

Scene Transition based on Image Registration for Web-based Virtual Tour

Eun-Young (Elaine)[†] Kang and Ilmi Yoon[‡]

[†]Department of Computer Science
California State University, Los Angeles, USA
eykang@calstatela.edu

[‡]Department of Computer Science
San Francisco State University, USA
yoon@cs.sfsu.edu

ABSTRACT

Web-based Virtual Tour has attractive advantages over traditional 3D rendering approach in that it does not require labor intensive 3D modeling process or high bandwidth for realistic virtual tour yet provides virtual tour with free navigation and immersive experience of walking around through the WWW. In this paper, we focus on presenting a robust image registration technique for a web-based virtual tour system, Easy and Effective Virtual Tour (EEVT). EEVT constructs virtual tour from a set of images. It uses several snap shots of conventional photos without special tools, builds a simple 3D space within each photo using the spidery mesh technique, and expands the virtual spaces by connecting each space together. The connection between images is achieved by image registration, which finds correspondences automatically and estimates transformations. The image registration process is crucial for virtual tour applications in order to compose smooth transitional scenes between two views so that virtual tourists perceive continuous scenes during navigation. Our registration method uses a parametric approach and it includes the following key features: 1) coarse-to-fine hierarchical estimation 2) fast computation based on image feature-correspondences 3) FFT-based global matching 4) automatic outlier removal by RANSAC. The expanded virtual space creates a sense of navigational freedom for virtual tourists with less distorted viewing.

Keywords: Image Registration, Tour Into Picture, View Synthesis, Web-based Virtual Tour

1. INTRODUCTION

Web-based Virtual Tour, more generally, interactive 3D graphics on the WWW (Web3D), has become a desirable and highly demanded application, yet challenging due to the nature of web application's running environment such as limited bandwidth and large computational requirement on the client side. Image-based rendering approach has advantages over a traditional 3D rendering approach in these kinds of Web Applications. Traditional 3D geometry-based graphics transmits 3D geometry with textures to the client for rendering. VRML/X3D, Java3D, and MPEG4 are examples of geometry based Web3Ds. All these geometry-based techniques fail to support photo-realistic virtual tour because geometry-based approach requires labor-intensive effort for modeling, creates huge datasets, and requires intensive computation power.

As opposed to geometry-based approaches, image-based rendering enables skipping the labor-intensive 3D modeling process of photo-realistic scene. As a consequence, the resulting model is much smaller and does not require high bandwidth or intensive computation power at the client side. QuickTime VR [Chen95] and IPIX are well known examples that use panoramic images. The virtual scenes generated from panoramic images directly enables skipping the modeling process, but these image-

based approaches require special cameras or effort to take panoramic views. In addition, QuickTime VR and IPIX provide only one fixed-point navigation (look-around and zooming in-out only) rather than 'walk around (free navigation)', that is a very important feature to provide immersive experience to virtual tourists.

Variations of image-based rendering approach attempt to allow more navigational freedom. Concentric Mosaics [Shum99] provides a much richer user experience by allowing users to move freely in a circular region and observe significant parallax and lighting changes. However, concentric mosaic requires much bigger datasets and more intensive per pixel computation than IPIX or QuicktimeVR. This technique also requires special tools and setups (a number of cameras mounted on a rotating horizontal beam that is supported by tripod) to capture concentric mosaic. Therefore, it becomes a difficult approach for ordinary users with a hand-held conventional camera. Pseudo-3D photo collage is a simpler approach to create virtual walkthrough experience using multiple images [Tanaka02]. While navigating, it provides a smooth transition between two 2D images (fade-out and center-in) based upon spatial-hyperlink specified by the user beforehand and gives the user a sense of motion to the next position even though scene itself is composed of 2D still images. Pseudo-3D is simple and effective tool for ordinary users to make much use of their images in a new spatio-temporal style. But the Pseudo-3D photo collage remains as connections of 2D images and limited to the illusion created by transitions between images, and does not provide free navigation inside the space by the viewers.

