



HAL
open science

Video/GIS registration system based on skyline matching method

Shupeng Zhu, Luce Morin, Muriel Pressigout, Guillaume Moreau, Myriam
Servièrè

► **To cite this version:**

Shupeng Zhu, Luce Morin, Muriel Pressigout, Guillaume Moreau, Myriam Servièrè. Video/GIS registration system based on skyline matching method. 20th IEEE International Conference on Image Processing ICIP 2013, Sep 2013, Melbourne, Australia. pp.3632-3636, 10.1109/ICIP.2013.6738749 . hal-01296969

HAL Id: hal-01296969

<https://hal.science/hal-01296969>

Submitted on 12 Apr 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

VIDEO/GIS REGISTRATION SYSTEM BASED ON SKYLINE MATCHING METHOD

S. Zhu, L. Morin, M. Pressigout*

G. Moreau, M. Servières

IETR
INSA Rennes
35700 Rennes Cedex 7, France

CERMA
Ecole Centrale de Nantes
44321 Nantes Cedex 3, France

ABSTRACT

Video/GIS registration in urban environments is necessary for establishing correspondences between real objects and geometric information in Geographic Information System (GIS) applications such as outdoor Augmented Reality (AR) and building texture acquisition. In this work, we present a registration system using a GPS prior and a skyline matching method to estimate the camera pose from images. A skyline rectification method through vertical vanishing point detection is also developed to enable arbitrary camera orientation, with non nul tilt. The proposed approach is robust, fully automatic and computationally inexpensive which makes it a possible solution for mobile device applications. The performance of our approach is demonstrated in several experimental evaluations. The possible refinement of camera location through a systematic method is also investigated.

Index Terms— Video/GIS registration, Skyline, Vanishing point, Temporal processing, Augmented reality

1. INTRODUCTION

Video/GIS registration in urban environments is necessary for establishing correspondences between real objects and geometric information in Geographic Information System (GIS) applications such as outdoor Augmented Reality (AR) and building texture acquisition. In this work, we present a registration system using a GPS prior and a skyline matching method to estimate the camera pose through the estimation of the three Euler angles. Camera pose is defined by extrinsic parameters, rotation matrix \mathbf{R} and translation vector \mathbf{t} . Internal parameters defined by matrix \mathbf{K} are supposed to be known. Rotation may alternatively be defined by three Euler angles Yaw (α), Pitch ($\frac{\pi}{2} - \beta$) and Roll (γ) (Fig. 1)[1].

The framework of the system will be presented in section 3. This system is an extension of the skyline matching method presented in [2] which can recover the yaw angle from the image with the constraint that all cameras have zero pitch and roll. In order to overcome these limitations, we have developed a method which can calculate pitch and roll

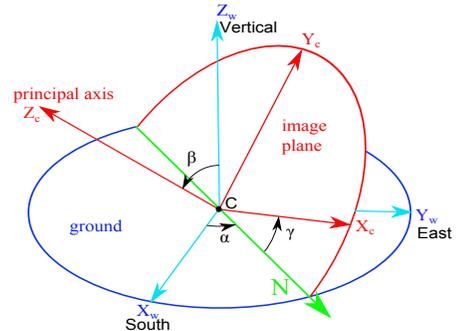


Fig. 1. Euler angles

angle using vertical vanishing point detection that will be presented in section 4. With such knowledge, features extracted from the original image can be rectified into a horizontal position. A temporal processing method was also developed using the consistency between video frames, which significantly improves the robustness of skyline matching method (section 5). The performance of the system is demonstrated by a series of tests using both synthetic and real images in section 6. A possible refinement of camera location estimation through a systematic method is also investigated (section 7).

