IMAGE RESAMPLING BETWEEN ORTHOGONAL AND HEXAGONAL LATTICES

Dimitri Van De Ville[*,1], *Rik Van de Walle*[1], *Wilfried Philips*[2], *Ignace Lemahieu*[1]

Ghent University – [1]ELIS, [2]TELIN
Sint-Pietersnieuwstraat 41
B-9000 Ghent, Belgium

ABSTRACT

Resampling techniques are commonly required in digital image processing systems. Many times the classical interpolation functions are used, i.e., nearest-neighbour interpolation and bilinear interpolation, which are prone to the introduction of undesirable artifacts due to aliasing such as moire patterns. This paper presents a novel approach which minimizes the loss of information, in a least-squares sense, while resampling between orthogonal and hexagonal lattices. Making use of an extension of 2D splines to hexagonal lattices, the proper reconstruction function is derived. Experimental results for a printing application demonstrate the feasibility of the proposed method and are compared against the classical techniques.

1. INTRODUCTION

Digital images are sampled on a regular lattice. The conversion of this representation from one lattice to another is called image resampling. This operation is indispensable for many applications such as printing. We propose a new method to resample between orthogonal and hexagonal lattices.

The standard procedure for linear resampling consists of two conceptual steps: first, an image in the continuous domain is reconstructed; second, this function is resampled on the target lattice [1, 2, 3, 4]. Shannon's sampling theorem assumes that images are band-limited, and proposes to choose the interpolation filter to the ideal low-pass filter. However, real-world signals are not band-limited and both the image and the interpolation function have a finite support. Due to the slow decay of the ideal interpolation functions (which are sinc-like), it is also quite difficult to approximate them on a finite support. Additionally, ideal interpolators tend to generate the Gibb's phenomenon, which becomes visually apparent in images as ringing along the edges.

Instead of holding on to the band-limited hypothesis, many authors, such as Unser [5, 6, 7, 8], set up a family of basis functions based on splines. These splines have a limited size of support, which expands as the order increases. Ultimately, a spline representation of infinite order approaches the ideal filter. For example, first-order spline interpolation is better known as "nearest neighbour" interpolation; second-order spline interpolation as bilinear interpolation. Higher orders, such as bicubic spline interpolation yield even smoother results.

The standard approach does not minimize the information loss. Annoying artifacts due to aliasing (such as moire patterns) might arise. Unser et al. [5, 9] derived an algorithm based on the principle of convolution-based least-squares spline approximation. In particular, the samples on the target lattice are chosen such that the mean squared error between the spline representation on the source lattice and a similar one on the target lattice is minimized. This theory was developed for a 1D spline representation, and extended to 2D orthogonal lattices by means of tensor-product splines (i.e., the 2D spline is the product of two 1D splines).

This paper discusses the case of resampling between orthogonal and hexagonal lattices, therefore requiring a spline definition suitable for hexagonal lattices. We propose a simple recipe to construct hexagonal splines and make use of these splines to derive the least-squares reconstruction function. To demonstrate the feasibility of the proposed approach, we implemented our method for the practical case of gravure printing, a printing technique which is very susceptible to aliasing artifacts when using classical resampling procedures.

2. TWO-DIMENSIONAL SPLINES

We will denote a 2D function in the continuous domain as $g(\mathbf{x})$, where $\mathbf{x} \in \mathbb{R}^2$. The L_2-norm of $g(\mathbf{x})$ is derived from the inner product: $||g|| = \langle g, g \rangle^{1/2}$. Analogously, we denote a discrete 2D array as $c(\mathbf{k})$, where $\mathbf{k} \in \mathbb{Z}^2$. We denote the l_2-norm of $c(\mathbf{k})$ as $||c||_{l_2} = \left(\sum_{\mathbf{k} \in \mathbb{Z}^2} c(\mathbf{k}) c^*(\mathbf{k})\right)^{1/2}$.

A 2D lattice can be characterized by a matrix $\mathbf{R} = [\mathbf{r}_1 | \mathbf{r}_2]$ constituted of two linearly independent vectors $\mathbf{r}_1$ and $\mathbf{r}_2$ [10]. It is convenient to define an array of impulses

[*]Dimitri Van De Ville is a Research Assistant with the FWO Flanders, Belgium

on the lattice sites:

$$\delta_{\mathbf{R}}(\mathbf{x}) = \sum_{\mathbf{k}\in\mathbb{Z}^2} \delta(\mathbf{x} - \mathbf{R}\mathbf{k}), \tag{1}$$

where $\delta(\mathbf{x})$ represents a Dirac function.