Tour Into the Picture (TIP) [Horry 97] constructs a properly textured pseudo-3D geometry space from a single picture using the spidery mesh and then allow a viewer to tour into the scene (Figure 1). TIP is simple, but the result is impressive. A user can feel a plain 2D picture becomes a 3D scene. It can be toured into and viewed from different viewpoints. However, TIP is developed to produce off-line animations with well-controlled camera trajectory. When viewers get close to the walls and turn their orientation more than certain degree, then images may be seriously distorted due to the texture warping of the simple geometry (Figure 2). Tour Into the Picture Revisited [Li01]

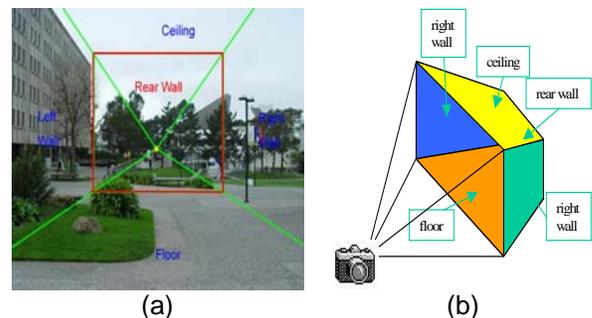


Figure 1: A spidery mesh on top of a photo and its matching 3D scene that will be texture-mapped from the portions of the photo.

observed a problem of TIP's original approach; the visual quality drops drastically when the viewpoint tours into the scene. It contradicted the real world experience where visual quality increases when viewer gets closer. Authors extended TIP by introducing the use of multi-resolution representation of the picture that the visual quality keeps nearly unchanged in the touring. However, navigating within a single scene created by a TIP makes virtual tourists feel confined and instigate a strong desire to navigate further beyond the given scene. Therefore, expanding virtual space by connecting multiple scenes became an essential task.

Inspired by TIP, we introduced Easy and Effective Virtual Tour (EEVT) on the WWW [Yoon05] and extended the use of TIP with multiple images. EEVT constructs virtual tour using multiple snapshots of conventional images without special tools, build simple 3D space within each photo using spidery mesh, and expand the virtual spaces using user intervention to specify correspondence. Since this system uses multiple images to expand virtual space that are consistent views with the user's arbitrary navigation, it is required to estimate inter-image transformation so that a smooth transition between the user viewpoints is automatically created. We describe EEVT in chapter 2 and focus on presenting image registration technique that enables smooth transitions for virtual tourists in remaining chapters.

2. Easy Effective Virtual Tour

Our Web-based Virtual Tour consists of two parts. One is EEVT-Maker that allows content developers to build virtual space with several snapshots of conventional images taken from hand-held cameras and the other is EEVT-Navigator for navigation of the built virtual space on the web.

2.1 EEVT-Maker

The web-based virtual tour is constructed using EEVT-Maker by following the steps below.

A. Photo connection

This step allows the virtual tour content developer (in short, developer) to specify the topology of input images that can represent non-planar relation between images such as up-

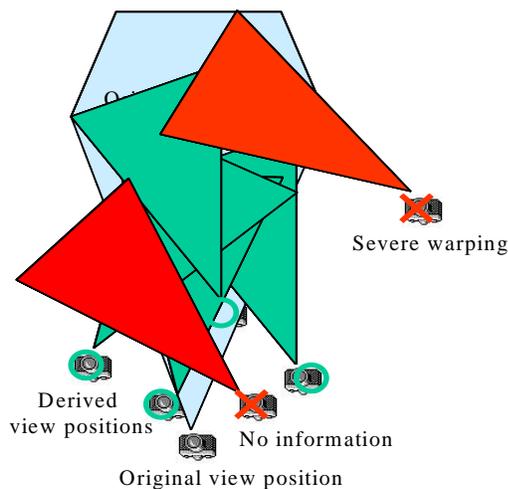


Figure 2 – TIP technique creates a virtual space from a single image and then new views (camera with green circle) can be derived by navigating the scene. However, certain views (camera with red cross) introduce severe warping or tried to see area where no information is available from a single scene. Therefore blocking/controlling of user movement based on view direction and positions is desirable.

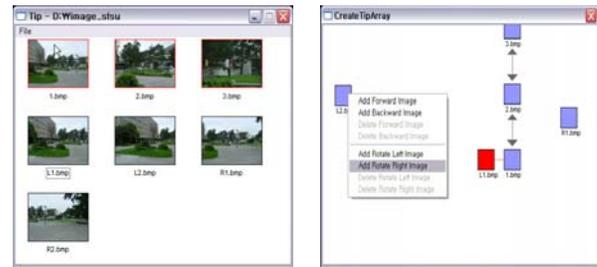


Figure 3 – Screen capture of photo connection in EEVT-Maker that shows photos in the folder, let users to drag and drop into the connection canvas where each photo is represented with a icon with the file name and then relation between images can be built by choosing the relation on the pop-up menu. Currently, possible relations are “Add forward image”, “Add backward image”, “Add Rotate Left image”, “Add Rotate Right image” as well as deletions.

forward, up-left, up-forward-left, down-forward, down-left, or down-right, etc. (Figure 3).