2. RELATED WORKS

There are 2 main types of approaches for camera registration from ground-base images. In image-based approaches, the input image is matched with a database of geo-localized images acquired off-line [3]. GIS-based approaches mostly rely on matching geometric features of 3D building models to image features. Various geometric features have been explored as the bridge to establish correspondence between image and GIS models, such as lines in [4], edges in [5] and angles between facades in [6]. They are usually recovered using partial 3D reconstruction from the original image. Their common limitation is that they can only represent characteristics of individual buildings but not their combination, which reduces their robustness, especially in dense urban environments where buildings are likely to be connected in large

*Thanks to RUBI3 project for funding.

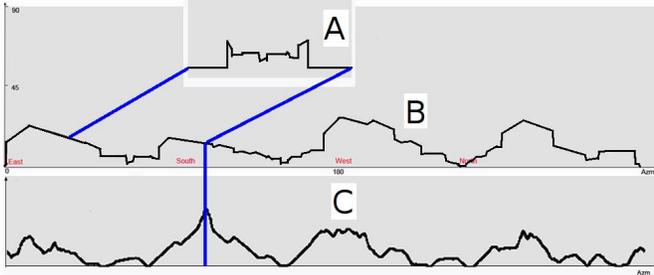


Fig. 2. Skyline matching: Cross-Similarity result (curve C) between Partial Skyline A and Panoramic Skyline B[2]

blocks.

To avoid such drawbacks, we investigated the skyline feature which can represent the overall geometric character of a street in an ground-based image and at the same time avoid direct building feature matching. In the work presented in [2], the basic idea was to compare the partial skyline feature from the image with a synthetic panoramic skyline generated from GIS data using the same camera location, the best match between them providing the camera orientation. However, this method assumes an up-right camera with strict horizontal viewing direction (or optical axis), which means zero pitch and roll. Therefore, if the assumption is valid, such a method provides yaw angle only. Such upright image condition can be satisfied through image rectification as proposed in the work of Vertura [7]. We will described our rectification process in section 4 based on the roll and pitch angles with the help of vertical vanishing point detection. Vanishing point detection is a well-studied problem, existing methods are generally based on either Hough transform, RANSAC or exhaustive search [8]. Kalantari et al. [9] introduced a new detection method based on Thales' theorem which is more efficient and robust than traditional methods. A similar method is used in our system to detect the vertical vanishing point. However, instead of rectifying the whole image into upright position like [7], we only rectify the skyline feature which helps to reduce the computation complexity.

outline in gray the parts that have been changed or that are new: rectify.... pose calculation

3. SYSTEM OVERVIEW

The proposed method relies on matching the partial skyline extracted from the images with the skyline generated from the GIS to estimate the camera orientation. Image skyline must first be rectified to compensate for pitch and roll. Fig.2 illustrates the core algorithm of this method. Curve A is the partial skyline extracted from the image, curve B is the panoramic skyline generated using 3D building models from GIS. Algebraically, both curves can be expressed as a function about azimuth angle φ and elevation angle θ . Here, $f(i)$ stands for

curve A and $g(i)$ stands for curve B. In our method, the partial skyline $f(i)$ is dragged along the panoramic skyline $g(i)$ and at each iteration, the Cross-Similarity factor is calculated as follows:

$$CS(fg)[i] = \sum_{j=0}^N \begin{cases} 1 - Sim & \text{if } Sim < 0.25 \\ 0 & \text{else} \end{cases} \quad (1)$$

Where $Sim = \left| \frac{f(j) - g(i-N+j)}{f(j)} \right|$. Finally, the yaw angle is fixed by the position of the maximum value of CS (correspondence peak) on curve C, thanks to the knowledge of correspondence between the azimuth angle φ of curve B and the world coordinate system.

