Floor Fingerprint Verification Using a Gravity-Aware Smartphone

Satoru Fujita^{a,*}, Tomoko Fujita^b, Kaoru Uchida^a

^aHosei University, 3-7-2 Kajino-cho, Koganei Tokyo 184-8586 Japan

^bYokohama National University, 79-1 Tokiwadai, Hodogaya-ku, Yokohama Kanagawa 240-8501 Japan

*Corresponding Author: fujita_s@hosei.ac.jp

Abstract

This paper presents our research on location estimation based on the identity of floor surface patterns, which we call “floor fingerprints,” from a photographic image of a floor taken with a hand-held smartphone. Because floor textures generally appear to lack sufficient features, it may seem difficult using general feature detection algorithms to find matching pairs of features in two corresponding floor images taken at an identical location but from different orientations of the camera and under different lighting conditions. We demonstrate, however, that use of a preprocessing image filter, involving gravity-rectified image adjustment against perspective distortion and enhancement of local image features, provides well-aligned detail sufficient to allow detection of paired features of floor textures. Although the enhancement filter reveals many noisy pairs in local feature detection, we show that it is possible to choose a valid image-to-image correspondence efficiently using our newly proposed B-ORB feature detector and RANSAC. Since matching a query floor image with large-scale floor images stored at the server requires a large amount of processing resources, we utilize GPGPU for the feature detection and matching. This paper proves the feasibility and efficiency of the proposed approach, based on our experimental results concerning the accuracy and processing time, and discusses possible solutions to a wide range of real-world indoor location applications.

Keywords: location estimation, feature detection, feature matching, smartphone application.

1. Introduction

Technology for accurately identifying a person's location is essential in various location-based services such as route-finding and real-time information provision. In spite

of various efforts made in this area, conventional solutions have been practically unreliable and/or expensive, especially in the case of indoor locations where the global positioning system (GPS) is not available. Triangulation-based location positioning using measurement of wireless wave power from access points is one candidate, but the infrastructure cost for setting up a large number of access points is impractically expensive. Pedestrian dead reckoning, called PDR,⁽¹⁾ using micro-sensors on smartphones and simultaneous localization and mapping, called SLAM,⁽²⁾ are other candidates, but the accuracy is insufficient for actual use.

Another approach is found in the area of architectural design, where special markers are pre-embedded in floor materials as texture patterns⁽³⁾. Floor photo images are analyzed to detect the special texture patterns, and the exact location is determined. Although such use of prepared and artificial means is useful in specially designed areas, it is not applicable to general spaces.

In the other topics, artifact-metrics, a technology for identifying and authenticating manufactured products based on measurable intrinsic, often visual, characteristics, such as microscopic random patterns created during the manufacturing process, is increasingly gaining research interest, and its range of applications is widening. As an example, it can verify a product's authenticity by capturing an image of its unique microscopic surface pattern as “fingerprint” and comparing it with the database. This identification technology is sufficiently reliable and actually much cheaper than those using physical ID tags⁽⁴⁾⁽⁵⁾.

In this paper, we apply the above-mentioned identification ideas to general location estimation. We show that it can estimate the user location accurately enough to offer local information services by capturing the unique surface pattern of a floor without any intentional markers and by discovering its match in the floor fingerprint database. As

(a) Original floor image (b) Enhanced & zoomed image

Fig. 1. An example of floor fingerprint.

a preliminary experiment, we tried one-to-one matching tests with two images, where we proved that we could find matching point pairs fairly well in many cases. We name this approach “floor fingerprinting”. Our hypothesis is that every floor surface has a unique fingerprint usable for individuality identification, and we can apply this feature to user location estimation. We first introduce the related planar detection mechanisms, then describe the basic idea of the floor fingerprinting technology and propose methods for filtering photo images, extracting the floor fingerprint from the images and matching fingerprints, and then demonstrate the feasibility of our approach by presenting our experimental results from our research work.

2. Planar Object Detection

Fig. 1 shows an example of a floor surface image and an enhanced floor fingerprint image using our proposed filter. Although we can observe almost nothing special in the image area in the red rectangle of Fig. 1(a), we actually can detect a considerable number of microscopic patterns if we zoom up the image and strengthen its contrast as is shown in Fig. 1(b). These microscopic patterns are unique to the floor, because they are configured with grime acquired afterwards even if the floor is printed as an industrial product.

