MULTIRESOLUTION VIEW MORPHING
IN THE WAVELET DOMAIN

Paul Bao
The Polytechnic University of Hong Kong
Dan Xu
Zhejiang University, P. R. China

ABSTRACT

This paper presents a new view-synthesis technique using the 2D discrete wavelet-based view morphing. The view morphing is completely based on pairwise images without the camera calibration and depth information of images. First a *Fundamental Matrix* related to any pair of images is estimated. Then using the fundamental matrix, the pair of image planes is rectified to be parallel with their corresponding points lying on the same *scanline*, giving an opportunity to generate new views with linear interpolation techniques. The pre-warped images are then decomposed into a hierarchical structure with the wavelet transform. Corresponding coefficients between two decomposed images are linearly interpolated to form the multiresolution representation of an intermediate view. Quantization techniques [10,11] can be embedded here to compress the coefficients for the purpose of reducing the morphing complexity. Finally, during the display, compressed images are decoded and an inverse wavelet transform is applied. A post-warping procedure is employed to transform the interpolated views to its desired position.

1. Introduction

Image Based Modeling and Rendering (IBMR) has become very popular recently in the computer graphics and computer vision communities. In IBMR systems a scene is represented by a set of pre-acquired images rather than a geometrical model, and a new view from a novel viewpoint is rendered by adaptively interpolating and combining the existing images.

View morphing [1], proposed by Seitz and Dyer in *SIGGRAPH'96*, introduces a new IBMR method via the image morphing technique. Different from the traditional image morphing concept where the two reference images may be very dissimilar in shape and radiometric [8], and hence so are the intermediate images, view morphing emphasizes the physical reasonability, which means that the intermediate images should be shape preserving. In other word, the results of view morphing should be the same as the projective rendering mosaics from different viewpoints. View morphing algorithm can be summarized as a three-stage process: First, the pre-warp source and destination images are pre-warped into a canonical configuration in the sense that the two image planes are parallel and corresponding points satisfy the scanline property. Second, intermediate images are generated by the image morphing techniques. Third, the intermediate sequences are post-warped into the original configuration. Compared with other IBMR methods, view morphing possesses the advantages that very few reference images are required and no geometry information is needed.

Image morphing is a well-received technique that generates a sequence of *intermediate* images such that the source image is gradually changed into the destination image. The main problem in image morphing is deriving a set of warping functions from specified corresponding features in the source and destination images. The warping functions are 2D geometry transforms that distort the source image towards the direction of the destination image, and vice versa. The two distorted images are then cross-dissolved by properly interpolating their position [7] and radiometric [8].

Mesh-based and field-based morphing are two commonly used image morphing methods. In the former approach, features are specified as a non-uniform control mesh, and a warp function is computed by a *spline interpolation*. The disadvantage of this approach is that the controlling mesh at each step is complicated. The latter provides an easy-to-use interface to specify features,

which are pairwise line segments in the source and the destination. Each line has a field of influence. A warp is computed by taking the weighted average of the influences of line segments. Both approaches are very time-consuming. The computation time is proportional to the image resolution as well as the number of features. In addition, they suffer from the influence of high frequency noise near the features. To accelerate the performance of the morphing and de-noise the influence near the features at the same time, the wavelet transform is a good approach for its multiresolution structure and nice features in both spatial and frequency domain. There has been some work on 3D metamorphosis in frequency domain. For example, in reference [5] Wang and Kaufmann introduce a wavelet-based volume morphing method to realize a smooth transition between two volumetric data sets. Also, the Fourier transform has been used for volume morphing [6].

In this paper we propose a fast view morphing method based on the wavelet transform. This method preserves all the features in both spatial and frequency domain while gaining a significant improvement in performance compared to spatial domain morphing.

2. Wavelet Domain View Morphing

Our approach, which can be viewed as an extension of *View Morphing* [1], consists of the following four steps:

Suppose that I_1 and I_2 are two reference images.

Step 1. **Rectification**: two transformation matrices H_1 and H_2 are found so that H_1I_1 and H_2I_2 have parallel image planes and the corresponding points are on the same scanlines, i.e., their y coordinates are the same. This process guarantees that the interpolation be linear and shape preserving.

