The Title of Your Paper Goes Here

Yuhang Li*
Xuejin Chen†

摘要

Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

CR Categories: I.3.3 [Computer Graphics]: Three-Dimensional Graphics and Realism—Display Algorithms I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Radiosity;

Keywords: geo-localization, point cloud, registration

Links: DL PDF

。

摘要

The abstract should summarize the contents of the paper and should contain at least 70 and at most 150 words. It should be written using the *abstract* environment. (Paper idea: 这个问题可以分三个层次（要不要写进论文？或者可以将related work按照这个来写）：
1. Alignment：求解overhead image（的edge image）和3D model的竖直投影对齐，假定相机光轴方向与竖直反向对齐。求解只有一个自由度的R,两个自由度的T和一个scale（相当于T的z方向的自由度）。
2. Geo-localization: 找到相机的6DoF pose。3D orientation and 3D location.
3. 2D-3D registration；求解KRT，K的自由度自定，至少包括f, u_c, v_c
以上三个层次都可以另外加入一些distortion，如长宽比，剪切比，偏心扭曲等。
)

(Paper idea: 目前我们的算法主要解决的是一个geo-localization的问题，求解camera pose(主要)。然后可以用camera pose来求correspondences (未解决)，最后用correspondences来做geo-registration问题（非主要）。)

1 Introduction

Background/Motivation (Paper idea: 一开始提data types是为了方便下面existing methods里面按照data type的不同展开。) Geo-localization has developed rapidly in recent years with the help of powerful sensors which generate a wide variety of types of data, (xuejin: such as 3D points, dsm??) As a key technique to image-based navigation，augmented reality (AR) and 3D reconstruction, geo-localization in urban outdoor environment has drawn massive attentions in the literature. Geo-localization is more challenging than indoor geo-localization because of more complicated environments [Arth et al. 2015a] with more occlusions, more classes of objects and less regular layouts. (xuejin: explain why outdoor scene is more complicated? occlusions? more classes of objects? no regular shape?)