B. Single Image Spidery Mesh Setup

Spidery mesh consists of one vanishing point, four radial lines, and two rectangles over the original picture (Figure 1a). The four radial lines radiate from the vanishing point. Each edge of the inner rectangle is parallel to one edge of the outer rectangle. The inner rectangle is used to specify the rear window in the 3D space. As a result, a scene of 5 walls (rear, floor, ceiling, left and right walls) is constructed with textures derived from the specified regions of the original image. For developer's convenience, EEVT uses default setting of spidery mesh for each photo and then spidery mesh for each image is automatically generated. Developer can always check and change the default setting interactively. This spidery mesh creates a 3D space where viewer can navigate within (Figure 1b).

C. Transition between Scenes

QuickTime VR or IPX uses tripod to take panoramic views, align and seamlessly connect images. Sometimes images are taken with lots of overlaps and then automatically/semi-automatically restore camera parameters to connect images. In either case, connected images result in cylindrical panoramic views and viewers can only rotate around or zoom in & out from a fixed point. In EEVT, each scene constructs its own 3D space and users may freely navigate inside each scene. Then, image registration recovers inter-image transformation in order to align both scenes and provide transition to the next scene. This step will be explained in later chapter in more detail.

The major objective of this EEVT is easy-of-use in its creation and ordinary amateurs with only conventional hand-held camera should be able to create virtual tours without knowing special image mosaic or registration knowledge. We only expect developers will take images with certain amount of overlaps, so that EEVT-maker can find correspondence automatically.

D. Navigator Guidance

When virtual scene is constructed by connecting more than handful images, then the virtual space become complex structure to navigators' impression. One of frequent complaints from the Web3D applications such as VRML with complex structures is that users easily get lost in 3D space. Users can come back to original position using reset, but it would be better to give good guidance about navigators trajectory or current location. Therefore, proper user guidance such as mini maps becomes an important feature for pleasant navigation as easily seen in many 3D games like Quake. The EEVT-maker can create a simple

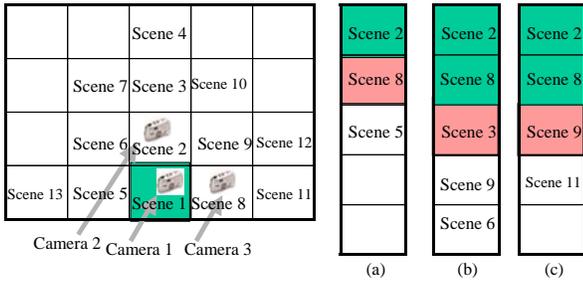


Figure 4 – (a)(b)(c) shows the download queue when viewer’s position moves from camera1 to camera2 or camera 3. Green cell represents the scene is downloaded and pink cell means the scene is currently being downloaded.

map automatically based on the relation between images and show the map along with view direction and location as well as trajectory during navigation. The map can be toggled in & out as needed by user.

2.2 EEVT-Navigator

The EEVT Navigator is the web-based viewer program, that is, IE & NS plug-in that allows viewers to navigate the virtual scene using their browsers. It provides walk-in and look-around navigation at each scene and smooth transition from one scene to the others as designed at EEVT-Maker. By making use of scene relations specified at EEVT-Maker, the EEVT-Navigator program can provide smart streaming that download the first scene and allow viewers to start navigation while downloading the next scenes according to viewer’s moving direction or orientation (Figure 4).

In summary, the developed EEVT is an easy to use (for content developer) and pleasant (for virtual tourists) multiple scene connection to expand virtual space as freely and many as content developers want. EEVT easily generates real time photo-realistic virtual tour from a few clicks of images, and provides realistic ‘walk-around’ visualization to the users by adding little overhead (less than 1k for each image) for storage and transmission to the original images.

