LOCATION DETERMINATION IN URBAN ENVIRONMENT FROM IMAGE SEQUENCES

Qingming Zhan [a,b,c], Yubin Liang [b,c], Yinghui Xiao [a,b]

[a] School of Urban Design, Wuhan University, Wuhan 430072, China
[b] Research Center for Digital City, Wuhan University, Wuhan 430072, China
[c] School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China
qmzhan@whu.edu.cn; lyb.whu@gmail.com; yhxiaoitc@126.com

KEY WORDS: Location Determination Problem, Bundle Adjustment, Image Matching, Point Transfer, Pose Estimation, RANSAC

ABSTRACT:

Location Determination Problem (LDP) is a classic and interesting problem both for photogrammetry and computer vision community: Given an image depicting a set of landmarks with known locations, determine that point in space from which the image was obtained. In this paper we try to use image sequences to automatically solve LDP in local Euclidean space in which no georeference information is needed. Overlapping image sequences are preferable for matching images obtained in cities. We implement a method which can semi-automatically solve LDP in urban scenario with state-of-the-art 3D reconstruction system.

1. INTRODUTION

Nowadays Google Maps and other city-scale 3D reconstruction systems with street view are widely used for visual exploration of cities. Those systems often rely on structured photos captured using sensors equipped with GPS and Inertial Navigation Units which make post-processing much easier. However, these systems only cover large cities and famous avenues attractive to tourists. Furthermore, many people do not need absolute georeference information in daily vision-related applications such as augmented reality. Only location information in local space is enough. For example, given an image taken from a place (Figure 1a), one can guess that the photo was taken from a window of a nearby building (Figure 1b) according to viewing direction of the given image. But it's difficult to locate the precise location. The authors of this paper are interested in such a problem: given an image I, locate the place in another image J where image I is taken.

Figure 1a

Figure 1b

Figure 1: Given image and building where the image was taken

The problem is defined as 'space resection' in photogrammetry community and 'pose estimation' or 'extrinsic camera calibration' in computer vision community. Extrinsic camera calibration is often carried in calibration field using well-designed targets/rigs. This is not the case in our problem because there're no pre-installed rigs and image I is taken arbitrarily. The difference between space resection and pose estimation is that the given image points in space resection is georeferenced, whereas pose estimation is usually in local Euclidean space. Location Determination Problem is a general definition both for the photogrammetry and computer vision communities: Given a set of m control points, whose 3-dimensional coordinates are known in some coordinate frame, and given an image in which some subset of the m control points is visible, determine the location (relative to the coordinate system of the control points) from which the image was obtained. (Fischler, Bolles, 1981) presented the well known model-fitting paradigm Random Sample Consensus (RANSAC) and use model inliers to solve the "perspective-n-point" problem (PnP). The PnP problem which is an equivalent but mathematically more concise statement of the LDP is originally defined in (Fischler, Bolles, 1981) as: Given the relative spatial locations of n control points, and given the angle to every pair of control points from an additional point called the Center of Perspective (CP), find the lengths of the line segments ("legs") joining the CP to each of the control points. The aim of the Perspective-n-Point problem (PnP) is to determine the position and orientation of a camera given its intrinsic parameters and a set of n correspondences between 3D points and their 2D projections (Moreno-Noguer, Lepetit, Fua, 2007). Therefore, the solution to our problem mainly resorts to pose estimation in reconstructed local Euclidean space. To automate the process, 3D reconstruction of the scene depicted in image I should be done first. Then the 3D reconstructed scene is used to determine the 3D position of the view point of image I. And the calculated 3D coordinate of view point is projected to image J to visually locate the position.

