot)$ and $h_i(\cdot)$ denote the columns of $\mathbf{U}$ and $\mathbf{V}$, respectively, σ_i denotes the ith element along the diagonal of $\mathbf{\Sigma}$ and $k = \min\{m, n\}$.