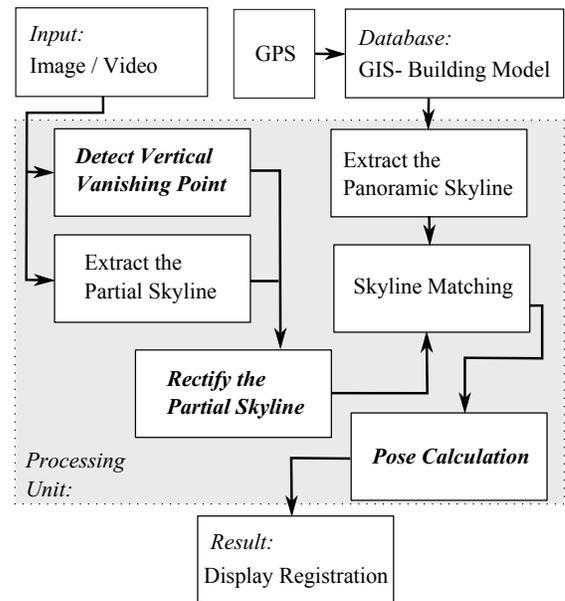


Fig. 3. Framework of the system

The framework of the system is presented in Fig.3. Basically, the system can be divided into two parts before the skyline matching process. At first, the location information provided by GPS is used to query the surrounding building model from the GIS and to generate the panoramic skyline from those models. Then, the input image or video is used in two kinds of algorithms, one focused on the extraction of the partial skyline from the image, the other one detecting the vertical vanishing point and computing the pitch and roll angles. After this, the partial skyline is rectified into the horizontal position based on such information. Finally the skyline matching process can be conducted in order to estimate the yaw angle. The last step is the registration of the camera pose with the GIS building models and the display of the superimposed result. More details may be found in [2] except for vanishing point detection and skyline rectification. So, we will focus on those two parts in the remaining of the paper.

4. SKYLINE RECTIFICATION

We will focus on the improvement of previous work [2], ie the skyline detection using the vanishing point detection.

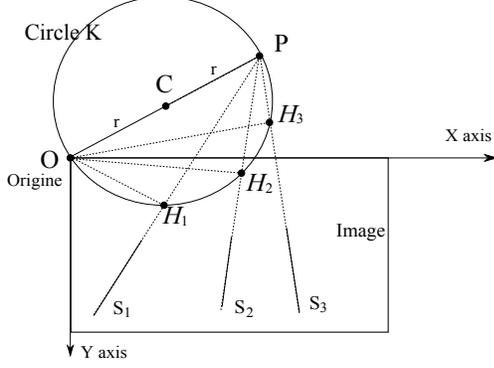


Fig. 4. S_1, S_2, S_3 are vertical lines on image, they converge at point P - vertical vanishing point. O is the original point of the image, OH_i perpendicular to S_i . Based on Thales' theorem, points O, H_i, P are on the same K -circle [9].

Based on Kalantari's work [9], the detection of vertical vanishing point sequels to the detection of the K -circle in Fig. 4. The K -circle is the geometric location of the H points realizing the minimum distance (i.e. (OH_i) is the orthogonal line to segment S_i) between a segment and the origin O . First, lines are detected using LSD [10], which is more accurate than Canny based methods. Short lines are discarded as noise and duplicated lines are also filtered to reduce the computation cost. To obtain vertical segments, remaining lines are filtered by their slope angles, keeping only those within 15° around vertical lines. Then all H points of remaining lines are calculated and a RANSAC procedure is employed to select the inlier set from all H points. Finally, the K -circle is estimated using least squares method on account of remaining H points and the position of vertical vanishing point is fixed.

Then, the pitch (β) and roll (γ) angle can be calculated based on projective geometry, a vertical vanishing point represents one infinity point in vertical direction which can be written as: $V_w = [0, 0, 1, 0]^T$ in homogeneous coordinates. The position of vertical vanishing point $VP_v(vp_x, vp_y)$ is thus determined by projection V_w onto the image :

$$\begin{bmatrix} vp_x \\ vp_y \\ 1 \end{bmatrix} \sim [\mathbf{K}] \begin{bmatrix} R_{11} & R_{12} & \sin \beta \sin \gamma & t_1 \\ R_{21} & R_{22} & \sin \beta \cos \gamma & t_2 \\ R_{31} & R_{32} & \cos \beta & t_3 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \quad (2)$$

where $[t_1, t_2, t_3]^T$ stands for the camera translation and R_{ij} represent other coefficients of rotation matrix which have no influence on the result in this case. \sim stands for equality up to a scale factor. Under the hypothesis of previously determined

intrinsic parameters, $[\mathbf{K}]$ is known, thus we have:

$$[\mathbf{K}]^{-1} \begin{bmatrix} vp_x \\ vp_y \\ 1 \end{bmatrix} = \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} \sim \begin{bmatrix} \sin \beta \sin \gamma \\ \sin \beta \cos \gamma \\ \cos \beta \end{bmatrix} \quad (3)$$