Step 2. **Image morphing in wavelet domain:** the two rectified images are first decomposed into pyramid structure by 2D-wavelet transform. Then interpolation techniques are used to construct the coefficients of new views level by level.

Step 3. **Coefficients coding and decoding:** wavelet coefficients are quantified to reduce the complexity and storage requirement, for example, *by vector quantization*. When being displayed, they can then be decoded quickly.

Step 4. **Reconstruction:** an inverse wavelet transform is performed to generate new rectified views. The final views with correct position, orientation, and radiometrics can be made through a post-warping process.

3. Epipolar Geometry and Fundamental Matrix

Since we concentrate on the common camera, the only information available comes from the images themselves. This is of great value in applications where camera parameters may vary due to different conditions of image generation or may not be available at all. Based on a knowledge of stereo vision, the geometric relations between two cameras are described in projective terms rather than in *Euclidean*. This *epipolar* information is entirely contained in a 3×3, rank 2 matrix, called the *Fundamental Matrix* [2,12]. It is the key concept for image measurements as it contains all the geometrical information correlating two different images. To estimate the fundamental matrix, *F,* we need to know the correspondence between two images. We may interactively specify eight or more corresponding feature points instead. For a given point *m* in the first image, the projective representation *l'* of its epipolar line in the second image is given by

$$l' = Fm \qquad\qquad (1)$$

Since the point *m'* corresponding to *m* belongs to the line *l'* according to epipolar geometry, we have

$$mF'm = 0 \qquad (2)$$

Therefore, the parameters of *F* can be estimated by linear least-square method or by a nonlinear method relying on relation (2). We employ a linear method to fit an initial matrix F_0 and then use the nonlinear iterative method to fine-tune a more stable and accurate solution of *F*.

To rectify the known image pair, we need to search for two transformations H_1 and H_2 such that the two warped images satisfy the scanline property. As described in [2], the necessary and sufficient condition under which two images are properly rectified is that their fundamental matrix has the form

$$F = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \qquad (3)$$

which can be multiplied by an arbitrary constant. In [1], Seitz and Dyer introduced a method to search for the rectification transformation H_1 and H_2, based on epipolar constrains of *F*.

4. Wavelet Transform

The wavelet transform has been widely used in computer graphics, such as surface modeling, radiosity, raytracing, texture mapping, volume visualization, and image compression. The wavelet transform possesses a nice feature in localizing image information in both the spatial and frequency domains. In addition, the wavelet transform provides a multiresolution hierarchical structure to image representation: it decomposes a function into a smooth approximation of the original function and a set of detailed information at different resolutions [5,9,10]. This leads to the progressiveness of the wavelet-based image processing techniques. In this section we shall briefly describe the wavelet transform and the 2D-wavelet multiresolution decomposition.

Formally, the smooth approximation of a function $f \in L^2(R)$ at any resolution 2^i is a projection denoted as: $A_{2^i} : L^2(R) \rightarrow V_{2^i}$, $V_{2^i} \in L^2(R)$, and the detail of *f* at any higher resolution 2^j is a projection of f onto a subspace O_{2^j} of $L^2(R)$, which is denoted $P_{2^j} : L^2(R) \rightarrow O_{2^j}, j > i$. We can properly select the projection functions such that the O_{2^j} are orthogonal to each other as well as V_{2^i}. For discrete functions there are two sets of functions which constitute the orthogonal basis of V_{2^i} and O_{2^j}, respectively. We call them wavelet function and scaling function and denote them by

$$y_{j,n} = 2^{-j/2} y(2^j t - n)_{n \in Z} \qquad (4)$$

$$f_{j,n} = 2^{-j/2} f(2^j t - n)_{n \in Z} \qquad (5)$$

Using these two functions, the discrete signal and approximation resolution 2^j are respectively defined as:

$$(D_{2^j} f) = 2^{-j/2} < f(u), y_{j,n} > \qquad (6)$$

$$(A^d_{2^j} f)_n = 2^{-j/2} < f(u), f_{j,n} > \qquad (7)$$

Instead of calculating the inner products in equations (6) and (7), a pyramid algorithm is applied for the decomposition of the function. For 2D discrete image functions, this algorithm is applied sequentially along each dimension (illustrated in Figure 1), where $\tilde{H}(n) = H(-n)$, $\tilde{G}(n) = G(-n)$ are low-pass and high-pass filters respectively, whose impulse response is defined as