Verifying floor fingerprints is a special case of matching corresponding planar objects in two perspective images. The most popular way to detect matching pairs of planes is to find a proper homography transformation matrix. Vincent et al. explored the basic theory of projective transformations using 3×3 homography matrix, and a way to find the matrix efficiently⁽⁶⁾. A general idea is that, if $\mathbf{x}$ and $\mathbf{x}'$ are homogeneous local coordinates of images in the same world from different viewpoints belonging to a plane, they are related by homography matrix $\mathbf{H}$ corresponding to the plane:

$$\mathbf{x}' = \mathbf{H}\mathbf{x} \quad (1)$$

$\mathbf{H}$ is a 3×3 matrix and, as the bottom-right element should be fixed to 1, it has 8 degrees of freedom and 4 point

correspondences can determine a single $\mathbf{H}$.

Local feature detectors and matching algorithms, such as ORB⁽⁷⁾ or AKAZE⁽⁸⁾, detect many matching pairs of local features, but they also include erroneous pairs. Therefore, schemes for selecting correct sets of pairs are important to determine feasible matrix $\mathbf{H}$. A random sample consensus, called RANSAC, is a good candidate of selecting correct data from a data pool involving many noisy and irrelevant data. Assumed that a data pool $\mathbf{P}$ has correct data with a probability of w , and a problem requires t independent data to generate a model, then the average number of iterations $E(k)$ and its standard deviation $SD(k)$ for identifying a correct set of t data from $\mathbf{P}$, are follows⁽⁹⁾.

$$E(k) = \frac{1}{w^t} \quad (2)$$

$$SD(k) = \frac{\sqrt{1 - w^t}}{w^t} \quad (3)$$

In order to determine the homography matrix, we need 4 correspondences of matching pairs, thus $t = 4$. If we assume $w = 0.5$, the expected number of iterations for finding correct correspondences is 16, and its standard deviation is also close to 16. Then it is highly probable that we can discover a correct set by randomly choosing 4 matching pairs over 32 times. After the discovery, we can verify its geographical consistency by transforming other local feature points with the obtained homography matrix, and deciding if the number of matching pairs supporting the matrix is more than a pre-defined threshold.

When w becomes smaller, the necessary expected trial count becomes larger in proportion to the inverse of w powered by 4. Vincent introduced how to reduce the total number of pair selection using ad-hoc rules, such as choosing pairs located far from each other⁽⁶⁾.

On the other hand, Kurtz et al. reported a way to reduce the t parameter. While the homography matrix has 8 degrees of freedom and needs 4 matching pairs for determination, the affine matrix only needs 3 pairs. In order to reduce t , they used a gravity sensor that measured a gravity vector with the local coordinate of the device, determined the homography matrix $\mathbf{H}$ from the gravity, and rectified perspective floor images into non-perspective ones at preprocessing time. They named this scheme as gravity-rectified feature descriptors, called GREFD⁽¹⁰⁾⁽¹¹⁾. When the gravity vector is represented as $\mathbf{g} = (g_x, g_y, g_z)^T$ in the local coordinate, it corresponds to $\mathbf{z} = (0, 0, 1)^T$ in the global coordinate. Then, let $\mathbf{g}_1$ and $\mathbf{g}_2$ be two vectors defining a plane with a normal $\mathbf{g}$. These two vectors can be defined as follows:

$$\mathbf{g}_1 = (-g_z, 0, g_x)^T \quad (4)$$

$$\mathbf{g}_2 = \mathbf{g} \times \mathbf{g}_1 \quad (5)$$

Then the (3×3) matrix $\mathbf{H}$ is defined as

$$\mathbf{H} = (\mathbf{g}_1 \ \mathbf{g}_2 \ \mathbf{z})^T \quad (6)$$

After calculating $\mathbf{H}$, they applied camera intrinsic parameter $\mathbf{K}$ to $\mathbf{H}$. Here, the most important thing is that the matrix $\mathbf{H}$ can be calculated by using gravity $\mathbf{g}$. Our definition is slightly different from their model, but the essential concept is common with them, as described in the next section. They succeeded in warping a floor image into ones in an orthogonal coordinate and comparing it with a corresponding flat floor image using RANSAC with a parameter $t = 3$. This scheme is not applicable to any planes except for the horizontal plane, but it is very efficient when we are to deal with floor images.

