the wavelet-based method computes a measure of local regularity and preserves this local regularity in the interpolated image. This regularity-based interpolation results in both perceptual and quantifiable improvement in the interpolated image. The algorithm provides the most improvement over traditional methods on images with strong, well defined edges that separate smooth, high-regularity regions.

REFERENCES


Two- and Three-Dimensional Image Rotation Using the FFT

Robert W. Cox and Raolqiong Tong

Abstract—Techniques for rotating two- and three-dimensional (2-D and 3-D) images using fast Fourier transforms (FFT’s) are presented. The methods are applications of the multidimensional chirp algorithm. In the 2-D case, one chirp transformation is sufficient, requiring four 2-D FFT’s. In the 3-D case, two successive chirp transformations are required, needing six 3-D FFT’s.

I. INTRODUCTION

Rotating an image to a new pixel grid is a basic operation required in many applications. Translation without rotation can be accomplished using the Fourier transform shift property

\[ f(x + a) = \mathcal{F}^{-1}[\mathcal{F}[f] \exp\{2\pi i a \cdot \mathbf{k}\}] \]

(1)

where \( \mathcal{F}[f] = \int_{\mathbb{R}^n} f(x) \exp\{-2\pi i (k \cdot x)\} d^n x \) is the continuous Fourier transform; \( \mathbb{R}^n \) denotes \( n \)-dimensional real Euclidean space with inner product \( \langle x, y \rangle = x^t y \). The discrete application of this requires a fast Fourier transform (FFT), scaling by the \( \exp\{2\pi i a \cdot \mathbf{k}\} \) factor, and an inverse FFT. The effect is trigonometric interpolation of the original image onto the translated grid. The purpose of this correspondence is to show how FFT’s can be used to interpolate two-dimensional (2-D) and three-dimensional (3-D) images to a new pixel grid that is both translated and rotated from the original grid.

II. TWO-DIMENSIONAL IMAGE ROTATION

Given the image \( f(p, q) \) on an \( N \times N \) square grid, we want to compute \( f \) on a grid rotated by an arbitrary angle \( \theta \) and shifted by an arbitrary translation \([a \ b]^t\). The first step is to compute the 2-D FFT of \( f \), so that

\[ \hat{f}(l, m) = \frac{1}{N^2} \sum_{p,q} f(p, q) e^{-2\pi i \frac{(p+a)+m+b)}{N}} \]

\[ f(p, q) = \frac{1}{N^2} \sum_{l,m} \hat{f}(l, m) e^{2\pi i \frac{(p+a)+m+b)}{N}}. \]

(2)

We evaluate (2) on the desired output grid

\[ f(p \cos \theta - q \sin \theta + a, p \sin \theta + q \cos \theta + b) \]

\[ = \frac{1}{N^2} \sum_{l,m} \hat{f}(l, m) e^{2\pi i \frac{(l+a+b)}{N}} e^{2\pi i \frac{(l+m+a+b)}{N}} \]

\[ \times e^{-2\pi i \frac{a \sin \theta + b \cos \theta}{N}}. \]

(3)

To evaluate (3), we need to be able to evaluate the sum

\[ \hat{g}(p, q; \alpha, \beta) = \frac{1}{N^2} \sum_{l,m} \hat{f}(l, m) e^{2\pi i \frac{(l+a)+m+b)}{N}} \]

(4)

Manuscript received January 6, 1998; revised September 16, 1998. This work was supported in part by the National Institutes of Health under Grant NS34798. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Jan P. Allebach.

The authors are with the Biophysics Research Institute, Medical College of Wisconsin, Milwaukee, WI 53226-0509 USA (e-mail: rwc@mcw.edu).
for arbitrary $\alpha$ and $\beta$, where $g(l,m) = N^{-2}\exp\{-2\pi i(la + mb)/N\}f(l,m)$. The one-dimensional (1-D) analog of (4) is the computation of

$$
\tilde{h}(p;\alpha) = \sum_l h(l)e^{2\pi ipl/\alpha}.
$$

This can be done efficiently using the chirp-z algorithm [1], [2]; that is, by expanding $2lp = l^2 + p^2 - (l - p)^2$, so that

$$
\tilde{h}(p;\alpha) = e^{\pi i\alpha/2} \sum_l \{h(l)e^{\pi i\alpha l/2}\}e^{-\pi i(l - p)^2/\alpha}.
$$

This expansion is a multiplication, a convolution, and a multiplication. The convolution can be done quickly using three FFT's (of $h(l)\exp\{\pi il^2/\alpha\}$, of $\exp\{-\pi il^2/\alpha\}$, and of the product of their transforms).

