Photo Traveler:

A System for Exploring Photos in 3D

Shuang He[1] Yue Qi[2] Fei Hou[3]

State Key Lab of Virtual Reality Technology and Systems, Beihang University, China
Email: {[1]heshuang; [2]qy; [3]houfei}@vrlab.buaa.edu.cn

Abstract

This paper presents a system - Photo Traveler - for exploring large photo collections of a scene with a visceral 3D sense. Based on 3d-reconstruction, Photo Traveler managed to rearrange those photos in 3D scene, enabling the user to explore photos as if looking through the real cameras, and move between photos as if strolling the scene.

Photo Traveler lays emphasis on high-density reconstruction of scene geometry at low cost of time. Towards this goal, a novel approach is proposed to obtaining reliable relationship between two images.

Keywords: photo exploring, image-based modeling, quasi-dense matching, structure from motion, Internet imagery

1. Introduction

Digital photography together with Internet enables the largest sharing database of image, and provides a great opportunity for innovational applications. Snavely et al. [1] presented a 3D photo browsing system, which was the predecessor of Microsoft's Photosynth. Vergauwen et al. [2] developed a web-based reconstruction service for cultural heritage applications. A new branch of computer vision on Internet imagery is also developing [3]. Photo Traveler is mostly inspired by Photosynth but devoted to practical problems discussed as follows:

- **Unordered photos:** an image sequence guarantees small baseline in successive frames and large baseline in distant frames, which are unknown to an unordered set. So we believe reliable estimation of pairwise motion is important, and introduce a new method consisting of improved normalization and robust estimation with feedback.
- **High-density scene geometry:** last two decades, the standard sparse structure from motion (SfM) approaches maturely developed [4, 5], but barely enough to representing the scene. On the other hand, dense matching is generally ill posed, and sometimes unnecessary in terms of time efficiency [6]. To overcome the insufficient sparse matching and fragile dense matching, we impose quasi-dense approach [7] to densify matching points.
- **Low cost of time:** as referred in [3], Photosynth spent days to handle a thousand photos. However, Photo Traveler saved a lot time by selective match propagation and batch recovery of cameras in SfM.

The paper is organized as follows: Section 2 presents our approach to quick reconstruction of cameras and quasi-dense geometry of the scene. Section 3 shows the exploring interface and techniques. Results and conclusion are given in Section 4.

2. Reconstruction of Camera Positions and Quasi-Dense Geometry

Starting from a collection of unordered photos of a scene, we are aiming at reconstructing quasi-dense geometry of the scene, and camera position of each photo. Towards this goal, our algorithm consists of five steps, as depicted in Fig.1. Each of these steps is described in the subsections.

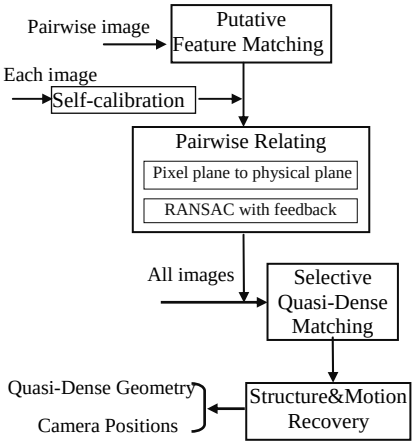


Fig. 1: Flow chart of 3D reconstruction of camera positions and quasi-dense geometry

2.1. Putative feature matching

The reconstruction starts with detecting features in each image using the SIFT detector introduced by D. Lowe [8]. Since each image could potentially match every other one in unordered datasets, we search matches between every two images. Once features have been extracted, we perform feature matching between each image pair ($\boldsymbol{I}_1$, $\boldsymbol{I}_2$), using the approximate nearest neighbours (ANN) *kd*-tree package of Arya et al. [9]. The specific algorithm is described in Table 1.