Nowadays, given a set of overlapping images of a scene shot from nearby camera locations, it's easy to create a panorama that seamlessly combines the original images and reconstruct the 3D scene using extracted correspondences among several images. (Fitzgibbon, Zisserman, 1998) presented method that could simultaneously localize the cameras and acquire the sparse 3D point cloud of the imaged objects using closed or open image sequences. (Lowe, 1999; Lowe, 2004) presented Scale Invariant Feature Transform (SIFT) operator to extract features that are invariant to image scale and rotation, which can be used to robustly match images across a substantial range of affine distortion and change in 3D viewpoint. (Zhang and Koseca, 2006) used SIFT to geo-locate images by finding geo-tagged image match in pre-built database. But we don't assume geo-location information such as geo-tags in our research for generality purpose. (Snavely et al., 2006; Snavely et al., 2008) presented state-of-the-art system (called Bundler) that can automatically structure large collections of unordered images and they have scaled up the Structure From Motion (SFM) vision algorithms to work on entire cities (Agarwal et al., 2011) using photographs obtained from image resource website like Flickr.

While systems like Bundler can automatically reconstruct 3D scenes using a large number of unordered images, the system might produce erroneous result due to lack of enough overlap between images and good estimate of focal length of images. The second problem can be solved by adding the CCD width database to the EXIF reading script. As to the second problem, the authors of the system recommend at least 15 degrees interval between nearby viewpoints, but this condition cannot be satisfied in our case as we don't limit the angle of viewpoints between image I and other images. As a result, the image I will not be registered with the other photos, because there weren't enough matches and angle between camera viewpoints is relatively large.

Based on the above observation, we present a method detailed in section 2 to solve the LDP problem in urban environment. The experiment result and discussion is described in section 3.

2. METHODOLOGY

If we've got an image I depicting a set of landmarks with known locations, then we can determine that point in space from which the image was obtained by space resection or pose estimation. So we can first reconstruct the 3D scene appears in image I using overlapping images collected afterwards and extend the image sequences to cover the whole building. Because we cannot register image I with other images as mentioned above, we cannot directly obtain the 3D position corresponding to points in image I by 3D construction of the scene. So we have to match image I with images used in 3D reconstruction and obtain the 3D position of points in image I by transfer: given the position of a point in one (or more) image(s), determine where it will appear in all other images of the set (Hartley and Zisserman, 2004). After we've got the 3D position of the viewpoint of image I by pose estimation, we can project it to the images covering the building and visually locate the place. So our method could be mainly separated to three steps:

1. 3D reconstruction of the scene: reconstruct the scene using image sequences covering the landmarks in image I and the nearby environment (e.g. the building).
2. Point transfer: match image I and images used to do 3D reconstruction and transfer image points with known 3D position to image I.
3. Pose estimation and viewpoint projection: solve PnP problem using the points in image I and their corresponding 3D position obtained in step 2. Project the calculated viewpoint of image I to images covering nearby environment.

2.1 3D reconstruction of the scene

We've compiled Bundler v0.4 under Linux and use the system to create a 3D reconstruction. We first extract image information (including focal length and image resolution) using Perl script. Interest points are detected in the given image I as well as each image in image sequences using SIFT operator. Images are matched against each other using approximate nearest neighbour search. Mismatches often result from clutters and shadows which are common in urban scenes. RANSAC is used to detect and remove outliers in point correspondences. The main program "bundler" solves the Bundle Adjustment problem using Levenberg–Marquardt algorithm. After all possible images have been registered, Bundler outputs 3D reconstruction containing the reconstructed cameras and sparse 3D points. The estimated extrinsic and extrinsic parameters of each registered camera contain:

f: the focal length,
k1, k2: radial distortion coeffs
R: 3x3 matrix representing the camera rotation
t: a 3-vector describing the camera translation

Parameters of each reconstructed point has the form:
position: a 3-vector describing the 3D position of the point
color : a 3-vector describing the RGB color of the point
view list: a list of cameras the point is visible in