3. IMAGE REGISTRATION FOR TRANSITION

It is vital for virtual tour to provide smooth transitional scenes based on the user’s navigation patterns (zoom and translation). In other words, new views need to be composed from the input images. In order to do so the accurate inter-image transformation should be estimated. In this paper, we propose to use a method estimating inter-image transformation using 2D image registration technique and use the recovered transformation to compose new intermediate views.

Seamless image registration or mosaics has been an active research area within computer vision for more than a decade [Sawhney99][Kang04]. QuickTime VR and IPIX also make use of this multi-image alignment to create panoramic images as a preprocessing and then allow viewers to see a portion of the panoramic image through a window. In EEVT, each scene constructs its own 3D space and users may freely navigate inside each scene. Unlike the panoramic view mosaic that connects images during development as a preprocessing step, EEVT estimates inter-image transformation represented using a few parameters during the construction stage and then uses the estimated transformation parameters during actual navigation for aligning both scenes and providing a smooth transition from one scene to another. The main reason that we do not generate a panoramic mosaic is that typically mosaics cannot provide an accurate representation for highly overlapping images with different resolutions. Mosaicing process usually degrades some

of input images either by subsampling or by supersampling. Considering that we allow the developer to capture images in different resolutions for zoom navigation, the more accurate intermediate view will be generated from compositing two or three input images on demand rather than from a mosaic image.

Most 2D image registration methods are parametric approaches, which recover affine, homography or higher order (eg. quadratic) parametric models [Irani96] [Szeliski97] [Morimoto98] [Sawhney99] [Kang04]. In parametric approaches, parameters’ initial values are guessed in various ways. Then, they are iteratively refined or re-estimated in the direction of reducing the errors between images using energy minimization method such as Levenberg Marquardt. The major limitation of all these methods is that they neglect to handle large inter-image motion and often suffer from a local minima problem. Our 2D image registration method is also a approach. However, our approach makes use of additional features in order to avoid the typical problems involved with parametric approaches and make the estimation process robust and fast. The features are 1) coarse-to-fine hierarchical estimation 2) fast computation based on image feature-correspondences 3) FFT-based global matching 4) automatic outlier removal by RANSAC.

In general, the estimation of parameters is highly dependent on the initial values and unfortunately arbitrary initialization tends to lead to a local minimum. Such limitations can be overcome by estimating the initial parameters’ values using global matching in the frequency domain [Reddy96]. However, the search for the parameters in the frequency domain is computationally expensive. Therefore, we use hierarchical approach to reduce the computation cost by limiting the expensive frequency matching in the coarsest level only. Our image registration process starts from creating multi-resolution image hierarchy for each photo. At the coarsest level, we convert images into their frequency representations, match overlapping images, and compute the initial parameter values. The estimated initial parameter values are then propagated to the second coarsest level. In the second coarsest level, we select a set of image feature points from each image, determine correspondences between these sets using the propagated parameters, and estimate parameters using correspondences. The estimated parameters and feature points are propagated to the level and in that level the same steps are performed except feature selection. This process repeats until it reaches the finest level of the input image resolution. The parameter estimation step at each hierarchy level uses RANdom SAMpling Consensus (RANSAC) that detects outliers and increases the robustness of the parameter estimation. For Virtual Tour, this registration process is performed only once during the development time and our performance test result shows that the registration process is fast and robust for many non-trivial input sequences [Kang05].

3.1 Dominant Motion Estimation

To select good initial parameters, we recover initial parameters in the frequency domain. This is based on FFT (Fast Fourier Transform) and searches for the optimal match according to information in the frequency domain [Reddy96].

Let I_1 and I_2 are the two input images that differ only by a displacement (x_0, y_0) .

$$I_2(x, y) = I_1(x - x_0, y - y_0)$$

The corresponding Fourier transform F_1 and F_2 will be related by

$$F_2(\xi, \eta) = e^{-j2\pi(\xi x_0 + \eta y_0)} * F_1(\xi, \eta)$$

The cross-power spectrum of two images I and I' with Fourier transforms F and F' is defined as

$$\frac{F(\xi, \eta)F'^*(\xi, \eta)}{|F(\xi, \eta)F'(\xi, \eta)|} = e^{j2\pi(\xi\gamma_0 + \eta\gamma_0)}$$

where F^* is the complex conjugate of F .

