논문 2005-42SP-3-6

복수 투영면을 사용한 도심지 가시화용 3 차원 모자이크 기술

(3D image mosaicking technique using multiple planes for urban visualization)

전 재 춘*, 김 형 석**

(Jaechoon CHON and Hyongsuk KIM)

요 약

도심지 풍경을 3 차원 가시화 하는데 적합한 새로운 모자이크 방법을 제안하였다. 도심지에서 도로를 따라 촬영한 일련의 영상들을 2 차원 모자이크하면 입체감이 느껴지지 않는다. 제안한 방법은 도심지의 도로를 따라 나타나는 물체들을 연속 촬영하여, 이 영상들을 대표하는 복수의 투영평면들을 구한 후, 도로를 따라 촬영한 영상들을 이 복수의 평면들에 투영함으로써 입체감을 얻게 하는 것이다. 이를 위해서 연속 촬영한 영상들 간의 특징 점들을 정합함으로써 3 차원으로 표현된 복수의 투영면을 우선 얻는다. 이 복수의 평면들은 인접 평면 간에 상호 연결하여 물체들의 개략적 윤곽 면을 구성하는데, 이 복수 평면에 2 차원 영상들을 투영함으로써 영상에 대한 입체감을 부여하는 기술이다. 이 논문에서 실제 도심지 영상을 3 차원 가시화 시키는 알고리즘에 대한 데모를 보였다.

Abstract

A novel image mosaicking technique suitable for 3D urban visualization is proposed. It is not effective to apply 2D image mosaicking techniques for urban visualization when, for example, one is filming a sequence of images from a side-looking video camera along a road in an urban area. The proposed method presents the roadside scene captured by a side-looking video camera as a continuous set of textured planar faces, which are termed "multiple planes" in this paper. The exterior parameters of each frame are first calculated through automatically selected matching feature points. The matching feature points are also used to estimate a plane approximation of the scene geometry for each frame. These planes are concatenated to create an approximate model on which images are back-projected as textures. Here, we demonstrate an algorithm that creates efficient image mosaics in 3D space from a sequence of real images.

Keywords: 3D image mosaics, multiple planes, urban visualization

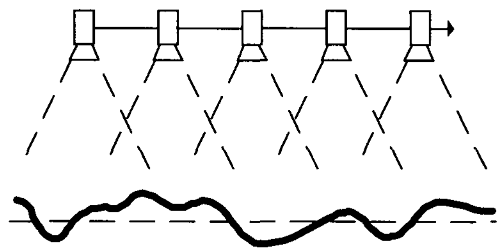
I. Introduction

By now, there are many Internet websites that supply still images for urban visualization. However, urban visualization based on still images is extremely monotonous because of the images' fixed angle and viewpoint. Recently, several websites have begun to supply panoramic images and 3D GIS viewer[1]. Although the panoramic images surrounding a viewpoint enables the user to pan and zoom inside the environment, the viewpoint of the panoramic mosaics remains fixed, just like with other still images. In contrast to the panoramic mosaics, the 3D GIS viewer allows users to view urban scenes from an arbitrary viewpoint and angle. The 3D data and real images used in the 3D GIS viewer are generally

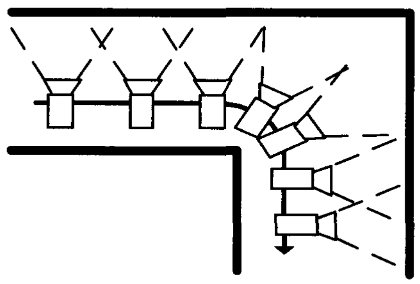
* 학생회원, 동경대학
(The University of Tokyo)
** 정회원, 전북대학교
(Chonbuk National University)
※ 본 연구는 2004년도 메카트로닉스의 지원에 의하여 이루어졌습니다.
접수일자: 2004년10월26일, 수정완료일: 2005년2월11일

obtained by using airborne and vehicle-borne systems. The system is composed of a laser range, CCD camera, IMU, GPS, and so on, and enables users to extract a detailed 3D surface of urban objects with high resolution and texture [2],[3]. The user using stereo matching algorithm also can obtain a detailed 3D surface of urban objects from a sequence of images along a road in urban areas[4],[5]. Unfortunately, it is difficult to apply these detailed 3D surfaces to current Internet or mobile systems because the speed is limited in terms of real-time transmissions. Also, air- and vehicle-borne systems are extremely expensive and many processes are needed to obtain the 3D data. To reduce the cost and the difficulty of creating suitable urban visualization on websites, image mosaicking is the efficient technique. However, it is difficult to apply a sequence of images taken directly from an urban area (see Fig. 1(b)). Therefore, we propose a novel image mosaicking technique suitable for 3D urban visualization on websites.