$$H(n) = < 2^{-1} f(2^{-1}u), f(u-n) > \qquad (8)$$
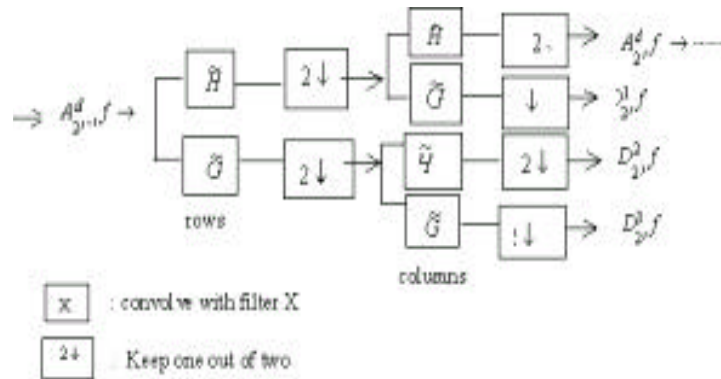$$G(n) = < 2^{-1} y(2^{-1}u), y(u-n) > \qquad (9)$$

Figure 1. Wavelet decomposition for 2D image.

5. Morphing in Wavelet Domain

Due to the spatial localization properties of the wavelet, it is possible to exploit traditional field morphing method in each of the sub-band images in the wavelet domain. We need to locate positions of all the feature lines.

5.1 ENERGY DISTRIBUTION OF A POINT

A simple way to determine the feature lines of source and destination images in the wavelet domain is to scale coordinates of the x- and y-directions of the features by a factor of ½ Unfortunately, experiments show that not all wavelets preserve this spatial property because the DWT is not a point-based transform; it is a vector-based transform. Therefore each coefficient in the wavelet domain may be influenced by many pixels in image space relying on the length of band-passing filters. That is, each pixel in image space will contribute to many coefficients in wavelet domain. In other words, we must know the distribution of energy of a point after a wavelet transform. Figure 2 illustrates the energy distribution of a point under different wavelet transforms. We transform an image with only one point into wavelet domain and observe its energy distribution. The higher the energy of a coefficient, the darker it is.

Notice that the wavelet transform introduces negative coefficients, so the coefficients in the wavelet domain require taking the absolute value. Even so, some influenced coefficients are not visible because their energies are too low and can be neglected. Figure 2 shows that the shorter the filter, the nearer the energy distribution is collected to the ideal position whose coordinates are equal to that of the point in image space scaled by a constant of ½level by level. Inversely, the smoother the wavelet, the larger the regions of energy influenced. Also note that these regions may be disconnected.
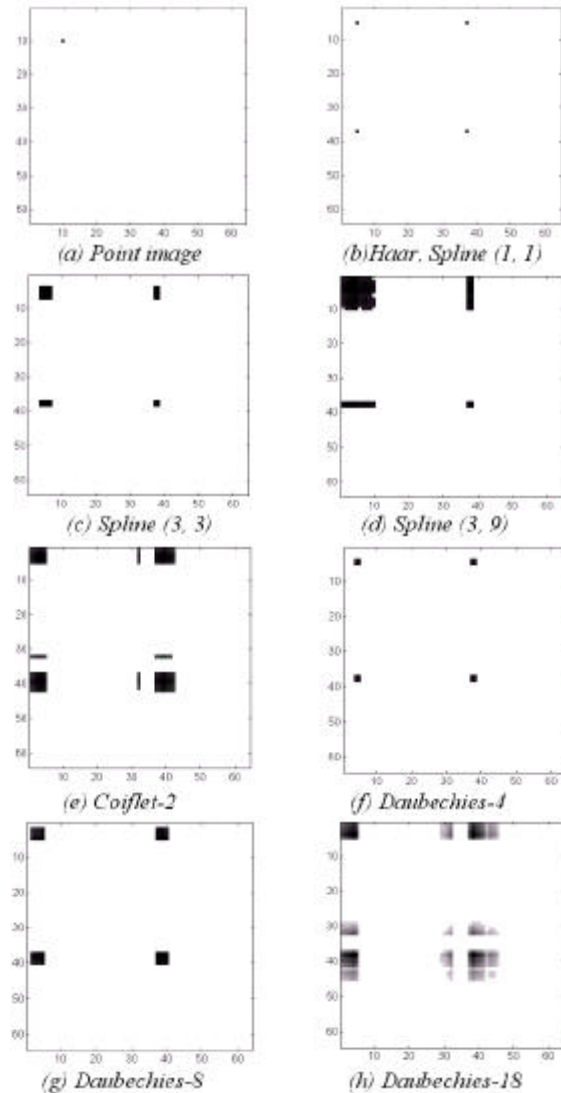
(a) Point image