The translation property of Fourier transform (Fourier shift theorem) guarantees that the phase of the cross-power spectrum is equivalent to the phase difference between the images. By taking the inverse Fourier transform of the representation in the frequency domain, we will have an impulse function; that is, it is approximately zero everywhere except at the displacement that is needed to optimally register the two images. In polar coordinates, the rotation between two images appears as translation. In the same way, the scale between two images appears as translation in log-polar coordinate. Therefore, we get the scale and rotation parameters by converting the inputs to different coordinates and computing cross-power spectrums. More precisely, the input images are converted into log-polar coordinates. If there is a significant scale change, the system rectifies the images with respect to the scale. Then, the rectified images are converted into polar coordinates to recover the rotation parameters.

3.2 Selection of Feature Points and Correspondences

The process of recovering parameters minimizes the gray level error using Least Squares measure such as:

$$E = \sum (I_j(x, y) - I_j(M_{ij}(x', y')))^2$$

where M_{ij} is a transformation. (x, y) and (x', y') are corresponding points. The error is measured only for correspondences between selected feature points (feature-based). Feature-based estimation enables the parameters to be recovered very fast. Features can be defined as corners [Zoghلامي97], high curvature points or lines and so on. In this paper, the features are extracted by the Harris corner detector or given by the developer. The Harris corner detector computes the locally averaged auto-correlation matrix derived from the image gradients, and then computes the eigenvalues of the auto-correlation moment matrix to compute a corner "strength", minimum values of which indicate the corner positions.

Feature-based methods are relatively vulnerable to noise or moving objects. Also, if the features are not well distributed over the image, the measured error misleads the parameter estimation. The drawback of feature-based approach can be reduced by updating (adding and rejecting) features in each iteration of the parameter estimation and enforcing the selected features to be evenly distributed over the image.

After extracting feature points from two images, we determine initial correspondences. The correlation of a correspondences is measured by Cross Correlation (CC) or Sum of Squared Differences (SSD). Two different measurement units can be used in different levels of parameter estimation. In general, computing CC takes longer time than computing SSD. In our approach, the CC method is performed in the coarse level of resolutions with a large window size, and SSD is computed in the finest level of resolution with a small size of inputs.

$$CC(f, g) = \frac{\sum ((f - \bar{f})(g - \bar{g}))}{\|f - \bar{f}\| \|g - \bar{g}\|}, \text{SSD} = \sum_i (f - f_i)^2$$

3.3 Coarse-To-Fine Refinement

In the first step, we create Gaussian pyramids for each input image. Then, the initial parameters are estimated in the coarsest resolution. Gaussian image can reduce noise corresponding to the high frequency components of a image and this makes the computation in frequency domain more robust [Burt 83]. The

rotation or scale parameters may not be accurate in the coarsest level because when the polar coordinate and log-polar coordinates are represented in a small image resolution, the angular element becomes quantized too much. But in the presence of large rotation or scale changes, rough initial values are still useful.

3.4 Parameter Estimation

For our virtual tour system, we recover a homography between two images. Homography is defined as:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} \sim \begin{pmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

$$x' = \frac{(p_{11}x + p_{12}y + p_{13})}{(p_{31}x + p_{32}y + p_{33})}, y' = \frac{(p_{21}x + p_{22}y + p_{23})}{(p_{31}x + p_{32}y + p_{33})}$$

By using the initial translation parameters and correspondences, we minimize the error:

$$E = \sum (I_j(x, y) - I_j(M_{ij}(x', y')))^2$$

Let us denote (x, y) and (x', y') two corresponding feature points. We obtain the following equations.

$$\begin{pmatrix} x & y & 1 & 0 & 0 & 0 & -xx' & -yx' \\ 0 & 0 & 0 & x & y & 1 & -xy' & -yy' \end{pmatrix} \begin{pmatrix} p_{11} \\ p_{12} \\ p_{13} \\ p_{21} \\ p_{22} \\ p_{23} \\ p_{31} \\ p_{32} \end{pmatrix} = \begin{pmatrix} x' \\ y' \end{pmatrix}$$

We use the linear least-square method to compute the parameters. The parameters obtained in coarser resolution are propagated to the finer resolution. For proper propagation to a higher level, the translation parameters are scaled up with respect to the scale factor used to create the image pyramid hierarchy. Based on the newly propagated parameters, we adjust the locations of feature points and target points and repeat the error minimization. Unlike the conventional linear least-square method, we add the following iterative refinement step in order to approximate the non-linear minimization method [Hartley97].