Table 1: Putative feature matching algorithm

Input: each image pair ($\boldsymbol{I}_1$, $\boldsymbol{I}_2$)
set rejection ratio rt_1 , rt_2
for each feature $ft \in \boldsymbol{I}_1$:
find 5 NN ft_1-ft_5 in $\boldsymbol{I}_2$ with distance d_1-d_5
set $e_{out}=d_5$ (consider ft_5 as outlier)
for $ft_{i=1...4}$:
if ($d_i/d_{out}<rt_1$)
find 2 NN ft_{i1}, ft_{i2} in $\boldsymbol{I}_1$ with distance d_{i1}, d_{i2}
if ($d_{i1}/d_{i2} < rt_2$ & $ft_{i1}= ft$)
accept < ft, ft_i >
Output: putative matches

2.2. Approximate self-calibration

Different from standard reconstruction approach, where camera is calibrated after projective reconstruction, we estimate an approximate intrinsic camera matrix in this step, taking advantage of the Exchangeable Image File Format (EXIF) metadata in each digital photo. Therefore, the approximate calibration result could be used in the normalization of image coordinates.

The intrinsic parameters of camera can be represented by a matrix:

$$K = \begin{bmatrix} f/dx & s & u_0 \\ 0 & f/dy & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

Where (u_0, v_0) is the principle point, s is the distortion factor, f represent focal length in physical measurement, (such as *mm*), dx, dy implicate the quantification scale that is one *pixel* represents a rectangle region with width of dx and height of dy in physical measurement.

Present imaging systems could perform very close to the ideal pinhole camera model. It is reasonable to ignore the distortion factor s and regard the pixel center as the principle point. On the other hand, the EXIF data in image file records much information about the camera setting,

such as the focal length *a(mm)*, the size of image *(W_{img} , H_{img})*, the camera maker, and model type. Given the camera maker and model type, it is easy to get the size of sensor *(W_{ccd} , H_{ccd})* according to industry standard. Thus, we can compute an approximate intrinsic camera matrix:

$$\tilde{K} = \begin{bmatrix} a\frac{W_{img}}{W_{ccd}} & 0 & \frac{W_{img}}{2} \\ 0 & a\frac{H_{img}}{H_{ccd}} & \frac{H_{img}}{2} \\ 0 & 0 & 1 \end{bmatrix} \qquad (2)$$

2.3. Pairwise relating

From pixel plane to physical plane: Hartley [10] pointed out it is important to normalize the image coordinates to eliminate the difference in order of magnitude. While recording with a digital camera, the object is first mapped to the film plane, and then re-projected to the pixel plane for digitization. For simplicity, the object is often directly projected into image plane, ignoring the differences in imaging systems, which result in the difference in order of magnitude.

Unlike standard normalization, we transform images from pixel image plane to physical film plane consistent with the measurement of focal length. The transform from physical point *(x, y)* to pixel point *(u, v)* can be deduced from Eq. (1), (2) as Eq. (3):

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} 1/dx & 0 & u_0 \\ 0 & 1/dy & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{W_{img}}{W_{ccd}} & 0 & \frac{W_{img}}{2} \\ 0 & \frac{H_{img}}{H_{ccd}} & \frac{H_{img}}{2} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \qquad (3)$$

RANSAC with feedback: Two images with non-coincident camera centers can be constrained with epipolar geometry expressed as a fundamental matrix F; two images with the same camera center can be mapped by a homographic matrix H from one to the other [11]. Therefore, before using RANSAC [12] for parameterized model estimation, we have to evaluate which of the two models - F or H - is best suited to explain the data. Different from analyzing the dataset itself [13], we present an improved RANSAC with feedback.

We first assume they comply with epipolar geometry and set ***F*** as default model, which will be tested by the result of singular value decomposition (SVD) of essential matrix. If the SVD fails, we reject the pair for not fitting 3D reconstruction. Considering the mismatch, we brought in tolerated error during positive depth constrain. The whole algorithm is described in Table 2.

