# Image Retrieval of First-Person Vision for Pedestrian Navigation in Urban Area

*Yoshinari Kameda and Yuichi Ohta*
*Graduate School of Systems and Information Engineering, University of Tsukuba, Japan.*
*kameda@iit.tsukuba.ac.jp*

## Abstract

*We propose a new computer vision approach to locate a walking pedestrian by a camera image of first-person vision in practical situation. We assume reference points have been registered with other first-person vision images. We utilize SURF and define seven matching criteria that derive from the property of first-person vision so that it rejects false matching. We have implemented a preliminary system that can respond to a query within 1/2 seconds for a path of approximately 1 km long around Tokyo downtown area where pedestrians and vehicles are always in images.*

## 1. Introduction

In urban areas, estimating one's position is not an easy problem by conventional sensors such as GPS, radio-wave strength, beacons, etc yet it is useful for personal navigation. Especially, visually impaired people need a new localization method that can work at any scenes in urban areas; from underground to narrow open-sky pavement with tall buildings. First-person vision is expected to be a key technology for that purpose. As their demand is to have a guide for a path that they plan to go through, their demanding navigation system does not need to cover whole downtown area.

We propose a new image retrieval method of first-person vision that works effectively and rapidly with a set of first-person view snapshots in urban area.

We assume reference points on a path are associated with images of other first-person vision taken by someone in advance. Our problem here is image retrieval of input image of first-person vision against a set of prerecorded first-person vision images.

## 2. Related works

In research literature of robot and vehicle navigation, various vision approaches have been proposed. However, since these approaches expect that there are not so many disturbing objects and/or the camera moves smoothly, they could not be directly applied to first-person vision images of pedestrians in downtown areas.

A first-person vision based pedestrian navigation can be recognized as a part of general image retrieval problem of snapshots. For querying photos, good object category recognition methods such as [1,2,3] have been proposed. However, since they are designed to treat general images or some limited scenes, their approach does not directly fit to our problem. First-person vision is a new research category and some approaches have already done such as [4], but they cannot treat wide variety of urban situations.

## 3. First-person vision image

Since we think the proposed method could be a help for visually impaired pedestrian navigation, we need to accept camera-mount constraints that comes from their physical limitation and their request. The camera is set on the frontal surface of his/her upper body and it is headed to their walking direction. It means that the camera is not rigidly mounted and its image may be interfered by other pedestrians or objects frequently in downtown areas.

We also have to think wide variety of situations from indoor to outdoor scenes in conjunction with stairs, gates, crowded pavements, etc.

As we have to deal with various interfering objects like other pedestrians, bicycles, and vehicles, we choose locality based operator SURF [5] for image features. As SURF has high ability of describing local features, it is helpful to recognize an image even when only a part of the image is available for matching.

Reference images are taken by a camera in the same setup. Note that reference images are also taken in practical situation at different time and day. That means interfering objects and light changes are inevitable even in reference images.

## 4. Verification of key point pairs

### 4.1. Query process overview

We first apply a conventional bag-of-features approach with SURF keys to find out the top candidate image among reference images for a new query image.

Then, by taking account that both the query image and the selected reference image are first-person vision images, we propose new criteria to verify the SURF key pairs so as to tell the selected one is acceptable.

### 4.2. Top image candidate

Suppose a pedestrian is going to walk along a specific path and there are $\mathrm{N}$ reference points in the path. Image DB holds $\mathrm{N}$ associated images, each reference image is numbered by $\mathrm{R}\,(1 \leq \mathrm{R} \leq \mathrm{N})$. SURF keys, denoted by $\mathrm{k_{iR}}$ for image $\mathrm{R}$, are calculated and stored in the DB in advance. Then a query image Q is given to the system and SURF keys $\mathrm{k_{iQ}}$ are calculated immediately.

Top candidate image $\mathrm{R}^*$ is given by

$$\mathrm{R}^* = \max_{\mathrm{R}}(\mathrm{CountSURFPairs}(\mathrm{Q}\,,\mathrm{R}))$$

$$\mathrm{K} = \mathrm{CountSURFPairs}(\mathrm{Q}\,,\mathrm{R}^*)$$

Where $\mathrm{CountSURFPairs}(\mathrm{Q}\,,\mathrm{R})$ counts the number of SURF pairs between image Q and R, by estimating only the similarity of SURF descriptors [3]. $\mathrm{K}$ means the number of the pairs found between image Q and $\mathrm{R}^*$. As this part is a simple similarity based search, we speed up this process by applying KD tree search.

