基於影像集之照片重構

李坤庭
國立臺灣大學
rufuslee@cmlab.csie.ntu.edu.tw

羅聖傑
國立臺灣大學
forestking@cmlab.csie.ntu.edu.tw

陳炳宇
國立臺灣大學
robin@ntu.edu.tw

ABSTRACT

本論文提出一個由影像集來協助照片重構之系統。傳統上如果想對一張已經存在的照片，重拍出另一張相同場景、角度，甚至於建築物之立體透視角度也完全一樣的照片，所必需要的條件包括照相的角度、鏡頭種類等皆要相同。然而，這對於一位專業攝影師來說，已經是件非常困難的事情，更何況是業餘的攝影愛好者。因此，我們提供了一個讓使用者能夠較易重構照片之系統。使用者只需要在已存在之相片場景中，隨意的在該相片之相機角度附近拍幾張照片，本系統會分析重新拍攝的照片之相對關係。接著，藉由使用者給的對應資訊，本系統可以自動估算出原本相機之角度並且產生一張照相角度及立體透視角度與已存在之照片相同之照片，達到照片重構之目標。此外，若使用者無法重回現場拍照，透過使用者所收集到的網路上的照片，本系統亦可達成相同的目的。

Categories and Subject Descriptors

I.4.9 [**Image Processing and Computer Vision**]: Applications

1. INTRODUCTION

Rephotography, an act of taking a photograph from the same viewpoint of the same scene of a reference photograph, is a very useful tool while presenting the evolution of a specific location, a building, or a city. Either casual or precise rephotography provides good materials for studying history. If two photographs are stitched well, one clearer way to visualize the past-and-now image is putting one on the top of the other then adjusting the transparency of these photographs. However, it is very challenging for a photographer to find the accurate viewpoint manually because it includes six degrees of freedoms (DOFs) [1]. Therefore, precise rephotography techniques have been scientifically studied for a long time. Existing research has proposed a real-time estimation technique for rephotography that guides users to the desired viewpoint. It is useful for the users in the action of taking the photograph. However, sometimes the users may have no chance to go back to the place and take the photograph. Therefore, rephotography using existing photos or videos could be useful as a post-process.

In this paper, we present a rephotography technique that "rephotographs" with a number of photographs. It takes an older photograph as the reference, and renders an image that has the same view by a number of existing photographs. Therefore, users only need to take a number of photographs at the same scene without a carefully study of the reference image in advance. These existing photographs are combined together to construct the point clouds of the scene by using the structure from motion (SfM) technique. Then we analyze the parameters of the reference photograph and render the photograph using image-based rendering (IBR) techniques [2]. Finally, we adopt a content-aware warping technique [17] to optimize the rephotography result.

Specifically, our contributions are as follows.

- We propose a technique for rephotographing the historical photos while user-specified viewpoint is not necessary,
- A hybrid technique that combines the image-based rendering and warping for generating the accurate view.

2. RELATED WORK

Rephotography. Rephotography is a popular research topic for a long time in photography field, and has acquire more attention in computer vision recently. Given a reference photograph of a scene, the goal is to take a photograph exactly from the same viewpoint as the reference one. Bae *et al.* [1] has proposed a novel system that allows users to go back to the scene, and guides users to reach the desired viewpoint based on the reference photograph. The system is useful because even amateur users can rephotograph a scene according to the guidance of the system. However, sometimes users have no opportunity to go back to the place and take the photo. Therefore, guiding users to rephotograph can't satisfy the need of these users. Instead of providing the guidance, our system adopts image-based rendering techniques to render a novel view of the scene based on a set of existing photographs. Users do not need to reach the accurate viewpoint of the reference photograph when taking photographs. 4D cities project [20] builds a time-varying 3D models of cities from historical photographs and modern photographs. Users can transit in the 3D space when

browsing these photos.

Structure from Motion. Estimating the camera position and reconstructing the scene via multiple images is a core problem in computer vision. Structure from motion (SFM) is a popular technique [10, 9]. It is an automatic recovery of camera motion and scene structure from two or more images and a self calibration technique. SFM can be used in a wide range of applications including the reconstruction of virtual reality models from video sequences, photogrammetric survey and special effect of movie. Photo tourism [23] adopts the technique, and computes the camera parameters from a collection of photos from the web. It integrates all photographs of the same scene in to a three dimension space as well as constructing the structure of the scene.