Related to a lattice is a Voronoi cell, which is defined as the set of all points that are closer to the origin $\mathbf{0}$ than to any other site of the lattice. The Voronoi cell is represented by its indicator function $\chi_{\mathbf{R}}(\mathbf{x})$:

$$\chi_{\mathbf{R}}(\mathbf{x}) = \begin{cases} 1, & \mathbf{x} \in \text{ Voronoi cell,} \\ 1/m, & \mathbf{x} \text{ on edge Voronoi cell,} \\ 0, & \mathbf{x} \notin \text{ Voronoi cell,} \end{cases} \tag{2}$$

where m equals the number of lattice sites to which $\mathbf{x}$ is equidistant. Note that this function, when periodically copied onto all the lattice sites, covers the complete plane: $\delta_{\mathbf{R}} \star \chi_{\mathbf{R}}(\mathbf{x}) = 1$, where the $\star$-operator denotes the 2D continuous convolution. It is said that the Voronoi cell tiles the plane.

Consider a function $g(\mathbf{x})$, which is sampled on each lattice site of a lattice $\mathbf{R}$. Using a shift-invariant 2D generating function $\phi(\mathbf{x})$, we define the approximation space $S(\phi)$ as follows [11]

$$S(\phi) = \left\{ s(\mathbf{x}) \;\middle|\; s(\mathbf{x}) = \sum_{\mathbf{k}\in\mathbb{Z}^2} c(\mathbf{k})\phi(\mathbf{x} - \mathbf{R}\mathbf{k}) \right\}, \tag{3}$$

where the coefficients $c(\mathbf{k})$ need to be chosen such that $s(\mathbf{R}\mathbf{k}) = g(\mathbf{R}\mathbf{k})$. As such, any function $s(\mathbf{x}) \in S(\phi)$ is characterized by a sequence of coefficients $c(\mathbf{k})$. Notice that these coefficients are not necessarily samples $s(\mathbf{R}\mathbf{k})$ at the lattice points. Several conditions are required to obtain a sensible continuous/discrete model [11], among which most importantly to form a Riesz basis.

Splines suitable for a regular orthogonal lattice, described by the unity matrix, can easily be obtained by using the tensor-product of two one-dimensional B-splines $\beta^n(\mathbf{x}) = \beta^n(x_1)\beta^n(x_2)$. The superscript n refers to the n-th degree of piecewise polynomials or to the $n+1$-th order of approximation [12].

Consider now a regular hexagonal lattice, described by

$$\tilde{\mathbf{R}} = \begin{bmatrix} \sqrt{3}/2 & 0 \\ -1/2 & 1 \end{bmatrix}. \tag{4}$$

We define the surface area of the Voronoi cell as $\Omega = \left|\det(\tilde{\mathbf{R}})\right| = \sqrt{3}/2$. Matrices and functions related to the hexagonal lattice are denoted by the ˜-notation.

To construct a spline basis on the hexagonal lattice, we are especially interested in preserving the convolution property because it plays an important role in the derivation of the least-squares approximation later on. Therefore, we first define the first-order hexagonal spline as the indicator function of the Voronoi cell: $\tilde{\beta}^0(\mathbf{x}) = \chi_{\tilde{\mathbf{R}}}(\mathbf{x})$. Note that this spline is normalized to the surface area of the basic cell: $\int \tilde{\beta}^0 d\mathbf{x} = \Omega$. By convolving this function with itself repeatedly, we construct hexagonal splines of higher orders:

$$\tilde{\beta}^n(\mathbf{x}) = \frac{\tilde{\beta}^0 \star \tilde{\beta}^{n-1}(\mathbf{x})}{\Omega}, \quad n \geq 1. \tag{5}$$

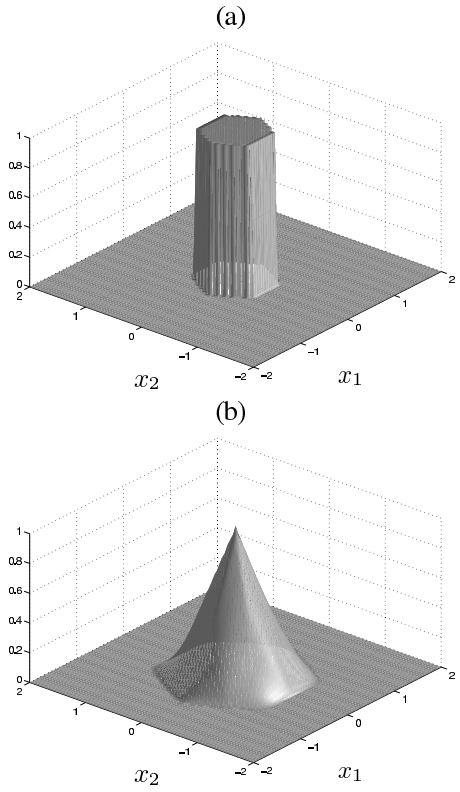

Fig. 1. Splines derived on a hexagonal lattice. (a) First order, (b) Second order.