(b)Haar, Spline (1, 1)

(c) Spline (3, 3)

(d) Spline (3, 9)

(e) Coiflet-2

(f) Daubechies-4

(g) Daubechies-8

(h) Daubechies-18

*Figure 2. Energy distributions of a point
with different wavelet transforms*

5.2 FEATURE LINES OF DIFFERENT DWT

So far we have observed that different wavelet selection may influence the determination of positions of feature lines in baseband as well as detail bands. To obtain feature lines accurately and adaptively, we design a simple algorithm that can quickly determine new positions consistent with the original line positions in the sense that the new control point of a line would not be selected at an unreasonable position.

Feature lines search algorithm:

for each multiscale level
    for each line and each end point of it
        1. Transform the point image into a particular wavelet domain, whose size is the same as that of the source and the destination.
        2. Select four new points from approximated band and detail bands respectively to represent positions of current considered end point.

3. In each subband, create a new line segment corresponding with current processing line.
  end % for line
end % for level

There are several criteria to select control points of the new lines from non-zero coefficients of the DWT-transformed point image in different subbands, such as selecting position of minimum energy, maximum energy, energy mean, and absolute maximum energy coefficients. Our experiment shows that a criterion based on the absolute maximum energy coefficients (i.e. selecting the darkest point as new feature control point) leads to better search performance.

5.3 WARPING OF THE WAVELET COEFFICIENTS

Once we have determined four new sets of feature lines in each subband on a specified level, it is obvious that we can compute wavelet coefficients to generate the distorted source or destination images by directly applying the field morphing algorithm in the wavelet domain. Intermediate images are obtained by interpolating values of wavelet coefficients of the distorted source and destination, then transforming the intermediate images to the image space using the inverse wavelet. Figure 3 shows a sequence of intermediate images: the first and last image are the source and destination respectively. The source images are obtained from Seitz's database.
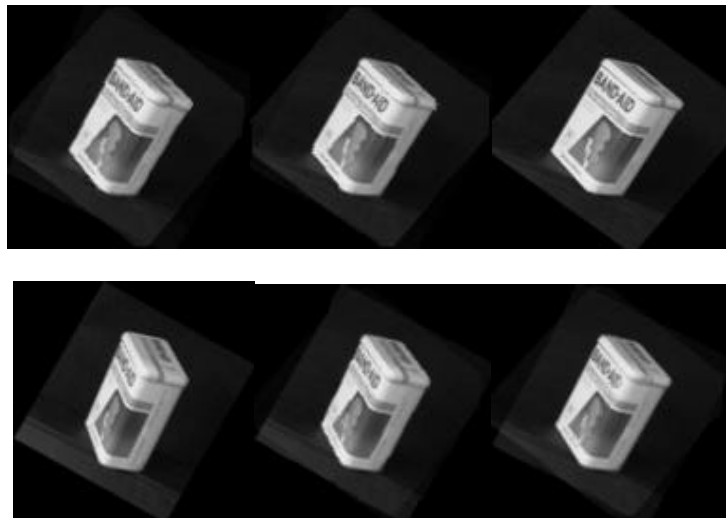

Figure 3. Sequence of transition between
the source and destination images

5.4 COMPARISON USING VARIOUS
KINDS OF WAVELET TRANSFORMS

Since different wavelet transforms have different spatial and frequency features, the new sets of feature lines obtained through the simple method described in section 5.2 may differ from each other. This implies that the morphing algorithm must be performed in each subband. This morphing algorithm is called the *complex algorithm*. If so, there seems no need to use the wavelet transform since it would not be capable of enhancing the performance of the field morphing algorithm. With an analysis of the output of feature lines in different band-passing subbands, we observe that the feature lines obtained in different subbands can be completely identical if we carefully select a proper wavelet.