This paper proposes a novel image mosaicking technique that creates an image mosaic in 3D space using our proposed multiple planes. The proposed method presents a roadside scene captured by a side-looking video camera as a continuous set of textured planar faces, called "multiple planes" (see Fig. 1(b)). The first step in the proposed method is to use feature points to calculate the camera's exterior parameters. A plane approximation of the scene geometry for each frame is automatically selected and matched by using a hierarchical strategy that includes the Smallest Univalue Segment Assimilating Nucleus (SUSAN) operator[14] and the contours tracking algorithm[15]. Let the approximated plane be an independent plane. The exterior parameters of each frame are calculated by using a classical non-linear space resection based on collinearity conditions[17]. The matching feature points are also used to estimate a plane approximation of the scene geometry for each frame. These planes are concatenated to create an approximate model on which images are back-projected as textures. We



(a) The projective plane of 2D image mosaic



(b) A moving camera in a turning point

그림 1. 2D 영상 모자이크 기술의 개념.
Fig. 1. The concept of 2D image mosaicking technique.

demonstrate that the proposed algorithm creates effective image mosaics in 3D space from a sequence of real images taken from urban areas.

Ⅱ. Related Work

An image mosaicking technique that builds an image covering a large area by registering small 2D images to a plane can be used in many different applications including the creation of satellite imagery mosaics[6], virtual reality environments[7], medical image mosaics[8], and video compression[9]. Generally speaking, there are two major types of image mosaicking techniques. In the first type, images of a perspective projection and ortho-satellite imagery, which is created by Direct Linear Transform (DLT) based on spatial data such as the Digital Element Model (DEM) and the Ground Control Point (GCP) [17], are registered to given 2D/3D data [6]. In the second type of mosaic technique, images of a perspective projection are conjugated without the 2D/3D data. This technique enables users to obtain 3D data and to extract textures. Our research here pertains only to the second type of technique. The image mosaicking techniques of the second type fall mainly into four categories: 360 degree panoramic mosaics based on a cylinder plane projection[7],[10], spherical mosaics based on a spherical plane projection[11], 2D image mosaics based on a single plane projection[12], and x-slit images that create image mosaics without a projective plane[13]. Because of the discrete transition in a sequence of panoramic and spherical mosaics, the walk-through in virtual reality is uneven. On the other hand, the 2D image mosaics enable us to view a wide urban scene and also to conveniently extract the texture of broad building facades. The 2D image mosaicking technique creates an image mosaic by projecting input images to a single plane (see Fig. 1(a)). If a perspective equation is used, the single plane is automatically set by the pose of the first frame and by the average depth of the pairs of feature points selected at the first and second frames. The perspective equation is expressed as follows:

$$x' = (ax + by + c)/(dx + ey + 1)$$

$$y' = (fx + gy + ch)/(dx + ey + 1)$$

where a, b, c, d, e, f, g, and h are derived by using the classical least square model with the pairs of selected feature points. Unfortunately, when taking a sequence of images from a side-looking video camera along a road in an urban area (see Fig. 1(b)), it is not effective to apply 2D image mosaicking techniques. Even though the x-slit image algorithm can create an image mosaic from the sequence of images, the image motion of each frame is limited to less than a single pixel to create an image mosaic with a high resolution. In addition, it is difficult to accurately calculate camera orientation within 0.1 pixels to create well-aligned x-slit images. Because the distance between the urban objects and the moving camera is very small and the image motion is generally over 10 pixels, applying the x-slit image algorithm to a mosaic of an urban area is not effective.

Since it is not effective to apply conventional image mosaicking techniques directly to urban visualization, this paper proposes the 3D image mosaicking technique that creates an image mosaic in 3D space using our proposed multiple planes. Section 3 of this paper describes the camera's exterior parameters and the automatically selected matched feature points, while section 4 explains the algorithm for our proposed multiple planes. Experimental results are shown in section 5, and the conclusion is presented in section 6.

Ⅲ. Exterior parameters and feature point tracking

In order to recover the 3D coordinate of matched feature points, which is used to estimate a plane approximation of the scene geometry for each frame, we need to know the exterior and interior parameters of the camera. In this paper, we discuss the exterior parameters and assume that the interior parameters have already been established. If P(X,Y,Z) denote the Cartesian coordinates of a scene point with respect to the camera (see Fig. 2), and if (x,y) denote the corresponding coordinates in the image plane, the image plane is located at the focal length f from the focal point $o(X_L, Y_L, Z_L)$ of the camera.