A similar method can be used to evaluate (4) efficiently. The integer expansion required is

$$
2(lp + mq) = l^2 + p^2 - m^2 - q^2 - (l - p)^2 + (m + q)^2,
$$

$$
2(mp - lq) = 2lm - 2pq - 2(l - p)(m + q).
$$

With this, (4) can be written as

$$
g(p,q;\alpha,\beta) = Z(q,p)^* \sum_{l,m} \{g(l,m)Z(l,m)\}Z(l-p,m+q)^* \quad (5)
$$

where $Z(p,q) = \exp\{\pi i[(p^2 - q^2)\alpha + 2pq\beta]\}$ and $Z(\cdot)^*$ denotes complex conjugation. This again is 2 multiplications and a convolution, which can be done with three 2-D FFT's. Counting the original FFT to get $\tilde{f}(l,m)$, four 2-D FFT's are required to carry out the rotation. (If the original data is available in Fourier space, as in magnetic resonance imaging [3], the initial FFT can be avoided—this amounts to reconstructing the raw MRI data directly to a rotated grid. If the same $\alpha$ is to be applied to many images, then the FFT of $Z$ need only be computed once.) We note that

$$
Z(p+s,q) = \exp\{2\pi is\alpha\}Z(p,s)^*Z(q,p)^2, \quad \text{and} \quad Z(p,q+s) = \exp\{2\pi is\beta\}Z(p,q)^*Z(p+s)^2, \quad \text{for arbitrary } s.
$$

These relationships mean that the coefficient array $Z(p,q)$ can be computed recursively, in much the same way as the sine/cosine tables needed for the FFT can be generated. (Accuracy must be a consideration when using such a recursive table generator.)

III. THREE-DIMENSIONAL IMAGE ROTATION

The factoring trick that allows the 2-D rotation to be done from Fourier space to image space with a single chirp transformation cannot be generalized to arbitrary higher dimensional rotations. Instead, two chirp transformations are necessary, one of which takes the original image to an intermediate rotated Fourier grid, and the second of which takes this transform to the final rotated image space grid.

As shown by Lawton [4], the chirp algorithm generalizes in $n$ dimensions to

$$
\hat{f}(Mq) = Z(q)\sum_p \{f(p)Z(p)\}Z(q-p)^* \quad (6)
$$

where $M$ is a real symmetric $n \times n$ matrix, $p$ and $q$ are integer $n$-vectors, and $Z(x) = \exp\{-\pi i(x,Mx)/N\}$. (Our presentation is limited to $n$ cubic lattices, unlike [4].) This expression is the discrete Fourier transform (DFT), since

$$
Z(q)Z(q-p)^*Z(p) = \exp\{-\pi i[(p,Mq) + (q,Mp)]/N\} = \exp\{-\pi i(p,M + Mt^T)q)/N\}.
$$

Since $M = M^T$, this is the DFT kernel for evaluation at $k = Mq$. The convolution can be evaluated efficiently using three $n$-dimensional FFT's (of $f(p)Z(p)$, of $Z(p)$, and of the product of their transforms). The coefficients $Z(p)$ can be generated recursively using the relation $Z(p+s)^*Z(p)^2 = \exp\{-2\pi is(s,Ms)/N\}$, for any integer vector $s$.

Lawton’s chirp algorithm can be generalized to allow evaluation on some grids generated by unsymmetric matrices. We have

$$
\hat{f}(A^TMq) = Z(q)\sum_p \{f(p)Z(Ap)\}Z(q-\alpha)^* \quad (6)
$$

where $M$ is still required to be symmetric and $A$ is an arbitrary real matrix. For this generalized convolution to be done using the FFT, $A$ must be an integer matrix. Not all unsymmetric real matrices $B = A^TM$ can be so factored. The real line $R$ is an infinite-dimensional linear space over the field of rationals $Q$. $B$ has $n^2$ elements, while $M$ has only $(n+1)/2$. If the elements of $B$ are linearly independent over $Q$, they cannot be expressed as finite integer combinations of elements from any symmetric $M$. Thus, (6) is more useful than (5) only in special circumstances. In the case of a 2-D rotation, the output grid is generated by the matrix

$$
\begin{bmatrix}
\cos\theta & -\sin\theta \\
\sin\theta & \cos\theta
\end{bmatrix} = \begin{bmatrix}
-1 & 0 \\
0 & 1
\end{bmatrix} = \begin{bmatrix}
\sin\theta & \cos\theta \\
-\cos\theta & \sin\theta
\end{bmatrix}.
$$

This factorization shows that the 2-D rotation algorithm in Section II is an example where (6) is useful, which is why the rotation can be accomplished with a single chirp transformation.

The general $n \times n$ orthogonal matrix $U$ can be factored into two symmetric orthogonal matrices using the technique of [4, Lemma 2]. In the case $n = 3$, the factor matrices can both be expressed in Householder form