The view list begins with the number of cameras the point is visible in and followed by a list of quadruplets <camera> <key> <x> <y>, where <camera> is a camera index, <key> the index of the SIFT keypoint detected in that camera, and <x> and <y> are the detected 2D positions of that keypoint in that camera. We use a pinhole camera model. The origin of the camera coordinate system the center of the image, the positive x-axis points right, the positive y-axis points up and the positive z-axis points backwards. Therefore, the estimated parameters of each camera specified above can be used to project a 3D point X into a camera (R, t, f) by:

$$P = R * X + t \quad (1)$$
$$p = -P / zp \quad (2)$$
$$p' = f * r(p) * p \quad (3)$$

where zp is the third coordinate of P. Equation 1 transforms the coordinates of a 3D points from a world coordinate system to the current camera coordinate system. Equation 2 commits perspective division and Equation 3 converts the coordinates to values in pixel. In the last equation, r(p) is a function that computes a scaling factor to undo the radial distortion (Equation 4):

$$r(p) = 1.0 + k1 * \|p\|^2 + k2 * \|p\|^4 \quad (4)$$

2.2 Point Transfer

We can obtain 3D points and their corresponding positions in image sequences from output of the first step. To find the projections of these 3D points, we must first establish the relationship between image I and images in sequences using a set of auxiliary point correspondences. If image I is registered with other images in bundle adjustment process, we would directly get the 3D position of image I by:

$$X(I) = -R' * t \quad (5)$$

And the projection of viewpoint of camera I into each camera would be calculated by Equation 1, 2 and 3. Then point transfer and pose estimation will not be necessary. But usually image I cannot be registered with other images (none in our experiment) due to large variations of scale and angles of viewpoint. The alternative procedure we take in this research is to transfer points from image(s) used in reconstruction to image I and use the transferred points to estimate the pose of I.

2.3 Pose estimation and viewpoint projection

We use EPnP (Lepetit et al., 2009) to estimate the pose of image I in the reconstructed space. The calculated 3D coordinates X(I) is then projected into images covering the nearby environment. The viewpoint of image I is determined when p'(I) lies in effective area of an image plane.

3. RESULTS AND DISCUSSION

To test the presented method, we firstly took the photo shown in Figure 1a. And the actual place where we took the image is marked with a red circle. Then we went to the square and took a collection of overlapping images (Figure 2). To guarantee the accurate 3D using Bundler, we keep relatively small angle between viewpoints of neighboring images. The reconstructed 3D scene is illustrated in Figure 3, in which the position of 3704 points and pose of 24 cameras are visualized. Figure 4 shows 12 of all 3704 points and their projections in an image. These projected points are transferred to image I in Figure 5. Figure 6 illustrates the actual position and projection of the calculated 3D viewpoint of image I to an image covering the building. Figure 7a, 7b and 7c give show the presented method tested another dataset.

The computation of our method is mainly cost by image matching procedure. Not all of the cameras can be registered using Bundler, and sometimes the reconstruction is not accurate. The result of point transfer has many outliers which often lead to fault estimation. In this research we cut off some outliers by hand and recalculate the viewpoint of image I using EPnP.

Figure 2: 9 of 24 images used to reconstruct the scene

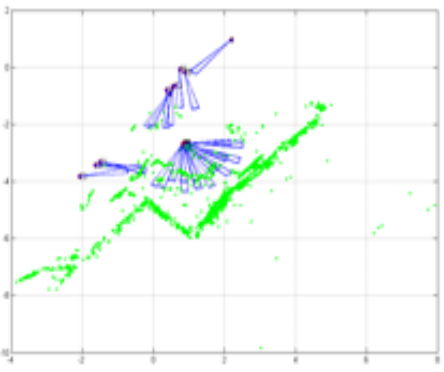

Figure 3: The reconstructed 3D scene (point cloud)

Figure 4: Reconstructed points projected to an image (red crosses)

Figure 5: Point transfer (green crosses)

Figure 6: The estimated viewpoint (green point) of image I and its actual place (red point)