Figure 1 shows the hexagonal splines of first and second order. The successive convolutions imply that the splines become smoother as the order increases. Interesting properties of the hexagonal splines (analytical form, Riesz basis, convexity, partition of unity, computation of the spline transform) can be found in [13].

3. LEAST-SQUARES RESAMPLING

Consider now two periodic lattices, i.e., an orthogonal source lattice $\mathbf{R}$ and a hexagonal target lattice $\tilde{\mathbf{R}}$. The two-dimensional splines of the previous section allow us to easily reconstruct an image in the continuous domain, using a reconstruction of the approximation spaces $S(\beta^n)$ and

$S(\tilde{\beta}^n)$. We are aiming at a convolution-based scheme

$$\tilde{g}^n(\mathbf{Rk}) = s(\mathbf{Rk}); \quad s(\mathbf{x}) = \sum_{\mathbf{k}} \Phi^n(\mathbf{x}-\mathbf{Rk}) g(\mathbf{Rk}), \quad (6)$$

where the reconstruction function $\Phi^n(\mathbf{x})$ realizes the least-squares approximation between both spline reconstructions.

The minimum L_2-norm approximation of a function $g(\mathbf{x})$ can be found by projection on $S(\tilde{\beta}^n)$. As such, the error $g(\mathbf{x}) - \tilde{g}^n(\mathbf{x})$ is orthogonal to $S(\tilde{\beta}^n)$. Since the original function $g(\mathbf{x})$ is only known at the lattice sites $\mathbf{Rk}$, we replace $g(\mathbf{x})$ by its approximation in $S(\beta^n)$, using a spline model for the source lattice. This enables us to write

$$\langle g^n(\mathbf{x}) - \tilde{g}^n(\mathbf{x}), \tilde{\beta}^n(\mathbf{x} - \tilde{\mathbf{R}}\mathbf{k}) \rangle = 0, \quad (7)$$

where $g^n(\mathbf{x})$ and $\tilde{g}^n(\mathbf{x})$ are the spline representations respectively on the orthogonal and hexagonal lattice. We can rewrite the expression as

$$\begin{aligned} &\langle g^n(\mathbf{x}), \tilde{\beta}^n(\mathbf{x} - \tilde{\mathbf{R}}\mathbf{k}) \rangle \\ &= \langle \sum_{\mathbf{k}} \tilde{c}(\mathbf{k}) \tilde{\beta}^n(\mathbf{x} - \tilde{\mathbf{R}}\mathbf{k}), \tilde{\beta}^n(\mathbf{x} - \tilde{\mathbf{R}}\mathbf{k}) \rangle \\ &= \sum_{\mathbf{k}} \tilde{c}(\mathbf{k}) \langle \tilde{\beta}^n(\mathbf{x} - \tilde{\mathbf{R}}\mathbf{k}), \tilde{\beta}^n(\mathbf{x} - \tilde{\mathbf{R}}\mathbf{k}) \rangle, \end{aligned} \quad (8)$$

where $\tilde{c}(\mathbf{k})$ are the spline coefficients on the hexagonal lattice. We now use the founding property of the hexagonal splines $\tilde{\beta}^n \star \tilde{\beta}^n(\mathbf{x})/\Omega = \tilde{\beta}^{2n+1}(\mathbf{x})$:

$$g^n \star \tilde{\beta}^n(\mathbf{x}) = \Omega \left(\sum_{\mathbf{k}} \delta(\mathbf{x} - \tilde{\mathbf{R}}\mathbf{k}) \tilde{c}(\mathbf{k}) \right) \star \tilde{\beta}^{2n+1}(\mathbf{x}). \quad (9)$$