Table 2: Pairwise relating

Input:
putative matches ***PUTA***{<$\boldsymbol{p}_1^i$,$\boldsymbol{p}_2^i$>; i=1,…,n}
approximate intrinsic camera matrix $\boldsymbol{K}_1$,$\boldsymbol{K}_2$

transform $p^i \in PUTA$ to film plane
set *model* (fundamental matrix ***F***)
RANSAC(model, matches ***M***);
compute essential matrix $\mathrm{E} = \mathrm{K}_2^{\mathrm{T}} \mathrm{F} \mathrm{K}_1$
projection matrix ***P****[4]*=SVD decomposition *(E)*
break_times$_1$=0, break_times$_2$=0;
for j=1,…,4
  for all $m^i\ (x^i, y^i) \in M$
    compute depth $\boldsymbol{Z}_1(P[j])$, $\boldsymbol{Z}_2(P[j])$
    if(!($\boldsymbol{Z}_1^i$>*0* & $\boldsymbol{Z}_2^i$>*0*))
      break_times$_1$++; break;
if(***break_times$_1$***= =***4***)
  set tolerated error ***T***
  for j=1,…,4
    for all $m^i\ (x^i, y^i) \in M$
      compute depth $\boldsymbol{Z}_1(P[j])$, $\boldsymbol{Z}_2(P[j])$
      if($\boldsymbol{Z}_1^i$<*0* || $\boldsymbol{Z}_2^i$<*0* || out of ***T***)
        break_times$_2$++; break;
  if(***break_times$_2$***= =***4***)
    reject all matches

Output: inliers and ***F*** , or rejection of ***PUTA***

2.4. Selective quasi-dense match

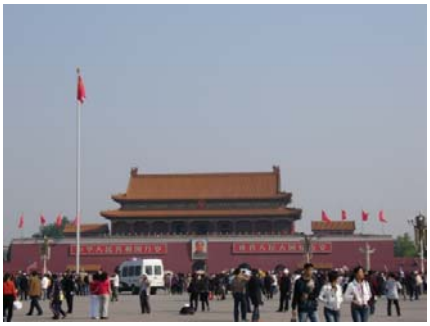
To densify the matching points, we apply the quasi-dense approach similar to that of Lhuillier and Quan [7], but we only choose the pairs with 'little' varied intrinsic camera and 'moderate wide' baseline. As we know, small baseline between two views indicates good matches but ill-posed 3d-reconstruction. The quasi-dense approach we imposed is proved effective in baseline extension for good matches. However, it only works for constant or little-varied intrinsic camera. When intrinsic camera varied a lot, it is likely to lead to the result even worse than sparse matching. Moreover, due to the time-consuming procedure, it is important to appropriately selecting the propagation pairs.

Fig.2 gives an instance of the quasi-dense matching between two views with wide-baseline and little-varied intrinsic camera. The experimental data are given in Table.3.

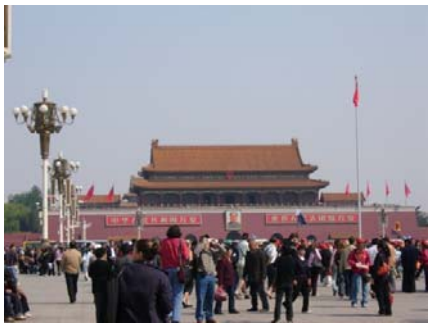
Fig.3 shows the rejection cases with too small baseline or degenerated configurations, and their corresponding experimental data can be found in Table.4.

Table.3: Selective quasi-dense matching data

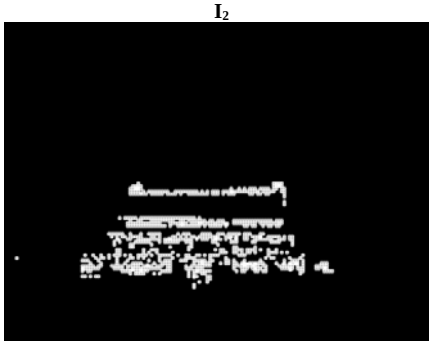
Self-calibration		Sparse match
I_1	I_2	
$\begin{bmatrix} 3228.5 & 0 & 640 \\ 0 & 3266.8 & 480 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 3835 & 0 & 640 \\ 0 & 3880.4 & 480 \\ 0 & 0 & 1 \end{bmatrix}$	73
SVD		Quasi-dense match
Rotation	Translation	
$\begin{bmatrix} 0.993 & 0.008 & -0.117 \\ -0.02 & 0.994 & -0.106 \\ 0.115 & 0.108 & 0.987 \end{bmatrix}$	$\begin{bmatrix} 0.78 \\ -0.012 \\ -0.199 \end{bmatrix}$	855