Figure 7a: Anoter test image

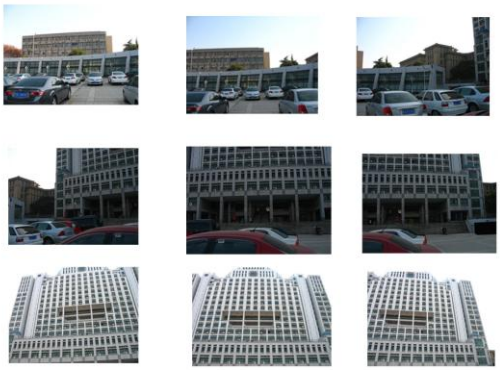

Figure 7b: Images used to reconstruct the scene

Figure 7c: The estimated viewpoint and its actual place

4. CONCLUSIONS AND FUTURE WORK

In this paper we present an approach to solve the Location Determination Problem in urban environment using image sequences. The outdoor scene depicted in the given image is reconstructed first. Points of images used in the reconstruction are transferred to the given image in order to obtain the control points. Then pose of the given image is estimated using PnP solver. The computation of the presented method mainly lies in matching and reconstruction process. In the future work, GPU computation should be considered to speed up the matching process. We may also try other feature operators such as Speeded Up Robust Features (SURF) and ORiented Brief (ORB) to evaluate their performance in our research environment. We will test patch-based algorithms for methods that find both dense and global matches have often had high time cost in the matching stage. In Parallel Tracking and Mapping (PTAM) there are no descriptors as in SIFT but "warped" patches which makes it fast and detectable at bigger angles, which makes it possible to register images with relative large angles between viewpoints. Another function that is worthy to add to our method is to geo-locate the reconstructions and the given image I as well if the image sequences come with geo-tags/GPS information. However, geographical information obtained from images is frequently incorrect, noisy and even missing, which means we must introduce robust estimation method to further improve the accuracy and automation of the presented method.

Acknowledgement

This research is supported by the National Natural Science Foundation of China (No. 40871211).

References

Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz S.M. and Szeliski, R., 2011. Building rome in a day. *Communications of the ACM*, 54(10), pp. 105-112.

Fischler, M.A. and Bolles, R.C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6), pp. 381-395.

Fitzgibbon, A.W. and Zisserman, A., 1998. Automatic camera recovery for closed or open image sequences. In: *Proceedings of the 5th European Conference on Computer Vision*, Freiburg, Germany, pp. 311-326.

Hartley, R.I. and Zisserman, A., 2004. *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge, pp. 379-383.

Lepetit, V., Moreno-Noguer, F. and Fua, P., 2009. EPnP: an accurate O(n) solution to the PnP problem. *International Journal of Computer Vision*, 81(2), pp. 155-166.

Lowe D.G., 1999. Object recognition from local scale-invariant features. In: *Proceedings of the 7th International Conference on Computer Vision*, Li ège, Belgium, pp. 1150–1157.

Lowe D.G., 2004. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), pp. 91-110.

Moreno-Noguer, F., Lepetit, V. and Fua, P., 2007. Accurate non-iterative O(n) solution to the PnP problem. In: *Proceedings of 11th IEEE International Conference on Computer Vision*, Rio de Janeiro, Brazil, pp. 1-8.

Snavely, N., Seitz, S.M. and Szeliski, R., 2006. Photo tourism: exploring image collections in 3D. *ACM Transactions on Graphics*, 25(3), pp. 835-846.

Snavely, N., Seitz S.M. and Szeliski, R., 2008. Modeling the world from internet photo collections. *International Journal of Computer Vision*, 80(2), pp. 189-210.

Zhang W. and Kosecka, J., 2006. Image Based Localization in Urban Environments. In: *Proceedings of the Third International Symposium on 3D Data Processing, Visualization, and Transmission*, North Carolina, Chapel Hill, USA.