In projective transformation, the correspondence is given by $x' = X/W$ and $y' = Y/W$ where

$$X = (p_{11}x + p_{12}y + p_{13}), Y = (p_{21}x + p_{22}y + p_{23}), W = (p_{31}x + p_{32}y + 1)$$

Note that in the projective transformation case, this Linear Least Square method minimizes the error term

$$\varepsilon = (X, Y) - W(x, y)$$

But we wish to minimize

$$\varepsilon = (X/W, Y/W) - (x, y) = (x', y') - (x, y)$$

If the equation had been weighted by the factor $1/W$, the resulting error would have been what we want to minimize. Since W is dependent on (x, y) , we cannot use a fixed weight, W , in the equation until we solve the equation. Therefore, we proceed iteratively to adapt W . Let's denote the weight in the first step as W_0 . In the next step, we can compute W_1 by finding P_{31} and P_{32} . We repeat this process at each step by multiplying the equation by $1/W_i$. If the number of repeated steps is n , the error measure by this process will be

$$\varepsilon = (Xn/Wn, Yn/Wn) - (x, y) = (x', y') - (x, y)$$

It approximates the error that we want to minimize. As an experiment, we estimated the parameters with or without this

step using the same four correspondences between images and the result proved that the iterative refinement would lead to significantly more accurate parameters for most cases.

3.4 RANSAC

Traditional least-square algorithms consider all correspondences to compute the desired parameters. If there are outliers within correspondences, the estimated parameter values are usually far off from the real parameter values. To prevent this situation, we use the RANdom SAMple Consensus (RANSAC) in the parameter estimation stage. RANSAC is different in that it attempts to eliminate the invalid matches. As stated by Fisher and Bolles [Fisher 81], RANSAC uses as small an initial data set as possible and enlarges this set with consistent data when possible. It partitions the data set into inliers and outliers based on a distance threshold, t . In our approach, we use the symmetric transfer error, d , as the error metric.

$$d = \sum_k [d((x, y)_k, M_{ij}(x', y')_k) + d(M_{ij}(x, y)_k, (x', y')_k)]$$

The idea is following: 3 or 4 points are selected randomly to estimate the affine or projective transformation. Then, the support for this transformation is measured by the number of points that match the transformation within the distance threshold, t . The random selection (called a sample) is repeated until the number of selection reaches the preset maximum iteration number or adaptively determined number. After trials, the largest consensus set is selected to estimate transformation parameters. As described, RANSAC performance is dependent upon the selection of points and the number of samples. In our work, for a sample in each iteration, we also use bucket-based selection to enforce points to be evenly distributed over the entire image. Also, the chosen sample is normalized before the estimation in order to increase stability of sample data.

4. RESULT

Figure 5 shows five 800x600 SFSU campus virtual tour images that were used for constructing virtual tour. Figure 6 shows an experimental result, that is, a mosaic image constructed from a subset of the images in figure 5 using our image registration method. It demonstrates the accuracy of the parameter recovery and verifies the ability of creating seamless transitional scenes for arbitrary navigation. Several new views generated by EEVT are captured and presented in figure 7.

The early version of EEVT is available at <http://tlaloc.sfsu.edu/~yoon/WVT> in both Web browser version and stand-alone version. The experimental result shows that EEVT-Navigator running on Pentium III 1.0 GHz, 256 Mbyte main memory on the LAN or cable modem (1M bps) environment and the EEVT-Navigator enabled smooth real time navigation (> 5fps) of the whole virtual scenes with the relatively large image size (800x600). Image registration processes 7~10 pairs of images per second on Pentium III 1.0 GHz, 512 Mbyte with 320x240 image size, which was measured under fully automatic mode. It produced little residual error and very accurate registration for many challenging pairs of images. When the user gives the correspondences, image registration time is negligible.

In the future, we plan to make use of higher resolution by supporting progressive transmission because digital cameras can capture much higher resolution than 800x600 these days. Also, we plan to integrate EEVT and the image registration module that are not fully combined at current.

5. CONCLUSION

In this paper, we presented a Web-based Virtual Tour System, EEVT and technical details of image registration that can

estimate parameters between images in order to produce smooth transitional scenes to the user. The developed EEVT is an easy to use and pleasant to tour with multiple scene connection. Image registration for EEVT's contributes to connect images and allows us to composite smooth transitional scenes between two views so that the expanded virtual space creates tremendous navigational freedom for virtual tourists with much less distorted viewing.