$$
U = V W = [I - 2vv^T](I - 2ww^T)
$$

where $\|v\| = \|w\| = 1$. Let $U$ represent rotation about the fixed axis $r$ by angle $\theta$. Then $v$ and $w$ are determined by requiring that $\langle v, r \rangle = \langle w, r \rangle = 0$ and $\langle v, w \rangle = \cos \frac{\theta}{2}$. (We note that $\cos \frac{\theta}{2} = \frac{1}{2}(1 + u_{11} + u_{22} + u_{33})^{1/2}$; the rotation is in the direction from $w$ toward $v$.) The vectors $v$ and $w$ are not unique: replacing $v$ by $U^*v$ and $w$ by $U^*w$ does not change the result. Here, $\alpha$ is any real constant; $U^*$ represents rotation about $r$ by angle $\alpha\theta$ (i.e., $v$ and $w$ can be rotated together along the unit circle perpendicular to $r$). For some orthogonal matrices $U$, it is possible to choose $v$ such that $vv^T$ is rational. In such cases, $U$ can be factored into an integer matrix (a multiple of $V$) and a symmetric matrix (a multiple of $W$). For general orthogonal $U$, this will not be possible, since the set of unit vectors $v$ with $vv^T$ rational is only countable. The set of fixed axis vectors $r$ is uncountable, so not every unit circle can contain such a $v$. (When the 2-D rotation is embedded in $R^3$ with $r = \hat{z}$, the appropriate unit circle is in the $xy$-plane, and the vector $v = [-1 0 0]^T$ can be used.)

Thus, for general 3-D rotations $U$, it is necessary to perform two chirp transformations. The first transformation takes $f(p)$ to $\hat{f}(Wq)$; that is, it evaluates the Fourier transform of the 3-D image on an intermediate rotated grid. (Any phase factors needed for translation should be multiplied into $\hat{f}$ after this first transformation.) The second chirp transformation takes $\hat{f}(Wp)$ to $f(VWq)$, back in image space on the final rotated grid.

A total of six 3-D FFT's are needed to perform these chirp convolutions. The computational load in the first chirp transformation can be reduced somewhat by one further trick. It is always possible to choose $w$ such that $w_3 = 0$. In this case, $W$ is really a 2-D rotation and the 2D algorithm of Section II can be applied to each $(p_1, p_2)$-plane of $f$ separately. A 1-D FFT from $p_3$ to $q_3$ at the end will result in $\hat{f}(Wq)$.

IV. DISCUSSION

Realigning image data to a common orientation is a necessary first step in longitudinal studies. This requires resampling the images to rotated and translated grids. In biomedical applications, high-order interpolation is often needed in order to make subtle changes...

1298 IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 8, NO. 9, SEPTEMBER 1999
over time more visible. For this reason, sinc interpolation has been recommended for such applications [5]. The image space implementation of sinc interpolation can be very slow unless the interpolation kernel is severely truncated, which may result in ringing artifacts. Unser et al. [6], Fraser et al. [7], and Eddy et al. [8] each separately proposed a faster technique using three shearing transformations to perform a 2-D rotation, and implementing the interpolation within each sheared row/column using (1). The methods put forward in the present correspondence clarify the results of [4] and show how multidimensional Fourier transforms can be used to perform image rotations directly. These techniques can be combined with other Fourier space processing, if desired (e.g., sharpening, differentiation, etc.).

REFERENCES


LC–M–S Filters for Image Restoration Applications

Doina Petrescu, Ioan Tăbuș, and Moncef Gabbouj

Abstract—A new filtering architecture is proposed, generalizing some previously introduced multilevel median filters. An efficient design procedure for the new filtering architecture is demonstrated for image restoration application. Simulation results show a good noise rejection performance, combined with a fine detail preservation capability.

Index Terms—Median filters, multistage filtering architecture, stack filters.

Manuscript received December 16, 1996; revised May 5, 1998. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Stephen E. Reichenbach. D. Petrescu was with the Signal Processing Laboratory, Tampere University of Technology, SF-33101 Tampere, Finland. She is now with Nokia Research Center, FIN-33720 Tampere, Finland. I. Tăbuș and M. Gabbouj are with the Signal Processing Laboratory, Tampere University of Technology, SF-33101 Tampere, Finland.

Publisher Item Identifier S 1057-7149(99)06816-5.

Fig. 1. LC–M–S filter architecture.

![LC–M–S filter architecture.](image1)

Fig. 2. Thirteen-pixel window shape.

![Thirteen-pixel window shape.](image2)

I. INTRODUCTION

Median filtering is efficiently used as an impulsive noise smoother for different signal and image processing applications. Although noise rejection is obtained, standard median filters have poor performance in preserving small details in images. The larger the filter window is, the better noise attenuation is obtained, but the worse detail preservation is achieved. Several solutions were proposed to improve the ability of median filtering to preserve particular features. One type of solutions replaced the standard median filter with weighted median or median-type filters optimally designed to preserve features of given shapes and sizes, [8]. Other types of solutions were introduced in [1], [4], as multilevel, nonadaptive, structure-preserving filtering architectures. These methods consist in combining in a median filtering stage the outputs of several subfilters (finite impulse response, median), spanning subwindows with different orientations inside the processing mask. Each subfilter is designed to preserve a feature in one fixed direction. Details covered by the subfilters are well restored, but thin lines along other directions are often distorted. Two-level architectures based on median or related filter classes have also been proposed in [6] and [9].

Almost all these filtering solutions have a positive Boolean function representation in the binary domain, and they belong to the larger class of stack filters [7]. Designing an optimal stack filter for a training...