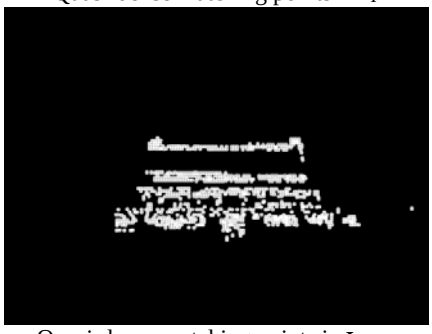
$\mathbf{I_1}$



$\mathbf{I_2}$



Quasi-dense matching points in $\mathbf{I_1}$



Quasi-dense matching points in $\mathbf{I_2}$

Fig.2: Selective quasi-dense matching with wide baseline & little-varied intrinsic camera

(a) Rejected sample with small baseline : intrinsic camera varied a little

(b) Rejected sample with small baseline : intrinsic camera varied a lot

(c) Rejected sample with wide baseline : intrinsic camera varied a lot

Fig.3: Rejected samples in selective quasi-dense matching, where (a) has good quasi-dense matching result but failed in SVD, (b) and (c) are degenerated cases when intrinsic camera varied a lot.

Table.4: Experimental data of rejected samples in selective quasi-dense matching

Fig.3		Self-calibration	Sparse match	SVD	Quasi-dense match
(a)	Left	$\begin{bmatrix} 1248.6 & 0 & 640 \\ 0 & 1263.4 & 480 \\ 0 & 0 & 1 \end{bmatrix}$	259	failure	1062
	Right	$\begin{bmatrix} 1498.3 & 0 & 640 \\ 0 & 1516.1 & 480 \\ 0 & 0 & 1 \end{bmatrix}$			
(b)	Left	$\begin{bmatrix} 2479.4 & 0 & 640 \\ 0 & 2508.7 & 480 \\ 0 & 0 & 1 \end{bmatrix}$	116	success	42 failure
	Right	$\begin{bmatrix} 4994.4 & 0 & 640 \\ 0 & 5053.6 & 480 \\ 0 & 0 & 1 \end{bmatrix}$			
(c)	Left	$\begin{bmatrix} 2943.1 & 0 & 640 \\ 0 & 2798 & 480 \\ 0 & 0 & 1 \end{bmatrix}$	44	failure	3 failure
	Right	$\begin{bmatrix} 1498.3 & 0 & 640 \\ 0 & 1516.1 & 480 \\ 0 & 0 & 1 \end{bmatrix}$			

2.5. Structure and motion recovery

After finding all matching image pairs, we connect the matches into tracks. Then we perform the SfM procedure similar to that of Snavely et al. [14], but apply batch recovery of camera in terms of efficiency.

Firstly, we select two views with both sufficient matches and large baseline for initial reconstruction. Next, we choose at most five cameras; each of them observes more reconstructed tracks than any other does. We add these cameras to the optimization set of camera at one time and compute their projection matrix. Then, we add tracks observed by the new cameras into the optimization set of 3d-point, if another recovered camera exists in the track. This procedure is repeated until no remaining camera observes any reconstructed 3d-point. The sparse bundle adjustment algorithm [15] is used each iteration to find the minimum error solution.

3. Photo Exploration in 3D

Knowing the position of each camera, we can register each photo with a common 3D coordinate frame, which gives the user a strong sense of spatial relationship. In addition, simple morphing techniques can provide smooth transitions as if strolling between photos.

3.1. Model navigation

Fig. 4 is a downside looking of the main scene of the Gate of Heavenly Peace in China represented as a quasi-dense point model. The cameras are rendered as frustums. If the user strolls towards a camera, the virtual camera will smoothly move into the photo view, while the neighbors will be rendered as semi-transparent faces.

Fig. 4: Scene model and cameras

3.2. Photo exploration

When the user visits a photo, we search its neighbors of virtual camera. As shown in Fig.5, we render the results as a visual-link graph of thumbnails by the distance of relevance, which could give a strong sense of spatial relationship. When the user moves from one photo to another, we use simply morphing techniques [1] to generate smooth transitions between relative cameras, providing a strong sense of spatial relationships. Therefore, the user can 'travel' through those photos with a visceral 3D sense.