3. Floor Fingerprint Verification

3.1 Overview of the Proposed Mechanism

Floor images, in general, show very few apparently distinctive local features and repeat similar small and weak patterns generated by grimes and basis materials, like wood and sand grains. Additionally, smartphone photo images are taken with various postures, such as scales, horizontal rotations, and vertical angles. We need a verification mechanism to find geographically consistent matching images from a large floor-image database efficiently and correctly under such conditions.

The general approaches to detect planar objects in images, which we explained in the previous section, are known to be effective when suitable number of explicit local features are available, such as corners and edges of the objects or black/white checker patterns painted on the floor. In these cases, the probability of the correct correspondences of the matching pairs, namely w in Eq. (2), is assumed to be in a range from 0.3 to 0.5. However, the floor images in real cases do not show sufficient number of local-interest points. So far as we know, the probability w is less than 0.03 in many cases, even after we enhance the image features, as we will describe in this section. Because it is technically very difficult to detect the explicit patterns in the floor images, we propose a set of new schema described in the successive subsections.

3.2 Image Capturing

Floor surface pictures are taken with a smartphone, which is held as horizontally as possible. Because our initial

experiments showed that it was difficult for users to keep smartphones correctly in a horizontal position, we propose a new scheme in which the gravity information of the device is stored in an attached file simultaneously with each image and later used to estimate its pose. Currently, most smartphones are equipped with an accelerometer accurate enough to make this possible.

3.3 Preprocessing

The first task of the preprocessing is to rectify the perspective images into those in the orthogonal coordinate using the gravity information. As described in the previous section, the idea is similar to GREFD that calculates the homography matrix and warps the images to those from the vertical viewpoint. We, first, rewrite the basis vectors in the local coordinate into ones in the global coordinate. When the gravity vector, which is the direction vector parallel to the z -axis in the global coordinate, is given as $\mathbf{g} = (g_x, g_y, g_z)^T$, this z -axis basis vector $\mathbf{g}'_3$ in the local coordinate is expressed as:

$$\mathbf{g}'_3 = (g_x/G, g_y/G, g_z/G)^T \quad (7)$$

where $G = \sqrt{g_x^2 + g_y^2 + g_z^2}$

$\mathbf{g}'_1$, which is the x -axis basis vector, is not uniquely determined in the local coordinate, but if it is assumed that its y element is zero in the local coordinate, it is expressed as:

$$\mathbf{g}'_1 = (g_z/G', 0, -g_x/G')^T \quad (8)$$

where $G' = \sqrt{g_x^2 + g_z^2}$

Finally, $\mathbf{g}'_2$, which is the y -axis basis vector, is calculated as the external product of $\mathbf{g}'_3 \times \mathbf{g}'_1$:

$$\mathbf{g}'_2 = \mathbf{g}'_3 \times \mathbf{g}'_1 = (-g_x g_y / G' G, G' / G, -g_y g_z / G' G)^T \quad (9)$$

Therefore, homography matrix $\mathbf{H}'$ from the global coordinate to the local coordinate is described in the following equation:

$$\mathbf{H}' = (\mathbf{g}'_1 \ \mathbf{g}'_2 \ \mathbf{g}'_3) = \begin{pmatrix} g_z/G' & -g_x g_y / G' G & g_x/G \\ \mathbf{0} & G'/G & g_y/G \\ -g_x/G' & -g_y g_z / G' G & g_z/G \end{pmatrix} \quad (10)$$

This homography matrix is not good enough to warp the perspective images, since the center of the image is shifted to the point straight down from the viewpoint, and the scale of the rectified image is different from the original one. In the worst case, no result is obtained. This problem can be solved by multiplying it by the affine matrixes with parameters of x -resolutions(r_x) and y -resolutions(r_y) of the image from its left and right.

$$\mathbf{H}'' = \mathbf{LH}'\mathbf{R} \quad (11)$$

(a) Original image (b) Rectified image

Fig. 2. Transformation based on gravity.