The perspective projection of a scene point P(X,Y,Z) on the image plane at a point where p=(x,y)

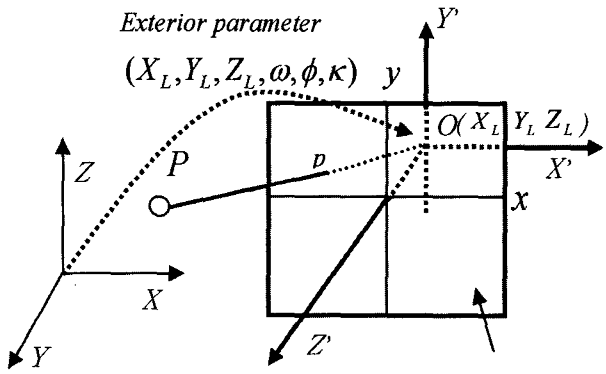


그림 2. 카메라와 영상 프레임의 외부파라미터.

Fig. 2. Exterior parameters of a camera and an image frame.

is expressed as follows:

$$\begin{bmatrix} x \\ y \\ f \end{bmatrix} = \lambda m \begin{bmatrix} X - X_L \\ Y - Y_L \\ Z - Z_L \end{bmatrix}, \quad (1)$$

where λ is the scale factor, f is the focal point of a camera, (X_L, Y_L, Z_L) is the camera position, and m is a 3×3 rotation matrix. This matrix is defined as follows:

$$m = \begin{bmatrix} C_\phi C_\kappa & C_\omega S_\kappa + S_\omega S_\phi C_\kappa & S_\omega S_\kappa - C_\omega S_\phi C_\kappa \\ -C_\phi S_\kappa & C_\omega C_\kappa - S_\omega S_\phi S_\kappa & S_\omega C_\kappa + C_\omega S_\phi S_\kappa \\ S_\phi & -S_\omega C_\phi & C_\omega C_\phi \end{bmatrix}, \quad (2)$$

where $C_w = \cos(w)$, $C_\phi = \cos(\phi)$, $C_\kappa = \cos(\kappa)$, $S_w = \sin(w)$, $S_\phi = \sin(\phi)$, and $S_\kappa = \sin(\kappa)$ and (w, ϕ, κ) is the camera pose. To eliminate the scale factor λ, we divided the first and second component equations by the third component equation in Eq.1, leading to the following more familiar collinearity equations:

$$x = f\frac{m_{11}(X - X_L) + m_{12}(Y - Y_L) + m_{13}(Z - Z_L)}{m_{31}(X - X_L) + m_{32}(Y - Y_L) + m_{33}(Z - Z_L)}$$
$$y = f\frac{m_{21}(X - X_L) + m_{22}(Y - Y_L) + m_{23}(Z - Z_L)}{m_{31}(X - X_L) + m_{32}(Y - Y_L) + m_{33}(Z - Z_L)} \quad (3)$$

Estimations of the exterior parameters are necessary in the photo-triangulation procedure since we must solve a non-linear problem, based on the collinearity conditions, to obtain a rigorous solution. An approximate solution can be achieved with a closed form space resection[16], or with a classical non-linear space resection based on collinearity conditions given more than four feature points. In this research, we used the classical non-linear space resection to obtain the exterior parameters. In cases where the frame number of the sequence is huge, the size of the designed mathematical matrix relating to the bundle block adjustment by independent models will also be huge, since a bundle block adjustment using GCP and matched feature point solves the exterior parameters of all frames in the sequence one at a time[17]. Personal computers have limited capabilities here, so we use the triplets algorithm. Assuming that the first frame is an arbitrary reference coordinate system, the exterior parameters of the second frame are first calculated by using a dependent relative orientation based on the collinearity equations. Then, the exterior parameters of each frame are calculated by using absolute orientation with the 3D coordinates of the feature points. Because the 3D coordinate of the feature points is computed by collinearity equations that use the two previous exterior parameters, the feature points must be tracked for three frames.