The solution of Eq. (8) can be written as

$$\tilde{c}(\mathbf{k}) = \frac{g^n \star \tilde{\beta}^n \star (\tilde{b}^{2n+1})^{-1}}{\Omega}(\mathbf{Rk}), \quad (10)$$

where $b^n(\mathbf{x}) = \beta^n(\mathbf{x}) \delta_{\mathbf{R}}(\mathbf{x})$ is the "sampled spline". This enables us to write the least-squares interpolation function for resampling from the lattice $\mathbf{R}$ to $\tilde{\mathbf{R}}$ as

$$\Phi^n(\mathbf{x}) = \underbrace{(b^n)^{-1} \star \beta^n}_{1} \star \underbrace{\tilde{\beta}^n \star (\tilde{b}^{2n+1})^{-1}}_{2} \star \underbrace{\tilde{b}^n}_{3}(\mathbf{x})/\Omega, \quad (11)$$

where the underbraced expressions indicate:

1. the direct spline transform to compute the spline coefficients on the source lattice;
2. the least-squares approximation filter;
3. the final convolution to reconstruct the function in $S(\tilde{\beta}^n)$ using the new spline coefficients.

The computation of Eq. (11) requires the solution of two direct spline transforms (the inverse filters). One solution is to invert the matrix corresponding to the set of linear equations induced by the forward filters [7]. Another approach could be to implement the inversion scheme by means of recursive filters similar to the approach proposed by Unser [14, 6]. However, the factorization of the filter corresponding to the hexagonal splines is not trivial. Therefore, we propose to numerically approximate the reconstruction function on a limited support by means of a well-known iterative procedure [15].

In the case of $n = 0$, the reconstruction function of Eq. (11) becomes $\Phi^0(\mathbf{x}) = \beta^0 \star \tilde{\beta}^0(\mathbf{x})/\Omega$. No inverse filters are needed and the support is limited. This case is sometimes referred to as "surface projection": neighbouring samples of the source lattice are weighted by the relative overlap of source's and target's cell.

For the second-order least-squares approximation, the reconstruction function is given by

$$\Phi^1(\mathbf{x}) = \beta^1 \star (\tilde{b}^3)^{-1} \star \tilde{\beta}^1(\mathbf{x})/\Omega. \quad (12)$$

The presence of the inverse filter $(\tilde{b}^3)^{-1}$ implicates that the theoretical support of $\Phi^1(\mathbf{x})$ is the whole plane [13]. However, the fast decay shows that an approximation on a limited support is appropriate.

Note that this approach can easily be applied to resampling from hexagonal to orthogonal lattices. In that case, the inverse filters can be computed by recursive filters given in [14].

4. RESULTS

We implemented the proposed resampling method for a gravure printing application. The images are resampled to a hexagonal lattice before halftoning. Figure 2 shows the results of a test image after resampling using classical interpolation functions in (a)-(b), and after least-squares resampling in (c)-(d). The corresponding reconstruction functions are depicted in (e)-(f). Kindly note that the moire patterns are better suppressed by the least-squares approach and the second order least-squares produces sharper results.

5. CONCLUSION

In this paper, we presented a new method to resample between orthogonal and hexagonal lattices. After proposing a simple recipe to derive a two-dimensional spline basis for hexagonal lattices, we applied the principle of a least-squares approximation to derive a suitable reconstruction function. Many applications involving hexagonal lattices can make use of this approach.