5.4.1 Orthogonal Wavelet Based Morphing

The Haar wavelet is the simplest one of this kind, and its output is the same in every directional subband. So also is the Daubechies-4 wavelet. Therefore, the morphing algorithm for Haar and Daubechies-4 is performed only once to compute the distortion map for each pixel of the source and the destination images in the baseband. Then the map is used to directly calculate the warped coefficients of the other subbands. This morphing algorithm is called the *fast morphing algorithm*. The approach only uses a quarter of the computation time of the traditional field morphing algorithm. Figure 4 illustrates a morph using Haar and Daubechies-4 wavelet transforms. Figure 5 gives a comparison between the results of fast and complex algorithms with the same wavelet. Both figures give the intermediate image of the sequence shown in Figure 3.
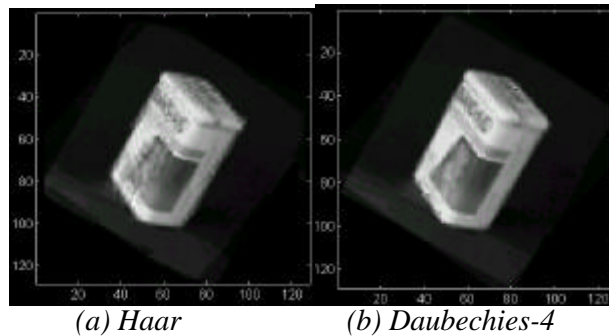

(a) Haar   (b) Daubechies-4

Figure 4. Fast morphing algorithm with simple wavelet

5.4.2. Biorthogonal Wavelet Based Morphing

Though the orthogonal wavelet-based fast morphing algorithms are very efficient, it requires the *orthogonality* of the wavelet, which restricts the choice of the wavelet for some better features. Complex morphing algorithm would give reasonable result for smooth wavelet, but it suffers inefficiency as the traditional field morphing method and introduces artifacts to the morph. To obtain high quality morphing images efficiently, we consider another wavelet family, *biorthogonal symmetric wavelet*. Biorthogonal symmetric wavelet possesses a good feature that the underlying filtering operations, rather than being one-sided as orthogonal wavelet, are two-sided and extended an equal distance into the past and future. Therefore, the fast morphing algorithm can be performed directly without determining the new positions of the feature lines in decomposed wavelet domain. Instead, we just divided coordinates of the feature control point by 2 when resolution is decreased to the half. Figure 2 (b)-(d) illustrate that the energy distributions are collected near the ideal position regardless of the filters selected. Without loss of generality, we use *Spline* wavelet in our experiments. Results are seen in Figure 6.

6. Reconstruction and Progressiveness

In the previous sections, we have discussed how to generate the multiscale representation for interpolated views in wavelet domain. In this section, we will address the issue how the views are reconstructed from the hierarchical representation.

6.1. POST-WARP

To reconstruct the intermediate views in the image space (spatial domain), we need to inversely transform the morphed images from the wavelet domain to the spatial domain and then transform these images from the canonical configuration to the original coordination system.

(a)   (b)

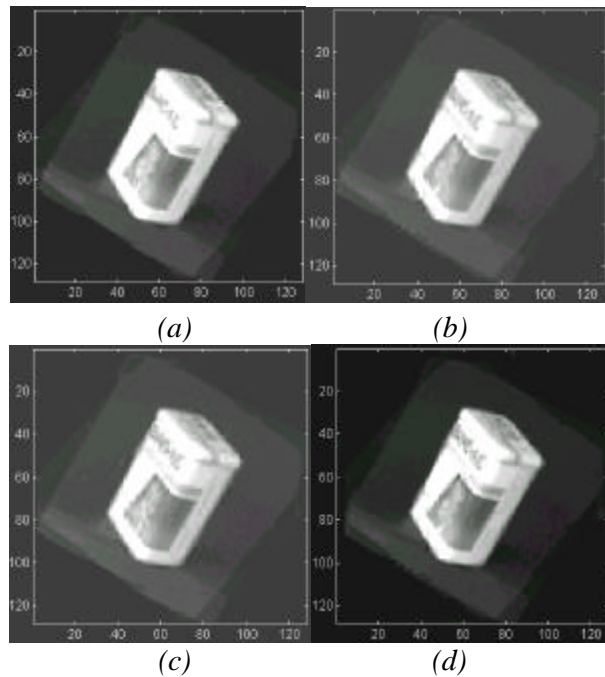*(c)   (d)*

Figure 5. Comparison of the fast and complex algorithms with the same wavelet and (c) are results of fast algorithm using Coiflet-2 and Daubechies-4 wavelet. (b) and (d) are results of complex algorithm using the same wavelet as (a) and (c) respectively.