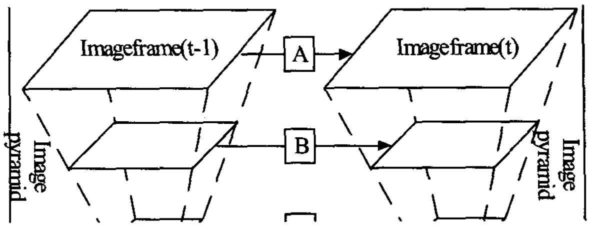
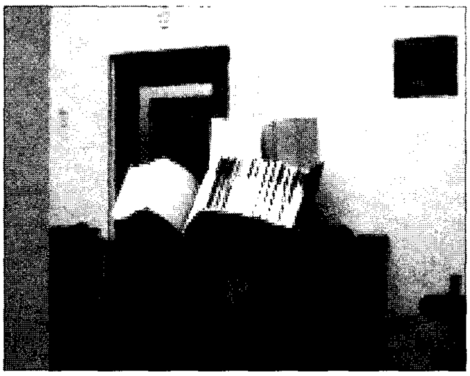


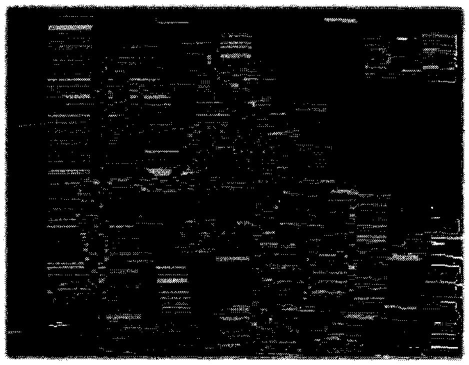
그림 3. 계층적 방법을 사용한 특징 점의 고속 정합. t 는 영상 프레임 수.
Fig. 3. Fast matching of feature points using a hierarchical strategy, where t is the number of the image frame.

Briefly stated, a contour matching algorithm reliably tracks several feature points for three frames. A set of chain feature points represents a contour. To reduce the computational cost of contour matching, a hierarchical strategy based on an image pyramid is generally adopted in the computer vision field. We also adopted the hierarchical strategy, as shown in Fig. 3. The C algorithm of Fig. 3 calculates the average shift on successive smallest frames in the pyramid. The B algorithm of Fig. 3 uses the average shift, which was calculated in the C algorithm, and also calculates the average shift on successive second smallest frames in the pyramid. The A algorithm of Fig. 3 searches for the deviation of similar contours around the average shift that was calculated in the B algorithm on successive original frames. The average shift is an average vector of matched feature points obtained from sparsely located

(a) Left image



(b) Right image



(c) Contour tracking

(d) Feature points selected on tracked contours

그림 4. 최적 정합 특징 점의 선정.
Fig. 4. Selecting the best-matched feature points.

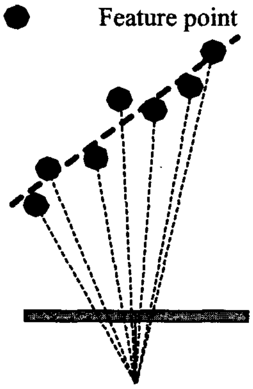
feature points extracted with the aid of an operator called SUSAN. We determined the corresponding feature points through information on both the strength similarity of the operator and the cross-correlation of the feature points. The contour matching algorithm is based on geometric constraints. In other words, the epipolar constraint between two sets of contours is used and the corresponding feature points on the contours around the average shift are computed (see Fig. 4(c)) (Han and Park, 2000). The contours are extracted by a canny edge operator (see Fig. 4(d)).

Since the well-distributed feature points are good for reducing errors of the exterior parameters, we used a method of selecting the best-matched feature points in a image, described as the tracked contours, and separated into 5 x 5 block-like the pattern on a chessboard. Then a feature point that includes the best matching rate in each block is chosen(see Fig.4(d)).

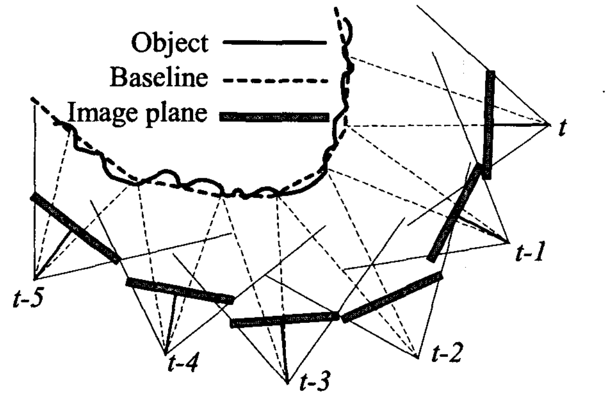
IV. Multiple planes for creating image mosaics

In this section, the design of multiple planes for creating image mosaics is introduced. The multiple planes are composed of independent planes approximated to the geometry of urban objects and the planes for panoramic mosaics in infinite areas. The independent plane, on which each frame is back-projected as texture, is an equation of the first order calculated by the classical least square using the 3D coordinate of feature points, as shown in Fig. 5(a).