$$\mathbf{R} = \begin{pmatrix} \frac{l_x}{r_x} & 0 & -\frac{l_x}{2} \\ 0 & \frac{l_x}{r_x} & -\frac{l_x r_y}{2r_x} \\ 0 & 0 & 1 \end{pmatrix} \quad (12)$$

$$\mathbf{L} = \begin{pmatrix} \frac{r_x}{l_x} & 0 & \frac{r_x}{2} - \frac{hr_x}{l_x} \frac{g_x}{g_z} \\ 0 & \frac{r_x}{l_x} & \frac{r_y}{2} - \frac{hr_x}{l_x} \frac{g_y}{g_z} \\ 0 & 0 & 1 \end{pmatrix} \quad (13)$$

In Eq. (13), h is a camera height from the floor, and l_x is the real length of the long edge of the pictures in meters. For example, h is 1.3 meter and l_x 1.4 meter. To adjust the (3, 3) element in homography matrix to 1, all elements are divided by the value of original (3, 3) element. If reverse transformation from $\mathbf{x}'$ to $\mathbf{x}$ is required, the inverse of $\mathbf{H}''$ is to be calculated and used.

Fig. 2 shows an image example rectified using this schema. Fig. 2(a) is the original perspective image, with its gravity vector (0.15 8.67 4.58). The pitch from the horizontal position is 1.09 radian. Fig. 2(b) shows the rectified image, in which the edges of corridor is rectified to be almost parallel while keeping the common center of the image, when h is roughly determined as 1.3 meter.

The second task of the preprocessing is to enhance local-interest features of the floor surface image. As shown in Fig. 1, the original color image include very few explicit edges or corners, and even if they exist on the image, they are weak and affected by some reflection of light. In order to deal with such conditions, we adopt a mean difference filter that calculates the brightness deviation of each pixel from the mean brightness of the neighborhood. Both the grayscale output and the binarized output are prepared. As for neighborhood, the $40 \times 40 = 1600$ pixel rectangular area is used.

3.4 Feature Extraction

As for technological approaches to image registration, which is the fundamental image processing task necessary

(a) Extraction by ORB (b) Extraction by B-ORB

Fig. 3. Distribution of extracted features.

for image discovery, we adopt a keypoint-based approach that is generally known to be more robust against deformations due to difference in sensors, viewpoints, lighting conditions and appearances caused by the changes in surface patterns over time. Especially, we use ORB, which is one of the strongest detectors for rotation, scaling and deformation⁽⁶⁾. ORB is also convenient to find a specified number of keypoints selected from keypoint candidates by using Harris corner detection as the criteria of strength for the keypoints. However, as shown in Fig. 3(a), the keypoints of the floor image, marked as small circles, are concentrated in comparatively smaller areas, such as brighter parts and tile edges. Although such uneven distribution in the result is reasonable, because the purpose of the general feature detectors is to find the strong corners and edges, it is not useful for detecting sufficient number of weaker features from the floor surface images. Therefore, we modified the sorting algorithm in ORB to extract keypoints evenly over the floor surface. We called it as balanced ORB or B-ORB. Since ORB holds images and keypoints in a pyramid structure each level of which has different resolution, we define the modified mechanism for sorting and selecting keypoint candidates as follows.

- (1) As for each pyramid layer of the resolutions, the upper limit number of keypoints to be extracted is loosened to be twice as the original limit.
- (2) The keypoint candidates in each layer are divided into smaller grid regions, in each of which stronger candidates are chosen to the upper limit number divided by the number of regions.
- (3) All candidates from all layers are aggregated into a single bundle, and then selected a set of stronger keypoints under the criteria of Harris corner detection.

Using this mechanism, the resultant keypoints are widely distributed, even if the stronger keypoints are concentrated in smaller regions. When we set the number of grid regions between 4 and 10, we obtain improved results as shown in Fig. 3(b).