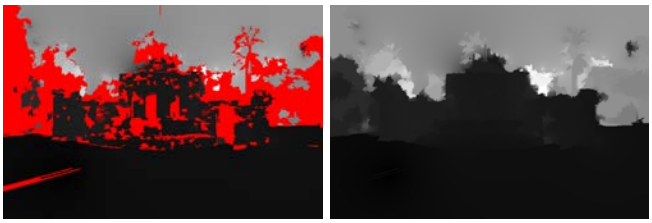
Image-Based Rendering. Image-based rendering is a technique that renders novel views directly from a set of input images [11]. According to the geometry model used, it can be classified into three categories: rendering with no geometry, rendering with implicit geometry and rendering with explicit geometry. Rendering with no geometry use no geometry information at all. Without geometry, the samping must be very dense, or the possible motion is restricted. There are many researches [22, 16, 7, 15] which belong to this category. The second category, rendering with implicit geometry, relies on positional correspondences across a small number of images to render new views. The term *implicit* implies that the geometry is not directly available [2, 21, 14]. The third category is rendering with explicit geometry, which has direct 3D information. It has the representation of 3D coordinates or depth along a sight line. The previous works [6, 5] render novel view using view-dependent texture maps. [8] stores several depth layers for every pixel to acquire the depth information. [13] computes the depth map by merging multi-view stereo algorithm and segmentation. With an iterative correction process, they can obtain the rough depth map and render the novel view. The rendering result with the explicit geometry is more reliable and precise; therefore, we render the novel view based on the explicit geometry method. In our work, we compute the depth map with three dimension space projection, which will be introduced in later section.

3. SYSTEM OVERVIEW

Figure 1 summarizes the whole process of our system. Given a set of photographs taken from the same scene, the system first analyzes the camera parameters as well as constructs the 3D point clouds of the scene. It is achieved by performing the structure from motion (SFM) technique [23] over these photographs (Section 4). Users need to determine a set of corresponding features between the historical reference photograph and the 3D point clouds. Then the camera parameters of the historical photograph are estimated. The depth map of each photograph is also computed for rendering the final novel view. The system then renders a novel view which matches the viewpoint of the historical photograph by projecting these 3D point clouds onto the historical camera. The depth map of the novel view is computed simultaneously, and we project the pixels of the novel view back to each input photographs to refine the result (Section 5). Finally, we perform inpainting [4] and content-preserving warping [17] to refine the viewpoint shifting caused by the noise (Section 6).

(a) (b)

Figure 2: (a) Original photo. (b) Segmentation result of (a).



(a) (b)

Figure 3: (a) Depth map without using neighbor's features. (b) Depth map using neighbor's features

4. SCENE RECONSTRUCTION

The first task is to reconstruct the scene of the input photographs set as point clouds. In this section, we introduce the technique of scene reconstruction in more detail. There are two steps: camera calibration and 3D point cloud construction, depth map reconstruction.

4.1 Camera Calibration and 3D Point Clouds Construction

To reconstruct the 3D structure of the scene covered by the set of photographs, we estimate the intrinsic and extrinsic matrix of each camera, and generate the 3D point clouds of the scene using the structure from motion technique [23]. We extract the focal length, radial distortion coefficient, rotation matrix, transformation matrix of each camera, and also a set of 3D points with color information.

4.2 Segmentation-based Depth Map Reconstruction

In this step, we perform segmentation for each input photograph using [3, 19]. The result is shown in Figure 2(b). We assume the depth values in the same segment is smooth. Next, we project the 3D point clouds onto each input photograph.There are several projected points in each segment. We assign the depth values of projected points with the z-axis distance under the camera coordinate. Specifically, we

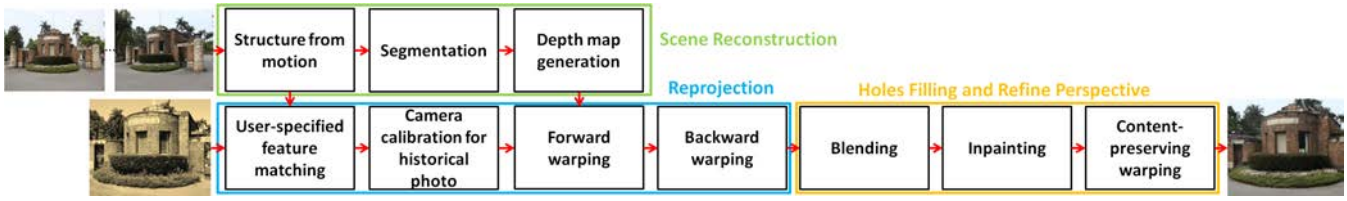


Figure 1: The block diagram of our system overview.