Fig.5: Photo exploring interface

4. Results and Conclusion

We have applied our system to exploring the Gate of Heavenly Peace of China from a personal collection of 128 photographs. The reconstructing time was about 85 minutes, and 64 photos were ultimately registered. During the pairwise relating, 36 matching pairs of image are rejected by epipolar geometry constrain, 18 pairs were selected for quasi-dense matching. The recovered camera positions were proved trustworthy in represented 3D scene, by view morphing in exploration interaction.

Besides rebuilding high-density scene geometry, our reconstruction method has improved a lot in terms of time and reliability. However, still existing several limitations that we would like to address in the future. The self-calibration method is highly dependent on the EXIF data of photographs, which may be not trusty as it can be removed or modified. It is hard to define 'little-varied' intrinsic camera and 'moderate wide' baseline, which may depend on the attributes of dataset. More experiments to various scenes are needy for robust evaluation of algorithm. Furthermore, we are planning to extend our system to exploring large scene by registering each small scene to a geo-referenced frame such as Google Earth.

5. Acknowledgement

This paper is supported by National Nature Science Foundation of China (No. 60533070 and 60773153), the Key grant Project of Chinese Ministry of Education (No. 308004), the Project of `Chinese Ministry of Science and Technology (No. 2006BAK12B09), the Project of Beijing Municipal Science and Technology Commission (No.Z07000100560714), National High Technology Project (863 Project) (2006AA01Z333).

6. References

[1] N. Snavely, S. Seitz, and R. Szeliski, "Photo Tourism: Exploring Photo Collections in 3D," *Proc. ACM Transactions on Graphics*, 25(3): 835-846, 2006.

[2] M.Vergauwen, and L.Van.Gool, "Web-based 3D Reconstruction Service," *Machine Vision and Applications*, 17(2): 321-329, 2006.

[3] N. Snavely, S. Seitz, and R. Szeliski, "Modeling the World from Internet Photo Collections," *Intl. Journal of Computer Vision*, 2007.

[4] P. Beardsley, P. Torr, and A. Zisserman, "3D Model Acquisition from Extended Image Sequences," *Proc. 4th European Conf. Computer Vision*, pp. 683-695, 1996.

[5] M. Pollefeys, F. Verbiest, and L. Van Gool, "Surviving Dominant Planes in Uncalibrated Structure and Motion Recovery," *Proc. 7th European Conf. Computer Vision*, 2002.

[6] O. Faugeras, and R. Keriven, "Complete Dense Stereovision Using Level Set Methods," *Proc. 5th European Conf. Computer Vision*, pp. 379-393, 1998.

[7] M. Lhuillier, and L. Quan, "A quasi-dense Approach to Surface Reconstruction from Uncalibrated Images," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(3): 418-433, 2005.

[8] D. G. Lowe, "Distinctive Image Features from Scale Invariant Keypoints," *Intl. Journal of Computer Vision*, 60(2): 91-110, 2004.

[9] S. Arya, et al, "An Optimal Algorithm for Approximate Nearest Neighbor Searching Fixed Dimensions," *Journal of the ACM*, 45(6): 891-923, 1998.

[10] R. Hartley, "In Defense of the 8-point Algorithm," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(6): 580-593, 1997.

[11] R. Hartley, and A. Zisserman, "Multiple View Geometry in Computer Vision," *Cambridge University Press*, 2000.

[12] M. Fischler, and R. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Application to Image Analysis and Automated Cartography", *Comm. ACM*, vol.24, pp. 381-395, 1981.

[13] P. Torr, A. Fitzgibbon, and A. Zisserman, "Maintaining Multiple Motion Model Hypotheses through Many Views to Recover Matching and Structure", *Proc. ICCV*, pp. 485-491, 1998.

[14] M. Brown, and D. Lowe, "Unsupervised 3D Object Recognition and Reconstruction in Unordered Datasets," *Proc. Intl. Conf. on 3D Digital Imaging and Modeling*, pp. 6-63, 2005.

[15] B. Triggs, et al, "Bundle Adjustment: a Modern Synthesis," *Intl Workshop on Vision Algorithms*, pp. 298-372,1999.