*e-mail:lyh9001@mail.ustc.edu.cn
†e-mail:xjchen99@ustc.edu.cn

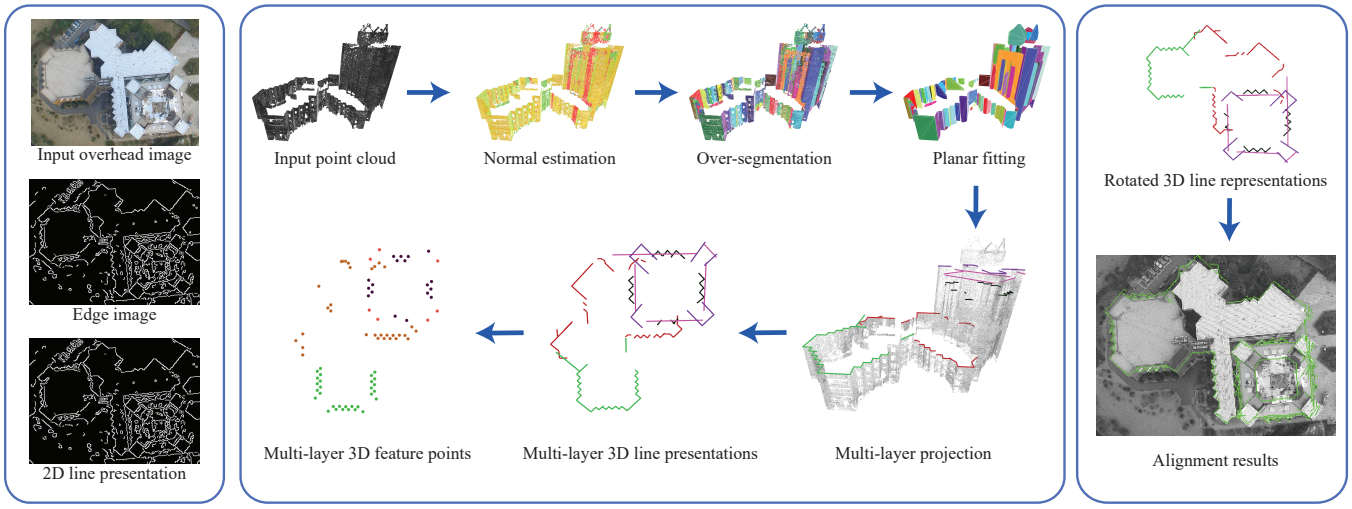

图 1: *Algorithm overview.*

Problem statement (xuejin: Do we solve a geo-localization problem?) (lyh: 目前paper将问题当做geo-localization问题来解，最后再提及用作初始化解registration问题（这部分没有创新点）) Outdoor geo-localization aims to compute a large-scale, robust and accurate camera registration in a global coordinate system with metric scale, while it is also different from Simultaneous Localization and Mapping (SLAM) (New: which does not employ models and therefore presents only relative poses in an arbitrary coordinate system with unknown scale and requires additional motions for initialization [Arth et al. 2015b].) (xuejin: I don't think this is the key difference. When you say global coordinated system, you just use the 3D model as reference coordinate system. No difference with SLAM.) (lyh: 相比于geo-localzation，SLAM有以下特点：

0. SLAM不需要格外的模型，只要图片/视频，绘制出来的地图是在一个未知的scale下做的。

1. 由于同时做定位和地图绘制，于是定位出来的结果也只是在这个未知scale下的坐标内的位置。

2. 由于是未知坐标，后续为了应用到真实世界，还要在SLAM的基础上加上一个alignment方法才能使用。

3. SLAM要有一个初始化的过程，不然没法自建起一个坐标系统。

) (xuejin: what is the key problem? what is input data and output?)

Existing methods and challenges When presenting a geo-localization problem, an image or a frame of video is often used as the query data, a 3D model is needed to provide a global coordinate, (New: a sensor prior is optionally employed,) and the camera pose with respect to the global coordinate system is to be estimated.

(Paper idea: 下面这段按照image的不同，简单讨论了现有的方法)

Some methods capture images or videos by cameras on mobile phones [Arth et al. 2015a; **?**; **?**; **?**] or vehicles [Taneja et al. 2015] on the ground, which are able to take advantages of perspective to compute the camera pose by estimating vanish points. Moreover, some other methods utilize the aerial devices, such as satellites and air plane, to take overhead images at high altitudes, which align the roofs of buildings in images with respect to 3D models (often 3D point clouds), assuming that building roofs in one overhead image captured in high altitude are of a same scale(lyh: 有着相同的scale) and

therefor projecting 3D models along the vertical vector to generate the global coordinate systems. When it comes to low altitude, perspective effect is more critical, making it a more challenging problem.

(Paper idea: 下面这段按照global coordinate system的不同，简单讨论了现有的方法)

To obtain the global coordinate system (xuejin: why to obtain coordinate system?) , some [] maintain databases of pre-registered multiple images. These methods limit themselves while having to collect numerous images covering target areas otherwise only popular spots are available. Another way is employing 3D models, such as digital surface models (DSMs) [] and 3D point clouds [].

(Paper idea: 下面提及点云的获得方法和特点来解释为什么要选择用shape matching方法)

To acquire 3D point cloud data, a common approach is structure-from-motion (SfM) method[] which is convenient requiring only a series of images to recover a point cloud of a scene. The LiDAR sensor in air plane provides a large-scale but low-quality point cloud[]. 3D models comprising with millions of points can be generated by ground laser scan devices, which provide highly accurate and dense data with metric scaling. However, these laser devices usually fail to capture buildings roof while only vertical facades of buildings are scanned. (xuejin: highlight: Missing of roof data makes the registration problem of bird-eye view and 3D models significant challenging.) (lyh: To be solved) More recently [Arth et al. 2015b] presents a novel technique that use an untextured 2.5D map (2D map and height) which is inaccurate in details.(xuejin: 3d models are also unavailable for many cities...)

Our key idea and contributions (xuejin: To solve what problem, we present our method?) In this paper, we present a novel algorithm to (New: geo-localize the camera of) an overhead image captured by a quadrotor in a low altitude by computing the 6-Dimension-of-Freedom(6DOF) camera pose of the image with respect to the point cloud model coordinate of metric scale. At first, we treat this problem as a shape matching problem between the edge image of the overhead image and the 2D line presentation of the point cloud (New: to generate an initial pose for subsequent steps.) (xuejin: why use shape matching? what kind of properties will solve the above problem?) We obsolete the high-altitude assumption that building roofs in overhead images are of same scales and propose a novel multi-level strategy to handle low-altitude overhead images where assign different scales to roofs in different heights. Moreover, once we achieve the shape matching results we determine the correspondences by matching corners of images with feature points of point clouds. At last, we geo-register these two types of data using sparse bundler adjust (SBA) algorithm.