### 4.3. Verification criteria

The number of the SURF pairs between image Q and R depends on their similarity. It could be affected by the location and direction difference of the cameras, objects in front of the cameras at the moment, and the time and day of taking the images.

From our experience on actual first-person vision videos in downtown areas in Japan, we can say that the number of found pairs is not so large (less than one hundred, approximately up to a couple of tens) even in the case that the cameras are at almost the same position, at the same direction, and there might be some false-positive pairs. Therefore, we need to invent new criteria so as to examine the top candidate based on the attributes of key-point features such as position, size, and orientation.

For the first-person view images, the objects close to the camera is usually useless for SURF based feature matching because the objects come to a side of the camera, and their appearance changes rapidly as the camera goes. Therefore, the size of the SURF keys in matched pairs is generally small and the corresponding object is relatively far from the camera.

We propose seven criteria for examining the set of SURF pairs for first-person vision image matching. The top image candidate will be accepted as the final answer to the query only when it satisfies all the criteria. Among them, (1) and (5)-(7) are effective criteria for the candidates with a few key pairs.

**(1) Too few pairs**

If there are only two pairs or below, the candidate is rejected.

**(2) Size consistency**

As a SURF key pair should represent same part of something in a scene, the size of the keys should be same if the camera is at the same position. When the camera position is different, their apparent size is changed in inverse proportion to the distance between the object and the camera. As we assume the matched SURF keys are relatively far from the camera comparing with the distance between the objects on which the keys are found, we approximate the normalization process just by utilizing their mean in an image. For an image P, normalized SURF key size $\mathrm{s'_{iP}}$ is given by the original SURF key size $\mathrm{s_{iP}}$ and its average $\overline{\mathrm{s_P}}$:

$$\mathrm{s'_{iP}} = \mathrm{s_{iP}}/\overline{\mathrm{s_P}}$$

Then we define the size difference $\mathrm{E_{size}}$ between two images Q and $\mathrm{R}^*$ by

$$\mathrm{E_{size}} = \frac{1}{\mathrm{K}} \sum_{1 \leq \mathrm{i} \leq \mathrm{K}} \left|\mathrm{s'_{iQ}} - \mathrm{s'_{iR^*}}\right|$$

Note that $\mathrm{E_{size}}$ should be 0.0 for the best match.

**(3) Direction consistency**

Directions of SURF keys of a pair should be same if the rotation of the camera around optical axis is same. Since the first-person vision camera is not so rigidly mounted on the body unfortunately, we need to normalize the camera rotation around the optical axis on comparison. We define the direction difference $\mathrm{E_{dir}}$ by:

$$\mathrm{E_{dir}} = \frac{1}{\mathrm{K}} \sum_{1 \leq \mathrm{i} \leq \mathrm{K}} |\mathrm{d'_{iQ}} - \mathrm{d'_{iR^*}}|$$

$$\mathrm{d'_{iP}} = \mathrm{d_{iP}} - \overline{\mathrm{d_P}}$$

$\mathrm{d_{iP}}$ means the direction of SURF key $\mathrm{i}$ in image P.

**(4) 2D Affine constraint**

As the matched points are assumed to be on rigid objects in a scene, their positions in the two images should follow epi-polar geometry constraint. However, as 1) the $\mathrm{K}$ might be a small number, 2) not a few of them might be liars, and 3) cameras may be different for query and reference image acquisition, it is not practical to utilize the epi-polar constraint directly for verification. Rather we introduce a 2D affine constraint here. A 2D-affine matrix can be obtained by giving three pairs, so it works when more than 3 pairs are found.

Suppose the estimated 2D affine matrix H. The 2D affine difference $\mathrm{E_{affine}}$ is defined by the residuals of the key points. $\mathrm{x_{iP}}$ denotes the location of the key in image P.

$$\mathrm{E_{affine}} = \frac{1}{\mathrm{K}} \sum_{1 \le \mathrm{i} \le \mathrm{K}} |\mathrm{H\ x_{iQ}} - \mathrm{x_{iR^*}}|$$

**(5) Area size**

Sometimes the number K of the matched pairs comes to single (see Table 1 in the experiment section) and the key points may cover very small area of a certain object in a scene. In this case, the object might be a poster or common signs that can be frequently found at different locations in the walking path. Therefore, if all the SURF keys locate in a very small area, it is better to reject the image candidate. We define the indicator $\mathrm{E_{area}}$ by:

$$\mathrm{E_{area}} = \min(\mathrm{area}(\{\mathrm{x_{iQ}}\}), \mathrm{area}(\{\mathrm{x_{iR^*}}\}))$$

A function $\mathrm{area}(\{\mathrm{x_{iP}}\})$ returns the size of area which a set of key points $\{\mathrm{x_{iP}}\}$ covers.