estimate the depth value of each pixel p_i as

$$D(p_i) = \frac{1}{W_i} \sum_{f_j \in S_k} \frac{d(f_j)}{||f_j - p_i||} \quad (1)$$

$$W_i = \sum_{f_j \in S_k} \frac{1}{||f_j - p_i||} \quad (2)$$

where f_j is the projected point of the j_th point in the point clouds, S_k is k_th segmentation, and $d(\cdot)$ is the depth value.

The depth map generated by the above equation is shown in Figure 3(a). There are some pixels that have no depth value (red colored area) because no 3D points are projected on these segments. To solve this problem, we use the projected points of their neighbor segments as its projected points, and compute the depth values as the same way. However, the depth values of a segment may be wrong because of projection errors. Therefore, we compute the color difference between projected points and average color of the segment. If the color difference between a segment and a projected point satisfy

$$\|c_p - c_s\| < \sigma, \quad (3)$$

we consider the depth value of projected point is credible. Figure 3(b) shows the result.

5. REPROJECTION

In this section, we describe the details of the historical camera parameters estimation and novel view generation.

5.1 Camera Calibration for Historical Photo

The method we use to calibrate the reference photograph is similar to [1]. They estimate the extrinsic and intrinsic matrix by minimize the sum of squared projection error of the matched points between the reference photo and existing photo. The matched features in their work are chose from the SIFT correspondences[18]. There are potential problems if the reference photo is noisy. Figure 4 shows a failure case of SIFT correspondence. Too many feature correspondences are wrong due to the differences of photograph qualities, noise, and perspective. Therefore, we let user directly find the point correspondences between the projected points of one existing photo and the reference photo. We apply the focal length of the camera which chose to match the correspondences as our initial guess for Levenberg-Marquardt algorithm, and the rest initial guesses are set as [1].

5.2 Forward warping

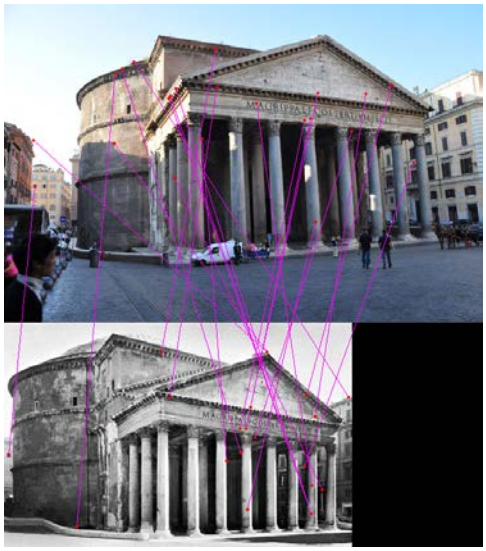


Figure 4: The SIFT correspondence between the reference and modern photo of "Patheon". Correct correspondences are rare.

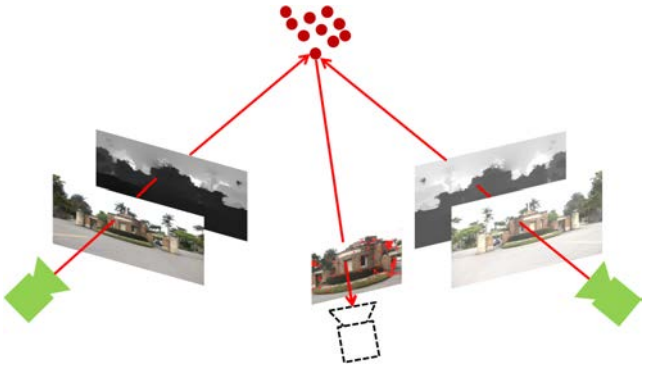


Figure 5: Projecting every pixel of the existing photos into the 3D space, and projecting them onto the novel view camera.