(New: (Paper idea: contributions) To sum up, the contribution of our paper is three-fold. At first, we introduce a novel multi-level strategy to handle the low-altitude image. Moreover, At last,)

2 Related work

In this section a brief review of existing approaches on Geo-localization is given...(lyh: 加上一个概括) (xuejin: more references are needed.)

2.1 Geo-localization with image database

With a large dataset of pre-registered images, image-based methods compute 6DOF camera pose of a given image. [**?**] maintains a dataset of streetside images and uses a vocabulary tree to recognize the location. A wide-baseline matching algorithm is presented by [**?**] to identify corresponding building facades generated from a image dataset in two views. It can handle significant changes of viewpont and lighting. [Zamir and Shah 2014] applies a multiple nearest neighbor feature matching method with a local feature constraint. (xuejin: In comparison, we do what?)

2.2 Geo-localization with models

[Kaminsky et al. 2009] aligns a SfM model of urban environment to a corresponding overhead image by computing an objective function that matches 3D points to image edges under a free space (space without buildings) constraints based on the visibility of points in each camera. They use not only the points of SfM model but also the camera poses of images which are both generated from image collection. [Karl et al. 2013] geo-registered 3D point clouds to a scaled map image by defining a normalized Hough similarity function and aligning planes (i.e., walls) in 3D point clouds to lines in 2D maps. [Zhang et al. 2014] treats the geo-localization as a shape matching problem and aligns 2D the vertical projection of the 3D building roofs and edges of satellite images. An extended Chamfer matching is used to handle noise and occlusions while a global constraint is introduced to optimize the alignment within a large region. In comparison, our method applies a fast version of Chamfer matching algorithm [Liu and Tuzel 2010] to **accelerate** the process where the matching algorithm is based on line segments instead of dense points used in [Zhang et al. 2014], and introduces a multi-level strategy to handle low-altitude overhead images where we assign different scales to roofs in different heights.

2.3 Geo-registration

...

3 Overview

(Paper idea: 其实不是每个点都满足这个投影公式的，只有对应点满足。但是在找到对应点之前的Alignment问题里面也要用到) (lyh: 要不要改成齐次坐标？) Given a 3D point $(x, y, z, 1)$ of the point cloud model, and a 2D point $(u, v, 1)$ of the overhead image captured by a quadrotor in a low altitude, our goal is to compute the transform matrix $\mathbf{RT}$ projecting the 3D the point to the 2D point using

$$m \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{KRT} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}, \quad (1)$$

where m is the normalization term ensuring the third dimension of the 2D point to be one, $\mathbf{K}$ is the pre-calibrated intrinsic matrix of the camera, $\mathbf{R}$ is the rotation matrix and $\mathbf{T}$ is the translation matrix.

(xuejin: do you calibrate the camera intrinsics? how?) (xuejin: three main steps: extraction of feature point correspondences, ..., as Fig 1 shows.) (Paper idea: 这段跟introduction里面的short overview相同) Our algorithm consists of three main steps as Fig 1 shows. At first, we treat this problem as a shape matching problem between the edge image of an overhead image and the 2D line presentation of a point cloud where we assume that the optical axis of the camera aligns to the gravity direction and apply a novel multi-level projection strategy to handle the low-altitude problem. Moreover, we determine correspondences by matching corners of images with feature points of point clouds. At last, these two types of data are registered with the correspondences using sparse bundler adjust (SBA) algorithm.

(xuejin: Explain every variable and every key elements with equations or mathmatical symbols. In order to solve RT, what things should be known?)

4 Shape matching

Our building point clouds are obtained by scanning buildings using a ground laser equipment. Only vertical facades of buildings, walls, are captured, and roofs are not able to be scanned by the ground equipment, as Fig 1 (a) shows. The overhead images are captured by a quadrotor in a low-altitude. We assume the optical axis of the camera aligns the camera aligns to the gravity direction. This assumption will be discarded in Section 6. The fact that vertical facades of a point cloud correspond to lines of building roofs in the overhead image [Zhang et al.

2014] inspires us to project the point cloud along the vertical vector and to treat this geo-localization (lyh: Geo-localization?) problem as a shape matching one in this stage. In this kind of problem, one or multiple templates and a target image are needed. We generate the 2D templates by projecting the point cloud with a multi-level strategy and apply the edge image of the input overhead image as the target image.