Finally, by comparing results of both sides, both pitch (β) and roll (γ) can be expressed as a function of vanishing point coordinates:

$$\gamma = \arctan \left(\frac{V_1}{V_2} \right) \quad \beta = \arctan \left(\frac{V_2}{V_3 \sin \gamma} \right) \quad (4)$$

Once pitch and roll are known, each point of partial skyline $[x_i, y_i, 1]^T$ can be rectified into upright position $[x'_i, y'_i, 1]^T$ by inverting the rotation process of pitch and roll:

$$\begin{bmatrix} x'_i \\ y'_i \\ 1 \end{bmatrix} = [\mathbf{K}][\mathbf{R}_{\text{inv}}][\mathbf{K}]^{-1} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \quad (5)$$

where:

$$[\mathbf{R}_{\text{inv}}] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \beta & \sin \beta \\ 0 & -\sin \beta & \cos \beta \end{bmatrix}^{-1} \begin{bmatrix} \cos \gamma & \sin \gamma & 0 \\ -\sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}^{-1} \quad (6)$$

One result is shown in Fig. 5. Tests on 200 synthetic images show the average error of algorithm is: 1.229° (standard deviation 1.245°) for pitch angle and 0.552° (standard deviation 0.329°) for roll angle, which is precise enough for the following process of skyline matching.

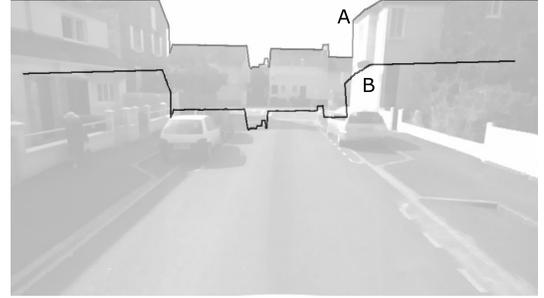


Fig. 5. A is the original skyline (Pitch= 12.71° , Roll= -0.74°), B is the skyline after rectification (Pitch= 0° , Roll= 0°)

5. TEMPORAL PROCESSING

One advantage of video input is that we can use the consistency between each frame and optimize the matching procedure which is called temporal processing. The core of skyline matching algorithm consists of using the Cross-Similarity peak between two skylines [2] and then to estimate the yaw angle. If more than one similar part is present in the panoramic skyline, matching might fail. This occurs typically for opposite orientations in street environments. In order to reduce



Fig. 6. Experiment Results

Table 1. Test results on 1419 Synthetical images

Accuracy($\varepsilon < X^\circ$)	Group 1 (%)	Group 2 (%)
0.5	21.21	23.04
1.0	67.51	71.32
2.0	87.81	91.68

such ambiguity, we assume that the difference of yaw angle between two successive frames is limited (30° is found to be enough), which provides a constraint on the peak searching range. In practice, such constraint can be defined by the GPS difference between two frames. However, if the camera orientation is different with the moving direction of its mounting platform, such constraint should be defined by the result of last frame.