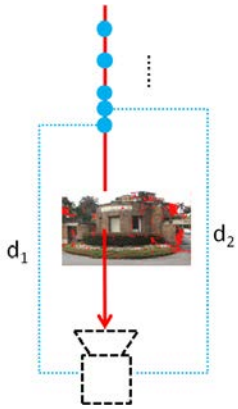


Figure 6: There are lots of points projected onto a single pixel.

We have all the camera parameters of each existing photo and its depth map, relatively. We reproject every pixel on the photo to 3D space. Next, we project all the 3D points onto the reference viewpoint in terms of the reference camera pose we estimate at the previous step (Figure 5). However, there may be several points projected on a single pixel. We assign different weight to each projected point based on the distance between the point and reference camera (Figure 6). The weight function is designed as

$$C(p_i) = \frac{1}{W_i} \sum_{P_j \in proj(p_i)} C(P_j) \frac{1}{D(P_j)} \tag{4}$$

$$W_i = \sum_{P_j \in proj(p_i)} \frac{1}{D(P_j)} \tag{5}$$

p_i is the i_th pixel on the reference view plane. $proj(p_i)$ is the set of points projected onto p_i. P_j is j_th 3D point. $D(\cdot)$ is the projected distance. $C(\cdot)$ is the color information. Figure 7(a) is the result of forward warping step. Besides, we can obtain a corresponding depth map of reference photo. We compute the depth map using the following equation

$$d(p_i) = \min \{D(P_j) | P_j \in proj(p_i)\} \tag{6}$$

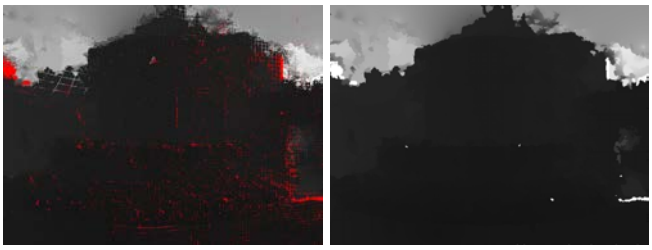
$d(\cdot)$ is the depth value, $D(\cdot)$ is the projected distance. The equation means that we set the depth value by the distance of the closest projected point. Figure 8(a) is the result depth map of reference photo. As what we see, there are a large number of noises on the depth map. We remove those noises by performing median filter. Figure 8(b) is the result of removing noises. In addition, because the points projected on the sky or ground region are rare even none, the depth value of sky or ground may be wrong. Therefore, we perform sky detection in [12] before computing the depth map of the reference photo. We first project non-sky and non-ground region to the reference viewpoint then project the rest. Figure 9 shows the comparison.

5.3 Backward warping

With the depth map and the camera pose of the reference photo, we reproject every pixel on the reference viewpoint to the 3D space and project onto existing photos(Figure 10). We acquire the color informations on the existing photos.

(a) (b)

Figure 7: (a)The result of forward warping. (b)The result of backward warping.



(a) (b)

Figure 8: (a)Depth map of the novel view before smoothing. (b)Depth map of the novel view after smoothing.

Then, we average the pixel colors and fill in the backward image. Figure 7(b) shows the result of backward warping.

6. HOLES FILLING AND REFINE PERSPECTIVE

In this section, we will describe the methods which combine the forward and backward warping photos of reference viewpoint. But the result may have holes, we also introduce the refined algorithm including inapinging [4] and content-aware warping [17].

6.1 Blending

Since we have the forward and backward warping photo, we blend these two photos together. The criteria of blending is that the holes of backward warping image are filled with forward warping image. Figure 11(a) shows the result.

(a) (b)

Figure 9: (a)Novel view generation without sky detection. (b)Novel view generation with sky detection.

(a) (b) (c) (d) (e)

Figure 11: (a)The result of blending the forward and backward warping photos. (b)The result after perform inpainging. (c)The result after perform content-preserving warping. Keeping an eye on the area of yellow rectangles of (b) and (c). The perspective of the wall is altered obviously. (d)The grids before perform content-preserving warping. (e)The grids after perform content-preserving warping