Let a point in a 2D template be $(u_p, v_p, 1)$ and a point in 2D target image be $(u_e, v_e, 1)$. We compute a rotation matrix $\mathbf{R}_{sm}$ with only one dimensional freedom, a scale s and a translation $\mathbf{T}_{sm}$ with two dimensional freedom subjected to

$$m \begin{bmatrix} u_e \\ v_e \\ 1 \end{bmatrix} = s\mathbf{R}_{sm}\mathbf{T}_{sm} \begin{bmatrix} u_p \\ v_p \\ 1 \end{bmatrix}, \tag{2}$$

where

$$\mathbf{R}_{sm} = \begin{bmatrix} \cos\phi_z & -\sin\phi_z & 0 \\ \sin\phi_z & \cos\phi_z & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{3}$$

$$\mathbf{T}_{sm} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{4}$$

[Zhang et al. 2014] projects all pre-scaled and pre-rotated points of a building point cloud model along the vertical vector to obtain the template image and extracts the edge image of overhead image as target image. Considering that parts of building roofs with different heights are of different scales in a low-altitude overhead image, it is unreasonable to keep the assumption in [Zhang et al. 2014] that the projection scales of all parts of building roofs are the same. (xuejin: emphasize this in introduction.) To handle the problem without this convenient assumption, we present a novel multi-height strategy, where we match the point cloud projections of different heights with the overhead image separately.

4.1 Point cloud processing

(lyh: 这一节只是说了how, 没有说why) A building point cloud, $\mathcal{P} = \{\mathbf{p}_i\}$, $i = 1, 2, \ldots, N_{Points}$, are pre-processed to get height-related templates for shape matching.

Normal If needed, we compute a normal $\mathbf{n}_i$ for each point $\mathbf{p}_i$ in the point cloud using Fast and Robust Normal Estimation (FRNE) [Boulch and Marlet 2012] which preservers sharp features such as edges and corners.

Vertical direction The vertical direction of building point cloud should correspond to the gravity direction in the realworld. If needed, the vertical direction can be computed using a Principal Component Analysis (PCA) based algorithm [**?**]. And then, we rotate the point cloud to ensure the third dimension of a point, z, presents the height of this point in real world.

Segmentation The point cloud is segmented into clusters $\mathcal{C}_j = \{\mathbf{p}_i^j\}, j = 1, 2, \ldots, N_{Clusters}$ by iterated region growing from a randomly selected point with points that are within a Euclidean distance of ρ and a normal difference of τ. This process expands to new adding points repeatedly until every point is assigned to clusters.

Planar fitting We fit a vertical plane to each cluster. More specifically, for each cluster $\mathcal{C}_j$, we use a RANSAC plane fitting algorithm in PCL [] to fit a finite planar whose normal is constrained to be orthogonal to the vertical vector. The boundary of each planar is the alpha shape of points.

Selected heights A histogram is built, whose horizontal axis represents the height in meter from zero to the building height and the vertical axis presents the number of particular points. To be specific, we find the maximum height h_j^{max} of points in a cluster $\mathcal{C}_j$ and assign this cluster to one of bins of the histogram, B_k, $k = 1, 2, \ldots, N_{Bins}$, making sure that this bin contains

图 2: *Contour extraction from point clouds.*

h_j^{max} in its height range. And then, for each bin, we sum up point numbers of clusters assigned to the bin as its value. Top n_s bins with maximum values are selected, indicated as $B_l^{selected}$, $l = 1, 2, \ldots, n_s$, and the weighted average height $h_l^{selected}$ of each B_l is computed by

$$h_l^{selected} = \sum_j Ind(j,l) w_j h_j^{max}, \tag{5}$$

where $Ind(j,l)$ is a indicator which equals to 1 if $\mathcal{C}_j$ is assigned to $B_l^{selected}$, otherwise equals to 0, and w_j is the weight which equals to the number of points in $\mathcal{C}_j$. (xuejin: Show figures about this step.)

4.2 sensor pose

...

4.3 Shape matching

At this stage, we treat the registration problem as a shape matching problem. In a shape matching problem two edge images are needed, a template image and a target image, and we should find a rigid transformation to fit the template image with respect to (lyh: a part of ?) the target image. To find the rigid transformation, we find the scale, the orientation and the translation respectively.