A simple but effective approach for this is to select four control points of a quadrilateral in two original images separately, and then linearly interpolate the intermediate quadrilaterals. Any two quadrilaterals can derive a transformation, which warp one image into the shape of the other. Therefore, two original images are warped toward a same intermediate image and then cross-dissolve the radiometrics to generate the final morphed images.
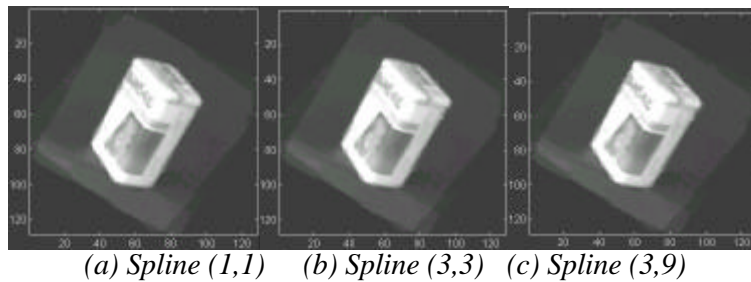

(a) Spline (1,1)   (b) Spline (3,3)   (c) Spline (3,9)
Figure 6. Morphing result using biorthogonal wavelet

6.2 VISIBILITY ANALYSIS

Due to the varying of viewpoints, some points of the scene may become invisible in some views. This gives rise to the problems of holes and folds appearing in the synthesized images. A fold is caused when multiple points are projected to the same position. It is relatively easy to be resolved. The common method is using Z-Buffering techniques. Whereas holes are more difficult to be resolved because we may have insufficient information at the hole position in either of the two original images. Traditional method uses known neighboring points for interpolation; however, it usually gives "ghost" appearing in the synthesized image. In our approach, we try to recover a conceptual surface, which can be used to predict the information at holes and folds.

6.3 PROGRESSIVENESS OF THE
WAVELET DOMAIN MORPHING

Our view of a morphing scheme possesses a nice feature for distributed and networked applications. First, with the multiresolution property of the wavelet transform, we can obtain a coarse morph at the top level. The morph can then be progressively refined level by level. This feature is very suitable for distributed and networked image and video applications, especially for distributed and Internet IBMR applications. Second, with the rich and effective wavelet-based image compression techniques [9,11], a compression scheme can be embedded in a wavelet domain, such as VQ. This would reduce a large number of data transported and the decoding and rendering can be performed at the receiver side.

7. Experimental Results

Compared with field image morphing, wavelet-based image morphing can accelerate the morphing process because wavelet analysis can be viewed as a sub-sampling of the original image, and it is known that field morphing algorithms depend on the resolution and the number of feature lines. Also, since the DWT transforms images into the frequency domain, some frequency-based analysis can be embedded, such as quantization [9], thresholding [11], and noise removal. An interesting effect that cannot be obtained from the spatial image domain based morphing method is mentioned in [3]: changing the transition rate of different frequency bands. For example, we can let the low frequency band change quicker and high frequency band slower, or vice versa. An experimental result is shown in figure 6. It is known that high frequency part of an image corresponds to the detail of the image. On the other hand, the low frequency part is related to the global information of the image. Thus by varying the transition rate of different frequency bands, we can obtain some progressive effect.

A disadvantage of wavelet based image morphing is that the quality of the morphed images is not as good as that of the traditional method. This accounts for sub-sampling of the DWT, but not the wavelet itself. We compared our results with the half size image morphing result from traditional field morphing algorithm and find that the results are the same. To improve the quality of the wavelet-based morphing, we can replace the high frequency part of the wavelet-morphed image with the corresponding high frequency part of traditional morphed image. An alternative method is to zoom in on the original image and then select feature lines to generate a more accurately specification. Of course, we can also get feature lines via an animator.