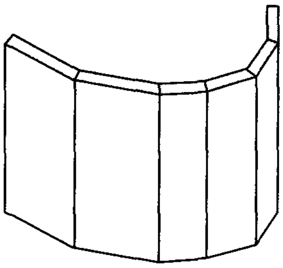
We assume that the rotation in the X′ axis of the camera in each frame is very small compared with other motions of the camera. In other words, the direction of the Y′ axis of the camera is similar to the height direction of urban objects. This first order equation is therefore only concerned with the XZ plane coordinate. Assume that the outer covering of urban objects and the position of a side-looking and moving video camera are built like the thick curve and rectangles shown in Fig. 5(b). In this case, the



(a) Independent plane



(b) Multiple planes in 2D space



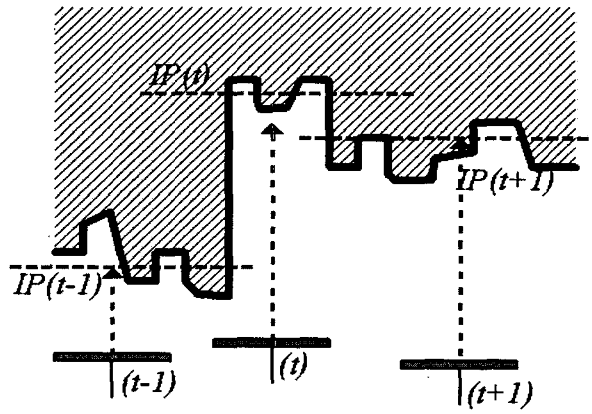
(c) Multiple planes in 3D space

그림 5. 각 프레임의 특징 점들로 구성된 독립 투영면.
Fig. 5. Independent planes formed with the feature points of each frame.

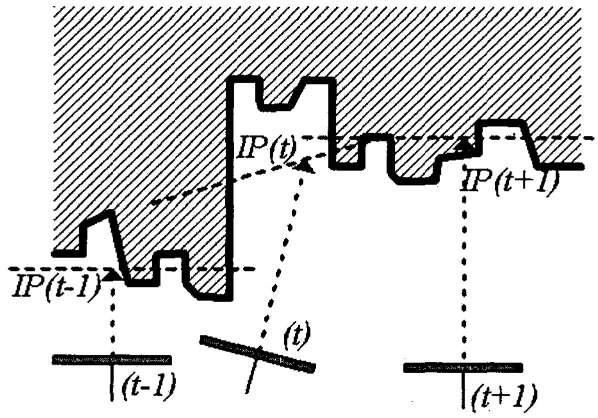
multiple planes will be designed as a dotted curve in Fig. 5(b) and will be presented as linked planes in 3D space, as in Fig. 5(c).

1. The modification of the fault independent plane

The multiple planes in Fig. 5(b) exemplify the ideal state. In most in instances, however, difficulties arise. Problems occur, for example, when some independent



(a)



(b)

그림 6. 평행한 독립 평면의 문제점, 즉, 독립평면 간의 엇 갈린 교차(IP).
Fig. 6. Problems of parallel independent planes, the fault intersection among independent planes (IP).

planes are parallel to the neighboring independent planes and generate a fault intersection, as shown in Fig. 6(a) and (b). We therefore need to modify the independent plane, as shown in Fig. 6.

We propose three modifications for the fault intersections of the planes. First, we evaluate all independent planes of each frame, $Z = a_t X + b_t$, and then detect the cases of Fig. 6(a) and (b). We then modify the pose and the position of the independent planes, as shown in Fig. 7. In the latter step, we must check whether or not the independent plane intersects the center of the previous independent from IP(t-1) to $T_{X,t-1}/2$ and from $-T_{X,t+1}/2$ to the center of the next independent

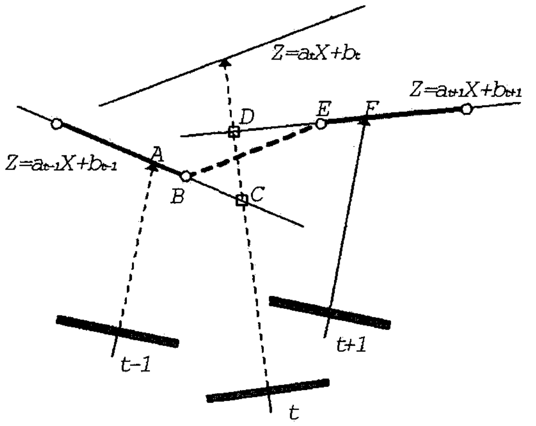


그림 7. 잘못된 교차 면 중 수정된 독립 면.
Fig. 7. Modified independent plane for a fault intersection among the independent planes.