6. REFERENCES

- [Aaron99] Aaron E. Walsh, Mikael Bourges-Sevenier, "Core Web3D," Prentice-Hall.
- [Antonini92] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using the wavelet transform", *IEEE Trans. OnImage Processing*, vol. 1, pp. 205-220, April 1992.
- [Burt83] P. Burt and E. Adelson, The Laplacian Pyramid as a Compact Image Code, *IEEE Trans. on Communication*, Vol. 31, No. 4, 1983.
- [Chen95] Chen, Shenchang Eric."QuickTimeVR - An Image-Based Approach to Virtual Environment Navigation," Computer Graphics, Proceedings of SIGGRAPH 1995.
- [Fisher81] M. A. Fisher and R. C. Bolles, Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, *Comm. Assoc. Comp.* Vol. 24, No. 6, pp. 381-395, 1981.
- [Hartley97] R. Hartley and P. Strum, Triangulation, *Computer Vision and Image Understanding (CVIU)*, 1997.
- [Horry97] Youichi Horry, K. Anjyo, K. Arai, "Tour into the Picture: Using a Spidery Mesh Interface to Make Animation from a Single Image," *Siggraph 1997*, pp 225 ~ 232.
- [Irani96] M. Irani, P. Anandan, J. Bergen, R. Kumar, and S. Hsu, Efficient Representations of Video Sequences and Their Applications. *Signal Processing: Image Communication, special issue on Image and Video Semantics: Processing, Analysis, and Application*, Vol. 8, No. 4, May 1996.
- [Kang04] Eun-Young Kang, Isaac Cohen, and Gerard Medioni, "A Layer Extraction System based on Dominant Motion Estimation and Global Registration", *IEEE International Conf. on Multimedia Expo (ICME)*, June 2004.
- [Li01] Li, N., Huang, Z., "Tour into the Picture Revisited," *WSVG 2001*, pp 41 ~ 48.
- [Morimoto98] C. Morimoto and R. Chellapa, Evaluation of Image Stabilization Algorithms, *Proceedings of IEEE ICASSP*, May, 1998.
- [Netscape] Netscape Plug-in Developer's Guide, <http://developer.netscape.com/docs/manuals/communicator/plugin/index.htm>.
- [Reddy96] B. S. Reddy and B. N. Chatterji, An FFT-based Technique for Translation, Rotation and Scale-Invariant Image Registration, *IEEE Trans. on Image Processing (IP)*, Vol. 5, No. 8, 1996.
- [Sawhey1999] H. Sawhney and R. Kumar, True Multi-Image Alignment and Its Application to Mosaicing and Lens Distortion Correction, *IEEE Trans. on PAMI*, Vol. 21, No. 3, 1999.[Shum 99] Shum, H.Y., He, L.W., "Rendering with Concentric Mosaics," *Siggraph 1999*, pp 299 ~306.
- [Sree00] Sree Kotay, "Streaming, Scalability, and the Viewpoint Experience Technology," Viewpoint white paper available at <http://www.viewpoint.com/developerzone/5-0.html>
- [Szeliski97] R. Szeliski and H. Shum, Creating Full View Panoramic Image Mosaics and Environment Maps, *Proceedings of ACM SIGGRAPH*, pp. 251-258, 1997.
- [Tanaka02] Tanaka, H., Arikawa, M., Shibusaki, R., "Pseudo-

3D Photo Collage,” Siggraph 2002, Sketch, Application & Web Graphics, pp 317.

[Yoon05] I. Yoon, A. Kang, J. Roberts, and S. Yoon, Easy and Effective Virtual Tour on the World Wide Web, SPIE Electronic Imaging, San Jose, California USA, January 16-20, 2005.

[Zoghiami97] I. Zoghiami, O. Faugeras and R. Deriche, Using Geometric Corners to Build a 2D Mosaic from a Set of Images, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 1997.



Figure 5 – original photos taken for constructing SFSU campus virtual tour



Figure 6 – Mosaic image constructed by image registration. It demonstrates the accuracy of the parameter recovery and verifies the ability of creating seamless transitional scenes for arbitrary navigation.



Figure 7 – New views rendered at WVT-Navigator. New views on each column are derived from the same column in figure 7.