Scale The scale between the overhead image and the real-world building (also between the overhead image and the metric-scale point cloud) is easy to compute using the intrinsic parameters and altitude parameter from the aircraft sensors, which is accurate enough for later processing. (lyh: Mathematical language?)

Orientation The camera orientation of the overhead image is equal to the alignment rotation between overhead image and point cloud projection image. A Hough transform based algorithm [Censi et al. 2005] is utilized, which is designed to compute the alignment rotation between two edge images, here, a template image and a target image. This algorithm is based on the fact that a rotation with specific angle of an image in space domain result in a translation along the angle axis in Hough transform domain.

Translation Chamfer distance matching is a classical shape matching technique, which sweep every possible transformation with brute force to compute costs between the transformed template image and the target image. It can handle all respects of transformation including scale, rotation, inspect and translation. However, chamfer distance matching could be extremely complex in computation and time-consuming. In fact, the computation complexity is directly proportional to the search space of (lyh: 与搜索空间的每一维的量化个数/粒度成正比) . Therefor, we only use chamfer distance matching technique to find the translation while fixing the scale and the rotation to the results computed above and the inspect to one. Also we apply to speed up a fast version of chamfer distance matching algorithm, Fast Directional Chamfer Matching (FDCM)[Liu and Tuzel 2010]. FDCM incorporates edge orientation information in the matching algorithm such that the resulting cost function is piecewise smooth and the cost variation is tightly bounded. Moreover, FDCM is proved to be a sublinear time algorithm using techniques from 3D distance transforms and directional integral images. To compute the transformation, FDCM requires as input two sets of line segments, a template set and a target set, from two input edge images respectively. In [Liu and Tuzel 2010], line segments of both two sets are estimated from corresponding edge images using a RANSAC line fitting algorithm. However, the proposal method use the line segments generated in point cloud processing as the template sets. (lyh: 为什么要用point cloud里面的lines来代替原方法的lines？)

(lyh: Why dont we just use FDCM to compute scale and rotation? Computation complexity?)

5 Feature point matching

At this point,

6 Geo-registration

7 Experiments

The proposed whole system was tested using our building point clouds and corresponding images. The building point clouds are obtained by scanning buildings using a ground laser equipment around our campus. The images are captured using a quadrone

8 Conclusion

Acknowledgements

To Robert, for all the bagels.

参考文献

ARTH, C., PIRCHHEIM, C., LEPETIT, V., AND VENTURA, J. 2015. Global 6DOF Pose Estimation from Untextured 2D City Models. *arXiv preprint arXiv:1503.02675*.

ARTH, C., PIRCHHEIM, C., VENTURA, J., SCHMALSTIEG, D., AND LEPETIT, V. 2015. Instant Outdoor Localization and SLAM Initialization from 2.5D Maps. *IEEE Transactions on Visualization and Computer Graphics 21*, 11, 1309–1318.

BOULCH, A., AND MARLET, R. 2012. Fast and robust normal estimation for point clouds with sharp features. In *Computer graphics forum*, vol. 31, Wiley Online Library, 1765–1774.

CENSI, A., IOCCHI, L., AND GRISETTI, G. 2005. Scan Matching in the Hough Domain.pdf. 39–44.

KAMINSKY, R. S., SNAVELY, N., SEITZ, S. M., AND SZELISKI, R. 2009. Alignment of 3D point clouds to overhead images. *CVPRw*, 63–70.

KARL, N., NICHOLAS, A., AND SCOTT, S. 2013. GEO-REGISTERING 3D POINT CLOUDS TO 2D MAPS WITH SCAN MATCHING AND THE HOUGH TRANSFORM. 1864–1868.

LIU, M.-Y., AND TUZEL, O. 2010. Fast Directional Chamfer Matching. *CVPR*, 1696–1703.

TANEJA, A., BALLAN, L., AND POLLEFEYS, M. 2015. Geometric Change Detection in Urban Environments Using Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence 37*, 11, 2193–2206.

ZAMIR, A. R., AND SHAH, M. 2014. Image geo-localization based on multiplenearest neighbor feature matching using generalized graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence 36*, 8 (Aug), 1546–1558.

ZHANG, X., AGAM, G., AND CHEN, X. 2014. Alignment of 3D Building Models with Satellite Images Using Extended Chamfer Matching. *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 746–753.