IP(t+1). The T_X refers to image motion in the X′ axis of the frame. If we detect a fault intersection, the independent plane is modified as the thick dotted line in Fig. 7(a) that passes the point

$$B = A + 2(A - C)/5 \text{ and}$$
$$E = F + 2(F - D)/5$$

The coordinate of points A, C, D and F is as follows:

$$\begin{bmatrix} X \\ Z \end{bmatrix}_A = \begin{bmatrix} \{X_{L,t-1} + (b_{t-1} - Z_{L,t-1})\frac{m_{31,t-1}}{m_{33,t-1}}\}/\{1 - a_{t-1}\frac{m_{31,t-1}}{m_{33,t-1}}\} \\ a_{t-1}X_A + b_{t-1} \end{bmatrix}$$

$$\begin{bmatrix} X \\ Z \end{bmatrix}_C = \begin{bmatrix} \{X_{L,t} + (b_{t-1} - Z_{L,t})\frac{m_{31,t}}{m_{33,t}}\}/\{1 - a_{t-1}\frac{m_{31,t}}{m_{33,t}}\} \\ a_{t-1}X_C + b_{t-1} \end{bmatrix}$$

$$\begin{bmatrix} X \\ Z \end{bmatrix}_D = \begin{bmatrix} \{X_{L,t} + (b_{t+1} - Z_{L,t})\frac{m_{31,t}}{m_{33,t}}\}/\{1 - a_{t+1}\frac{m_{31,t}}{m_{33,t}}\} \\ a_{t+1}X_D + b_{t+1} \end{bmatrix}$$

$$\begin{bmatrix} X \\ Z \end{bmatrix}_F = \begin{bmatrix} \{X_{L,t+1} + (b_{t+1} - Z_{L,t+1})\frac{m_{31,t+1}}{m_{33,t+1}}\}/\{1 - a_{t+1}\frac{m_{31,t+1}}{m_{33,t+1}}\} \\ a_{t+1}X_F + b_{t+1} \end{bmatrix} \quad (4)$$

The modified independent plane selected the line passing points B and E in order to create multiple planes with a smooth surface. An image mosaic will be obtained by linking the modified independent planes and general independent planes, as shown in Fig. 8.

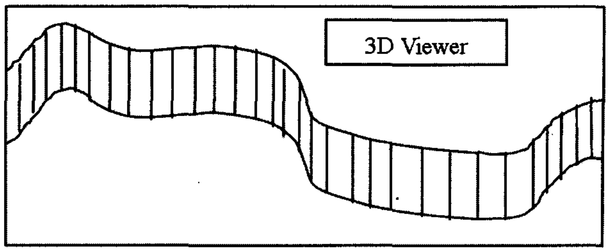


그림 8. 제안한 방법에 의한 개념적 결과
Fig. 8. Conceptual result by the proposed method.

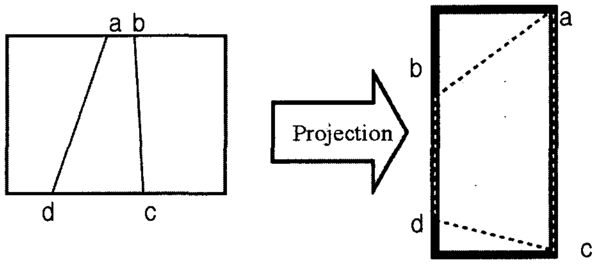


그림 9. 프레임과 독립 면간의 역 투영관계.
Fig. 9. Back-projection relationship between a frame and a independent planes.

2. Image projection on the multiple planes

Equation 3 using given the camera's exterior parameters and the multiple planes can be the equation for back-projection to the multiple planes. In Fig. 9, the X and Z parameters of a, b, c, and dare calculated by using Eq. 4. The Y parameter of a, b, c, and d are calculated as follows:

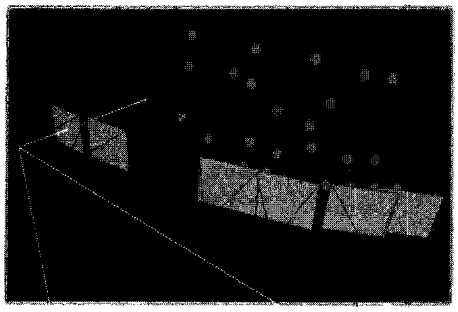
$$Y = \frac{(fm_{21} - ym_{31})(X - X_L) + ym_{32} - fm_{22})Y_L + (fm_{23} - ym_{33})(Z - Z_L)}{ym_{32} - fm_{22}} \quad (5)$$

where y is the upper or lower boundary of the image plane. The x parameter of a, b, c, and d in the frame is calculated as follows:

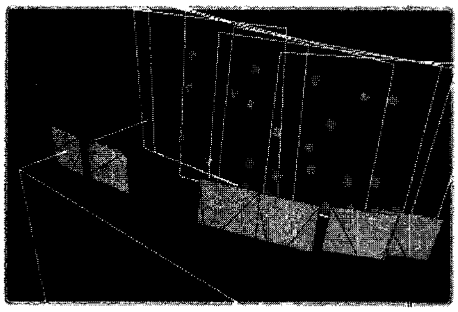
$$x = f\frac{m_{11}(X - X_L) + m_{12}(Y - Y_L) + m_{13}(Z - Z_L)}{m_{31}(X = X_L) + m_{32}(Y - Y_L) + m_{33}(Z - Z_L)} \quad (6)$$

V. Experimental results

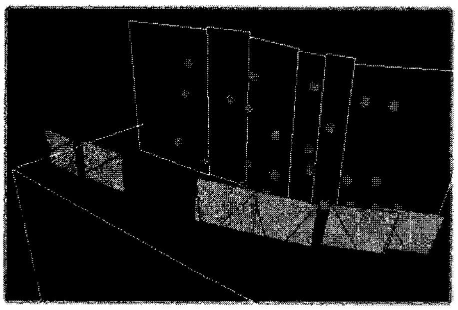
To realize the proposed method and display the results in 3D space, we used Visual C++ 6.0 and OpenGL software. We applied the proposed method to a sequence of real images taken from a moving video



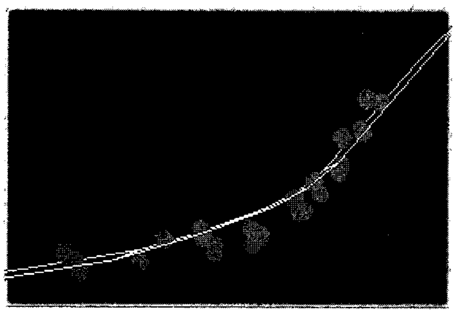
(a)



(b)



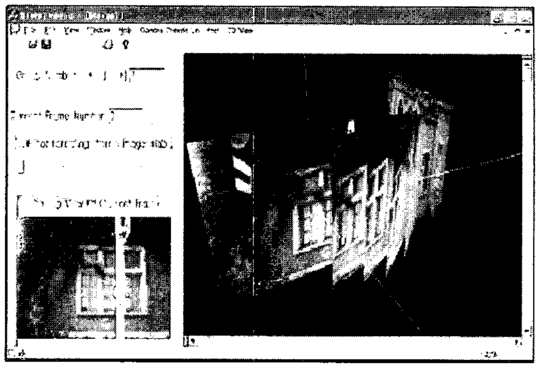
(c)



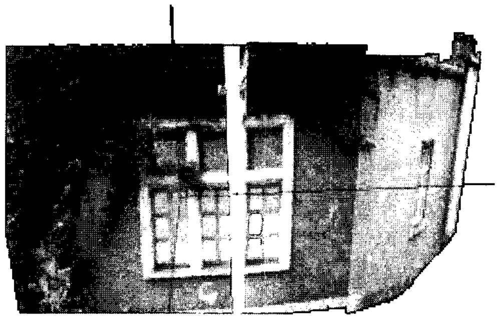
(d)

그림 10. 3D 복수 평면들; 원은 특징 점 위치를 나타내고, 사각형은 영상 프레임을 의미

Fig. 10. The 3D multiple planes: the circle is the feature points; the square is the frame.

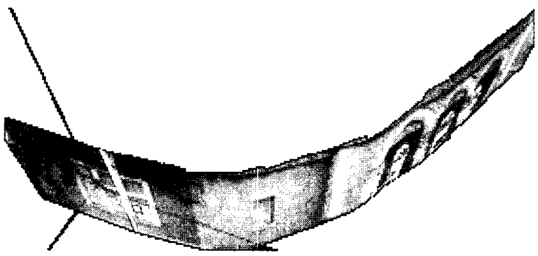


(a) A software developed by our research



(b)Left side of a buildi

(c) Front side of a building



(d) View with a down ward angle

그림 11. 복수 평면들에 영상의 투영

Fig. 11. Back-projection of the frames to the multiple planes.

camera. Fig. 10(a) shows the relationship of a moving camera and the feature points, while Fig. 10(b) shows the frames projected onto their own independent planes. Fig. 10(c) and Fig. 10(d) show the 3D vectors of the multiple planes connected with the entire size of the frame. Fig. 11(a) shows the projection of the frames to the multiple planes. Fig. 11(b) shows the left side of a house. Fig. 11(c) shows the front side of the house with results created by projecting image frames onto the multiple planes of Fig. 10(c). Fig. 11(d)shows the results when viewed from a downward angle.