**(6) Axis inversion by 2D affine matrix**

In addition to the 2D affine constraint (4), we also detect axis inversion by investigating H. Suppose u and v axis on one image are projected to $\mathrm{u'}$ and $\mathrm{v'}$ by the 2D affine matrix H. Then the angle between u and $\mathrm{u'}$ should be less than $\pi/2$ because the first-person vision cameras will not twist largely to $\pi/2$, or mirroring should never happen from a query image to reference image.

This inversion can be detected by checking the sign of diagonal elements of H. That means $\mathrm{H\ u} \cdot \mathrm{u} \geq 0.0$ for u and same for v. If at least one of the diagonals is negative, axis inversion occurs and the candidate is rejected.

**(7) Triangular vector direction**

Since not a few queries got only three pairs, we add one more geometric constraint which examines the correctness of the matching with only three pairs. The three key points form a triangle. We assume that corresponding edges between the two images should make an angle less than $\pi/2$ because the first-person vision camera will not rotate so hard.

## 7. Experiment

We have implemented the core part of our first-person vision image retrieval system to evaluate the performance of the proposed method.

We chose a walking path of approximately 900 meters in downtown Tokyo area. It starts around at the underground gate of a railway station and it first goes through a popular underground mall for 300 meters, then going up to the ground level by stairs and an escalator. In next 250 meters, it goes through a pavement at a side of two-lane road, crosses the road, and heads up to the entrance of a big department store. Then, it turns into a narrow street that lasts 250 meters. The last part of the path includes a big crossing. There are always people on the streets, so it is almost impossible to take reference images without interfering objects.

We have conducted the test many times, but show four results for one reference image set.

The video for the source of reference images was taken from 12:15 at a sunny winter weekday, by a Panasonic DMC-FX37 at 28mm focal length. The original video size is VGA and it lasts for about 12 minutes. The reference images are taken by 3.0 fps at a size of QVGA. The number of reference images N is 2,100. Hence the reference points on the path can be considered to be located 0.5 meter interval at the maximum walking speed section. Reference images have 333.1 key points in average, and 699,595 for the total. As we choose 64 dimensions to describe SURF and utilize KD tree approach, the system uses about 220~280 MB memory.

The results of four query videos are shown in Table 1. Video-1 was taken just after the reference video was taken by the same camera. Video-2 was taken 6 days after the reference video, from 13:56 on a weekday. Video-3 was taken 47 days before the reference video, where decoration of the underground mall was different because of season's greeting sale. Video-4 was taken on the same day of reference video, but taken by iPod Nano.

We examined experiment parameters based on preceding tests and set the thresholds to 0.2 for $\mathrm{E_{size}}$, 45.0 degree for $\mathrm{E_{dir}}$, 10.0 pixel for $\mathrm{E_{affine}}$, and 50.0 square pixel for $\mathrm{E_{area}}$. The acceptance ratio of the experiment was ranging from 13.8 to 27.5 %, so a user will receive answers at every 8~4 frames. It is not critical as we think this method will be combined with a pedometer module for total pedestrian navigation system.

Figure 1 shows the verified estimated positions of video-1 by path distance (distance from the starting position). Horizontal axis indicates the position of query, and the vertical axis indicates the estimated position. Ideally, the graph should be in $\mathrm{y} = \mathrm{x}$ shape. The figure clearly shows that the system succeeds in estimating the position well for most of the queries. Note that the Figure 1 is not accurate as the grand truth of the path distance is not obtained. The path distance here is inferred by measuring the passing time of thirty checkpoints and assuming constant walking speed, except for points where waiting for signal at crossings.

Figure 2 shows some snapshots of the queries and results. 1st and 3rd columns are queries, and right next are the corresponding top image candidates. If thick lines are drawn on a candidate image, it means it is rejected due to the proposed criteria. Thin red circles and lines show the matched SURF pairs, and green or blue line is showing 2D affine matrix estimation result. The 1st snapshot was a success example of rejecting the false positive candidate, while 4th and 8th seem to be rejecting true negative candidates. However, as there are some misfit SURF pairs in these pairs (see carefully Figure 2), we can say those rejections are reasonable.