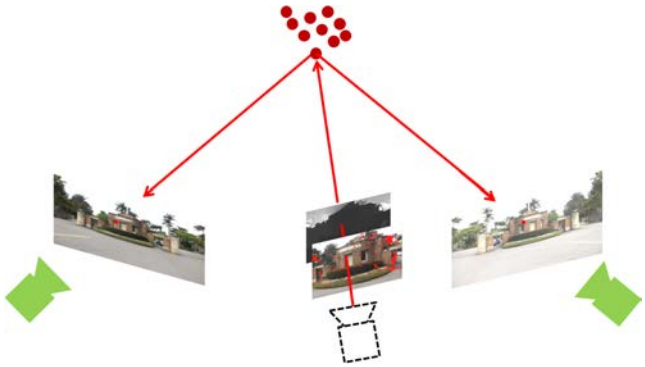


Figure 10: Reprojecting every pixel of the novel view to the 3D space and projecting onto each camera viewpoint of existing photo. Acquiring the color informations of existing photos.

	NTU	Chien Kuo	Patheon	St. Paul
# of photos	16	14	11	44
Resolution	855×570	535×356	1024×768	700×567
Depth map(sec)	5.558	14.72	33.18	28.62
Ref. resolution	600×450	400×265	465×300	400×283
Novel view(sec)	10.93	5.38	7.78	5.34

Table 1: Information of the testing data and the execution time

6.2 Image Inpainting

Although we have the forward and backward warping image, there are probably holes in the result image(the red areas in Figure 11(a)) because of occlusion, projection error or lack of informations of the scene structure. We fill holes using [4]. Figure 11(b) shows the result.

6.3 Content-preserving Warping

If we shoot at the same viewpoint with different cameras, the photos will look different in perspective from each other. Even the reference camera pose we estimate is accurate, the rendering result may be different from the reference photo (yellow area of Figure 11(b)). We use content-preserving warping[17] to refine the result image(Figure 11(c)).

7. RESULT

We implement our work in an environment of dual-core 2.1 GHz laptop, and take photographs with canon 500D. In our experiment, we set the threshold of maximum color difference σ to 15. Table 1 shows the information of our testing data and the execution time. It includes the number of photos, resolution of existing photo, depth map generation time, resolution of reference photo and novel view generation time for each scene. Figure 12 (a),(b) show the result of "National Taiwan University" photo sequence. The viewpoint we estimate is close to the actual viewpoint of the reference photo. Figure 12 (c),(d) show another result of "Ruins of St.Paul", which is collected from the internet. In this case, the correctness of the viewpoint we estimate is good. Because of diverged angles of the viewpoints of the photographs on the internet, we may not retrieve all the information in the scene of the reference photo. If the photo sequence lacks too much information of the scene in the reference photo, the holes would be very large, e.g. Figure 12 (d) and unable to be restored by inpainting. Finally, Figure 12 (e),(f) show the result of "Chien Kuo High School" photo sequence.

8. CONCLUSION AND FUTURE WORK

In this paper, we have proposed a technique which automatically rephotographs by user-specified feature matching between the set of existing photographs and the reference photograph. Our contributions are summarized as follows. First, according to the 3D point cloud and estimating the camera parameters constructed by SFM, we compute the depth maps for each photo. Second, we estimate the camera pose of the reference photo by the result of SFM and user-specified correspondences. The results show that it is very close to the actual camera viewpoint. Third, we reduce the artifacts produced by the wrong depth value of the sky and ground using sky detection.

In the future work, we will first improve the precision of the depth map because it is one of the critical parts of our work. Second, since the camera is definitely different from the historical one, the perspective in the historical photo is different from now. Therefore, solving the problem of the inconsistence of the perspective is important. We believe our work will benefit the computer vision, image-based rendering and computational photography.