V. Conclusion

A novel image mosaicking technique suitable for 3D urban visualization on websites is proposed. The proposed method enables a roadside scene to be captured by a side-looking video camera through a continuous set of textured planar faces. The proposed method adopts panoramic mosaicking techniques to reduce the ghost effect around infinite range areas, and also uses the seam-line detection algorithm to create a image mosaic. This paper demonstrates that the proposed method uses a sequence of real images to create image mosaics in 3D space effectively.

The results of this research can be applied to 3D virtual city visualization and games on websites, cell phones, and PDAs. Moreover, because the results represent continuous data, this method will overcome the drawbacks inherent in using a sequence of discrete panoramic and spherical mosaics for a visual environment with a large area.

Reference

[1] Web addresses for visualization projects in cities, http://www.casa.ucl.ac.uk.

[2] Zhao, H. and R. Shibasaki, "A vehicle-borne urban 3D acquisition system using single-row laser range scanners", IEEE Transaction onSMC Part B: Cybernetics, 33(4), 2003.

[3] Fruh, C. and A.Zakhor, "Data processing algorithms for generating textured 3D building faade meshes from laser scans and camera images", Proc. 3D Data Processing, Visualization and Transmission 2002, Padua, Italy, 2003.

[4] Pollefeys, M., R. Koch, M. Vergauwen, L. Van Gool, "Automated reconstruction of 3D scenes from sequences of images",ISPRS Journal Of Photogrammetry And Remote Sensing (55) 4, pp. 251-267, 2000.

[5] Sato, T., M.kanbara and N. yokoya, "Dense 3D reconstruction of an outdoor scene by hundreds-baseline stereo using a hand-held video camera", In IJCV,47(1/2/3), pp. 119-129, 2002.

[6] USGS Hurricane mitch program projects, http://mitchnts1.cr.usgs.gov/projects/aerial.html.

[7] Szeliski,R., "Video mosaic for virtual environment", Comput. Graph. Applicat., pp. 22-30, 1996.

[8] Chou, J.S., J.Qian, Z.Wu, H.Schramm, "Automatic mosaic and display from a sequence of peripheral angiographic images", Proc. SPIE, vol. 3034, pp. 1077-1087, Medical Imaging,1997.

[9] Standard MPEG4, Information technology-coding of audio-visual objects, ver. 1. ISO/IEC 14 496, 1999.

[10] Shum, H.-Y., R. Szeliski, "Systems and experiment paper construction of panoramic image mosaics with global and local alignment", In IJCV, 36(2), pp. 101-130, 2000.

[11] Coorg,S., S. Teller, "Spherical mosaics with quaternions and dense correlation", In IJCV, 37(3), pp. 259-273, 2000.

[12] Zhu, Z., Hanson, A.R. and Riseman, E.M., "Generalized parallel-perspective stereo mosaics from airborne video", IEEE Transactions on PAMI, 26(2), pp. 226-237, 2004.

[13] Assaf Zomet, Doron Feldman, Shmuel Peleg, and Daphna Weinshall, "Mosaicing new views: the crossed-slits projection", IEEE Transactions on PAMI, 25(6), 2003.

[14] Smith, S.M., J.M. Brady, "SUSAN - a new approach to low level image processing", In IJCV, 23(1), pp. 45-78, 1997.

[15] Han, J. H., Park, J. S., "Contour matching using epipolar geometry", IEEE Transactions on PAMI, 22(4), 2000.

[16] Zheng, Z. and Wang, X, "A general solution of a closed-form space resection",PE&RS, 58(3), pp.327-338, 1992.

[17] Edward M. Mikhail, James S. Bethel and J. Chris McGlone, "Introduction to modern photogrammetry", John wiley & sons, Inc., pp. 80-151, 2001.

저 자 소 개

전 재 춘(학생회원)
1998년 전북대학교 정밀기계공학 졸업.
2000년 전북대학교 대학원 제어계측공학과 졸업.
2002년 한국전자통신연구원 연구원.
2002년~현재 동경대학 토목공학과 (박사)

김 형 석(정회원)
1980년 한양대학교 전자공학과 졸업.
1982년 전북대학교 대학원 전자공학과 졸업.
1992년 University of Missouri, Columbia, Dept. of Electrical and Computer Eng. 박사.
1993년 9월~현재 전북대학교 정교수