Table 1 shows the time complexity based on the different levels of the wavelet transform. Level 1 represents the spatial-domain image morphing, whereas levels 2, 3 and 4 represent 2, 3, and 4 levels of the wavelet transform respectively. There is virtually no improvement from level 4 to level 5 for both fast and complex wavelet-based morphing algorithms. That is due to the huge wavelet transform overhead. The experiment was conducted with *Matlab* 5.0 on a 300MHz Pentium PC, which in general was very slow. However, the table illustrates that the wavelet-based image-morphing algorithm can be 5 times faster than that of the spatial-domain image-morphing scheme. With more powerful software and hardware, the wavelet-based image morphing could provide realtime performance.

Level	Spatial domain morphing	Wavelet-based complex morphing	Wavelet-based fast morphing
1	31 seconds		
2		18.5 seconds	14.6 seconds
3		11.8 seconds	9.5 seconds
4		7.4 seconds	5.5 seconds
5		7.3 seconds	5.4 seconds

*Table 1. Performance comparison between the spatial
domain and wavelet-based image morphing*

8. Conclusion

In this paper we described an effective view morphing method in which the main image processing was performed in the wavelet domain. We discuss the influence of using different kinds of wavelets. We also gave different strategies with respect to different wavelet families. Our method requires no camera calibration, which makes it superior for the IBMR system and communication applications.

9. Acknowledgment

REFERENCES

[1] S. Seitz & C. Dyer, "View Morphing", *SIGGRAPH'96, Computer Graphics*, pp.21-30, 1996

{2] T. Beier and S. Neely, "Feature-based image metamorphosis", *SIGGRAPH'92. Computer Graphics*, **26**(2), 35-42, 1992.

[3] G. Wolberg, "Digital Image Warping", IEEE Computer Society Press, 1990.

[4] T. Nishita, T. Fujii and E. Nakamae, Metamorphosis using Bezier clipping. *Proceedings of the First Pacific Conference on Computer Graphics and Applications*, Seoul, Korea, World Scientific Publishing Co., 162-173, 1993.

[5] T. S. He, S. Wang and A. Kaufmann, Wavelet-Based Volume Morphing. *Proceedings of Visualization'94*, Washington DC, IEEE Computer Society Press, 85-92, 1994.

[6] J. F., Hughes, Schedual Fourier Volume Morphing. *SIGGRAPH'92,* Computer Graphics, 43-46, 1992.

[7] S. E. Chen and L. William View Interpolation for Image Synthesis, *SIGGERAPH'93, Computer Graphics,* 279-288, 1993.

[8] S. J. Gortler, R. Grzeszczuk, R. Szekiski and M. F. Cohen, The Lumigraph, *SIGGRAPH' 96, Computer Graphics,* 43-54, 1996

[9] J. M. Shapiro, Embedded Image Coding Using Zerotrees of Wavelet Coefficients, *IEEE Trans. Signal Processing,* **41**(12), 3445-3462, 1993.

[10]   P. Bao, & D. Xu, Complex Wavelet-based Image Mosaics using Edge-Preserving Visual Perception Modeling, *Computers and Graphics,* **23**(3), June, 1999.

[11]   P. Bao, & B. Leung, Wavelet Transform Image Coding based on Visual Perception Modeling

BIOGRAPHIES

Paul Bao is an assistant professor at the Computing Department, the Hong Kong Polytechnic University, Hong Kong. He received his Ph.D. in computer science from the University of Calgary, Canada in 1988. Prior to joining the Hong Kong Polytechnic University, he was on the faculty of the Computer Science Department, the University of Calgary in 1989. Then he joined the IBM Canada Ltd. as a senior staff analyst from 1990 to 1995. Dr. Bao has published over 50 papers on the international journals and conferences in the areas of computer graphics image processing, image-based rendering, interval analysis and neural network.

Contact information:
Dr. Paul Bao
Computing Department
The Hong Kong Polytechnic University
Phone: 852-27667266
Fax: 852-27740842
Email: mailto:csbao@comp.polyu.edu.hk

Xu Dan is currently a PhD candidate at State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China. She was a research assistant in the Computing Department, The Hong Kong Polytechnic University, between September 1997 and September 1998.

Contact information:
Xu Dan
State Key Lab of CAD&CG,
Zhejiang University, Hangzhou, China
Email: mailto:xyxu@ynu.edu.cn