9. REFERENCES

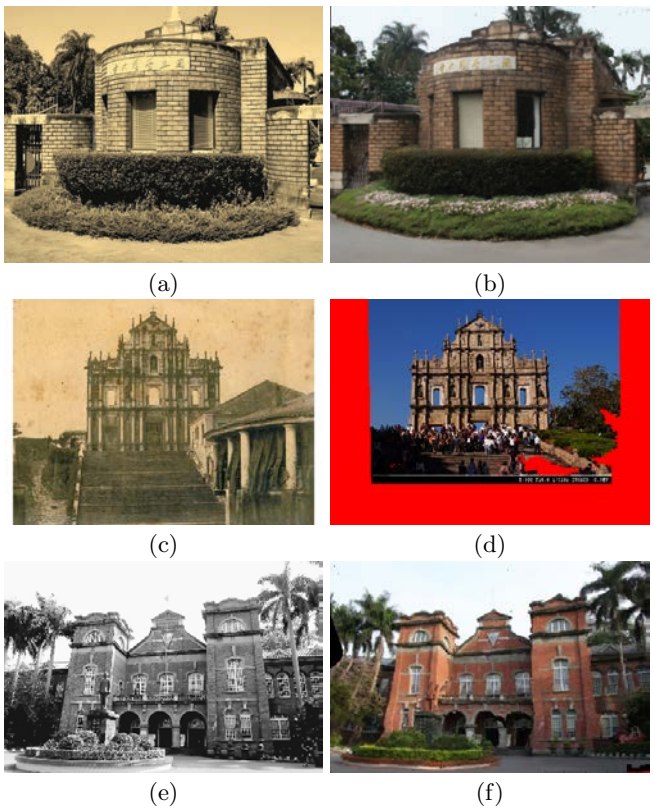


Figure 12: (a)Reference photo of "NTU". (b)Rephotograph of "NTU". (c)Reference photo of "Ruins of St.Paul". (d)Rephotograph of "Ruins of St.Paul". (e)Reference photo of "Chien Kuo High School". (f)Rephotograph of "Chien Kuo High School".

[1] S. Bae, A. Agarwala, and F. Durand. Computational rephotography. *ACM Trans. Graph.*, 29:24:1–24:15, July 2010.

[2] S. E. Chen. View interpolation for image synthesis, 1993.

[3] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24:603–619, May 2002.

[4] A. Criminisi, P. Perez, and K. Toyama. Object removal by exemplar-based inpainting. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 2:721, 2003.

[5] P. Debevec, C. Taylor, and J. Malik. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. pages 11–20, 1996.

[6] P. Debevec, Y. Yu, and G. Borshukov. Efficient view-dependent image-based rendering with projective texture-mapping. pages 105–116.

[7] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen. The lumigraph. pages 43–54, 2001.

[8] S. J. Gortler, L. wei He, and M. F. Cohen. Layered depth images, 1997.

[9] R. I. Hartley. Estimation of relative camera positions for uncalibrated cameras. In *Proceedings of the Second European Conference on Computer Vision*, ECCV '92, pages 579–587, London, UK, 1992. Springer-Verlag.

[10] R. I. Hartley. In defense of the eight-point algorithm. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19:580–593, June 1997.

[11] S. B. K. Heung-Yeung Shum. A review of image-based rendering techniques, 2000.

[12] D. Hoiem, A. A. Efros, and M. Hebert. Automatic photo pop-up. *ACM Trans. Graph.*, 24:577–584, July 2005.

[13] A. A. M. A. D. S. Ke Colin Zheng, Alex Colburn. A consistent segmentation approach to image-based rendering, 2009.

[14] S. Laveau, O. Faugeras, O. Faugeras, P. Robotique, and P. Robotvis. 3-d scene representation as a collection of images and fundamental matrices. Technical report, 1994.

[15] M. Levoy and P. Hanrahan. Light field rendering. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, SIGGRAPH '96, pages 31–42, New York, NY, USA, 1996. ACM.

[16] A. Lippman. Movie-maps: An application of the optical videodisc to computer graphics. *SIGGRAPH Comput. Graph.*, 14:32–42, July 1980.

[17] F. Liu, M. Gleicher, H. Jin, and A. Agarwala. Content-preserving warps for 3d video stabilization. In *ACM SIGGRAPH 2009 papers*, SIGGRAPH '09, pages 44:1–44:9, New York, NY, USA, 2009. ACM.

[18] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60:91–110, November 2004.

[19] P. Meer and B. Georgescu. Edge detection with embedded confidence. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23:1351–1365, December 2001.

[20] G. Schindler and F. Dellaert. Inferring temporal order

of images from 3d structure. In *In Int. Conf. on Computer Vision and Pattern Recognition*, 2007.

[21] S. M. Seitz and C. R. Dyer. View morphing. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, SIGGRAPH '96, pages 21–30, New York, NY, USA, 1996. ACM.

[22] H.-Y. Shum and L.-W. He. Rendering with concentric mosaics, 1999.

[23] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM SIGGRAPH 2006 Papers*, SIGGRAPH '06, pages 835–846, New York, NY, USA, 2006. ACM.