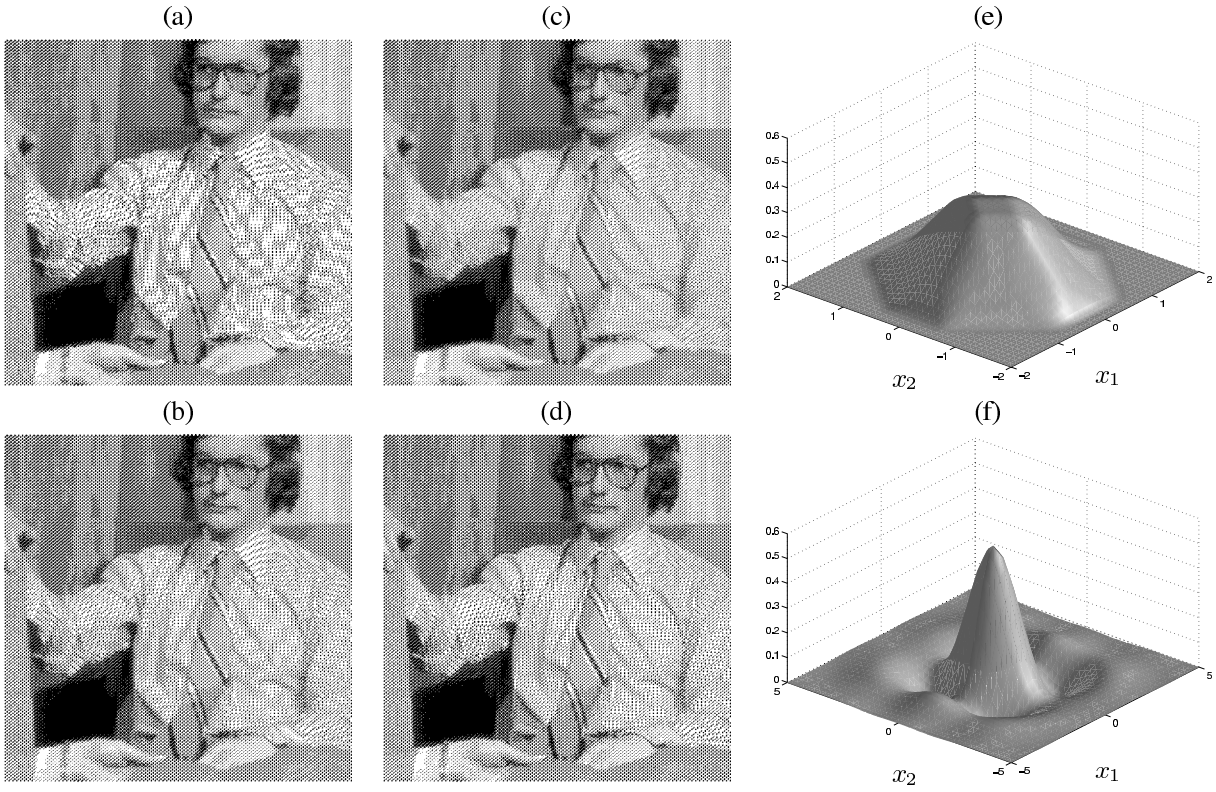

Fig. 2. Results after resampling the test image "shirt" to the gravure lattice. (a) Interpolative resampling (first order), (b) Interpolative resampling (second order), (c) Least-squares resampling (first order), (d) (Least-squares resampling (second order). (e) Reconstruction function for "surface projection" $\Phi^0(\mathbf{x})$. (f) Reconstruction function $\Phi^1(\mathbf{x})$.

6. REFERENCES

[1] A. Rosenfeld and A.C. Kak, *Digital Picture Processing*, Academic Press Inc., 2 edition, 1982.

[2] D.P. Mitchell and A.N. Netravali, "Reconstruction filters in computer graphics," *ACM Computer Graphics*, vol. 22, no. 4, pp. 221–228, Aug. 1988.

[3] J.C. Russ, *The image processing handbook*, CRC Press Inc., 1992.

[4] R. Soininen, "Digital Image Resizing Methods," M.S. thesis, Tampere University of Technology, 1994.

[5] M. Unser, A. Aldroubi, and M. Eden, "Enlargement or reduction of digital images with minimum loss of information," *IEEE Transactions on Image Processing*, vol. 4, no. 3, pp. 247–258, Mar. 1995.

[6] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing," *IEEE Transactions on Signal Processing*, vol. 41, no. 2, pp. 821–848, Feb. 1993.

[7] C. de Boor, *A Practical Guide to Splines*, vol. 27 of *Applied Mathematical Sciences*, Springer-Verlag, 1978.

[8] H.S. Hou and H.C. Andrews, "Cubic splines for image interpolation and digital filtering," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 26, no. 6, pp. 508–517, Dec. 1978.

[9] M. Unser and I. Daubechies, "On the approximation power of convolution-based least squares versus interpolation," *IEEE Transactions on Signal Processing*, vol. 45, no. 7, pp. 1697–1711, July 1997.

[10] R.A. Ulichney, *Digital Halftoning*, MIT Press, 1987.

[11] M. Unser, "Sampling—50 years after Shannon," *Proceedings of the IEEE*, vol. 88, no. 4, pp. 569–587, Apr. 2000.

[12] P. Thévenaz, T. Blu, and M. Unser, "Interpolation revisited," *IEEE Transactions on Medical Imaging*, vol. 19, no. 7, pp. 739–758, July 2000.

[13] D. Van De Ville, W. Philips, and I. Lemahieu, "Least-squares spline resampling to a hexagonal lattice," *Signal Processing: Image Communication*, accepted for publication.

[14] M. Unser, "Fast B-spline transforms for continuous image representation and interpolation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 3, pp. 277–285, Mar. 1991.

[15] R.W. Schafer, R.M. Mersereau, and M.A. Richards, "Constrained iterative restoration algorithms," *Proceedings of the IEEE*, vol. 69, no. 4, pp. 432–450, Apr. 1981.