We conducted the experiment on a notebook computer with Intel Core2Duo U9400 (1.4GHz). As shown in Table 1, it is currently not available for video rate speed, but it could be speeded up further. Table 1 also shows SURF key pair distribution in queries. The top candidates with only three pairs take certain portion (actually most frequent) in queries and some of them are true-positive. In Figure 2, 2nd, 7th, and 10th snapshots in Figure 2 have only three key pairs.

## 8. Conclusion

We proposed and examined the new first-person vision image retrieval method for pedestrian navigation in urban area. Further evaluation with ground truth positions and on-line tests are expected.

| Video No. | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| # query | 21030 | 28260 | 24060 | 25110 |
| # verified | 5782 | 3911 | 5854 | 4151 |
| ratio [%] | 27.5 | 13.8 | 24.3 | 16.5 |
| # key | 344.6 | 297.8 | 366.2 | 229.6 |
| surf [msec] | 126.3 | 119.2 | 137.0 | 132.4 |
| query [msec] | 230.8 | 211.8 | 256.9 | 220.9 |
| 0-1 pairs [%] | 0.2 | 0.1 | 0.1 | 1.3 |
| 2 pairs [%] | 4.5 | 12.2 | 12.2 | 23.2 |
| 3 pairs [%] | 20.2 | 36.1 | 36.1 | 26.6 |
| 4 pairs [%] | 16.7 | 18.4 | 18.4 | 10.7 |
| 5 pairs | 10.5 | 10.1 | 10.1 | 6.3 |
| 6-10 pairs | 25.7 | 17.0 | 17.0 | 16.1 |
| 11- pairs | 22.3 | 6.3 | 6.3 | 15.9 |

**Table 1. Query results of 4 videos.**

**References**

[1] R. Fergus, P. Perona, A. Zisserman. A sparse object category model for efficient learning and exhaustive recognition. *CVPR*, 2005.

[2] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. *ICCV*, 2003.

[3] J. Philbin, O. Chum, M. Isard, J. Sivic, A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. *CVPR*, 2007.

[4] H. Kang, A. Efros, M. Hebert, T. Kanade. Image matching in large scale indoor environment. *1st workshop on Egocentric Vision*, 2009.

[5] H. Bay, A. Ess, T. Tuytelaars, L. V. Gool. SURF: Speeded up robust features. *CVIU*, vol. 110, no. 3, pp. 346-359, 2008.

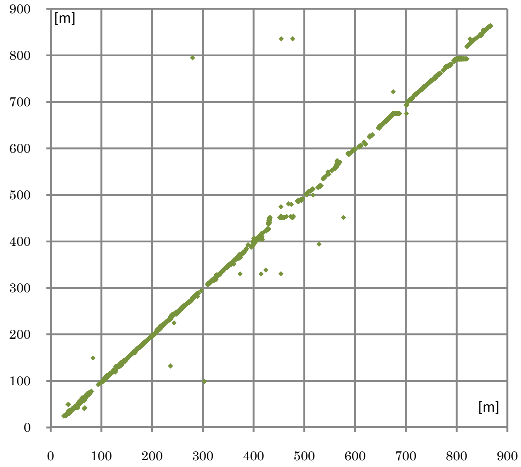

**Figure 1. Query results of Video-1 by path distance.**

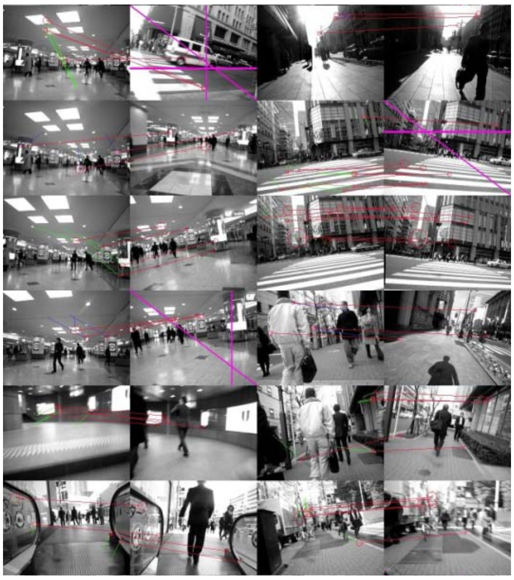

**Figure 2. Some query results of Video-1.**