6. EXPERIMENTS

The performance of the system is examined using both synthetic and real images on a PC platform (Inter Core2 CPU E8400. 3.00GHz). First, 1419 sequential synthetic images are generated by simulating a car driving through the city center of Rennes, France. The rotation error ε is calculated as the angle between the optical axis of the estimated result and the correct result. In order to evaluate the effect of temporal processing, the test is divided into two groups. Temporal processing is employed only for group 2. Table 1 shows the test results, which are divided into three accuracy levels. $\varepsilon < 0.5^\circ$ corresponds to an almost perfect registration. $\varepsilon < 1.0^\circ$ corresponds to a satisfying registration. For $\varepsilon > 2.0^\circ$ the mismatch becomes obvious and it is considered as a bad registration. From the results we can see that group 2 is better than group 1 in all levels, which means the temporal processing indeed improves the robustness of the system. For group 2 the percentage of bad registration is less than 10% which validates the proposed method. The average computation time is 671 ms, from which the most time consuming part is the vanishing point detection (472 ms).

Real images are also tested. Fig. 6(a) shows one registration result with an image taken from a new developed sub-

urban area. In spite of negative influence of trees and street lights on the skyline shape, we can see the registration result is quite good. This shows that the proposed method can endure large occluding elements in the skyline.

7. LOCATION OPTIMIZATION

The location optimization method is a new contribution with respect to previous work. As for now, the proposed approach is based on the assumption that the camera position provided by GPS is accurate. However, in reality, GPS position is not so precise (errors around 1-5 m). It is better if we can refine the camera position using our system. Based on such concern, a systematic method is developed to solve this problem. The main idea is that instead of just matching the partial skyline with only the panoramic skyline generated from GPS position, it is compared with a series of panoramic skylines generated from surrounding locations. During this process, the similarity value of each location is recorded, and the position with highest similarity is considered as the optimized camera location. A hierarchical grid search with 3 resolution levels is performed in order to reduce computational complexity. Fig.6(b) and Fig.6(c) compares the result obtained with original location with an obvious error and the result obtained after location optimization.

8. CONCLUSION

We have presented a method for Video/GIS registration based on skyline matching and supported by vanishing point detection and temporal processing. Using synthetic and real image sequences, the efficiency and robustness of the proposed approach has been proved. One great advantage of our method is that it is full automatic and it can be initialized from any image. Our future work is focused on the improvement of the algorithm computation efficiency to make it a possible solution for real-time applications. We also aim to improve the robustness of the proposed method with respect to noisy images (involving for example the detection of urban furniture) and cloudy sky.

9. REFERENCES

- [1] J. Diebel, “Representing attitude: Euler angles, unit quaternions, and rotation vectors,” 2006.
- [2] S. Zhu, M. Pressigout, M. Servières, L. Morin, and G. Moreau, “Skyline matching : A robust registration method between video and gis,” in *3U3D*, 2012.
- [3] C. Arth, M. Klopschitz, G. Reitmayr, and D. Schmalstieg, “Real-time self-localization from panoramic images on mobile devices,” in *ISMAR*. IEEE, Oct. 2011, pp. 37–46.
- [4] T. Collet, G. Sourimant, and L. Morin, “Automatic Initialization for the Registration of GIS and Video Data,” in *3DTV*. IEEE, May 2008, pp. 49–52.
- [5] G. Reitmayr and T. Drummond, “Going out: robust model-based tracking for outdoor augmented reality,” IEEE, Oct. 2006, pp. 109–118.
- [6] N. Bioret, M. Servières, and G. Moreau, “Urban Localization based on Correspondences between Street Photographs and 2D Building GIS Layer,” in *CORESA*, 2009.
- [7] J. Ventura and T. Höllerer, “Structure and motion in urban environments using upright panoramas,” *Virtual Reality*, Feb. 2012.
- [8] C. Rother, “A new approach to vanishing point detection in architectural environments,” *Image and Vision Computing*, vol. 20, no. 9-10, pp. 647–655, Aug. 2002.
- [9] M. Kalantari, F. Jung, and J. Guedon, “Precise, automatic and fast method for vanishing point detection,” *The Photogrammetric Record*, vol. 24, no. 127, pp. 246–263, Sept. 2009.
- [10] G. Rafael, J. Jérémie, M. Jean-Michel, and R. Gregory, “LSD: a Line Segment Detector,” *Image Processing On Line*, pp. 1–10, 2012.