3.5 Feature Matching

The images taken at the same location with different postures are presumed to have comparatively larger number of detected local keypoint feature pairs that can be matched after a common transformation. Based on this assumption, general feature matching mechanism is applied to find matching pairs between two images. We adopted brute-force matcher implemented for GPGPU in OpenCV, because of its good performance⁽¹²⁾. General matcher can find many pairs with similar local keypoints, but the requirement in floor image matching is completely different, since most of all features are very weak and, to make matters worse, there are many similar small patterns on the floor. As a result, most of calculated matching pairs are incorrect. Actually, our preliminary experiments showed that only 2% of the matching pairs were correct. This corresponds to $w = 0.02$ in Eq.(2), requiring an inhibitive large number of random selections to find correct pairs using RANSAC. As we discussed in the explanation of DREFD, we need to select 4 pairs of points for calculating homography matrix, but 3 pairs for affine matrix. This difference of $t = 4$ and $t = 3$ produces a large difference in computational cost. Moreover, in our case, affine matrix is restricted to rotation, scaling, and two values of translations, then the matrix can be determined by 2 pairs, allowing the third pair to be used for validating the consistency of the matrix, which makes the computational cost much smaller. Our proposed algorithm for finding correct matching pairs and determining if two images are identical, can be summarized by referring to Fig. 4 as follows:

- (1) Select two matching pairs randomly and calculate affine matrix with rotation, scaling and translation, as shown by the yellow lines in Fig. 4.
- (2) Select the third matching pair randomly and transform the point in the original image to the target image using the calculated affine matrix, as shown by the star marks in Fig. 4.
- (3) If the third matching pair can be consistently mapped and the distance between a transformed point and the corresponding point in the target image is smaller than the predefined threshold, then proceed on farther check if the affine matrix can consistently map other matching pairs.
- (4) If the number of pairs supporting the affine matrix is larger than its threshold, two images are determined as matched. Otherwise, go back to the step (1) and repeat up to the maximum loop limit.

Fig. 4. Matching with affine transformation.

Fig. 5. Successful matching result.

- (5) If the prefixed number of loops are tried without success, two images are determined as different.

Fig. 5 demonstrates a successful matching result. The small circles are the extracted local keypoints. As we can see, a considerable number of local feature points are extracted, not only along the tile edge lines, but also on the surface plains. The lines between the left and right images indicate the correspondences between the successful matching point pairs.

3.6 Match Discovery

To estimate a user's exact location using a floor fingerprint, we need to find a match with a given query image in a large-scale image database containing a collection of floor fingerprint images. Here, the main challenge is reducing the computational cost and memory usage against the size problem. Although we have not fully addressed this scalability problem yet, we apply a cache mechanism to improve the total performance. The local keypoint features of all images in the database are precomputed for caching. When a query image is given to the system for discovery, only the descriptors of the precomputed features are loaded, but not the original images. This saves the amount of computation necessary for the floor fingerprint verification.

Table 1. Matching result and performance.

	Wood-type	Stone-type	Linoleum
Detect Pairs	3126	3776	2347
Matched Pairs	582.3	49.0	36.5
Loading (ms)	388	428	401
Extraction (ms)	144	140	89
Matching (ms)	178	179	154
Verification (ms)	0	22	77
Validation (ms)	8	3	3
Total (ms)	689	773	726

4. Experiments

4.1 Floor Image Matching

The base one-to-one matching feasibility experiments for floor fingerprints are conducted using floor images of wood-type, stone-type, and linoleum (plastic-type). For each floor type, we prepare multiple images, such as images taken with different rotations, under different lighting conditions at different occasions, and using different smartphone cameras, but we keep the surface of the smartphone horizontally to preclude perspective distortions. Floor pictures are taken with Xperia Z3 and Nexus 6P, both with resolutions of around 4000x3000, and then down sampled to 1/4 sizes in length. The computations were conducted on a laptop PC with Core i7-4710MQ 2.50 Hz and GeForce GTX 950M.

Table 1 shows identification test results for one-to-one images. Multiple image examples are tested for each type, and the average of candidate pairs detected by the general matcher (“Detected”) and of correct pairs whose total geometrical locations are consistent with each other (“Matched”) are shown. While wood-type showed good results, because wood grains clearly appeared in the images, linoleum tiles failed to be matched in some cases, because of lack of features.

For all types, the execution times were less than 800 ms, most of which were spent on loading and filtering two images. Extraction and matching times are respectively spent on B-ORB and the brute-force matcher. They counted for 30 to 50 % amount of time. Verification time spent for finding correct matching pairs was less than 100ms. The verification time for wood-type floors was 0, meaning that RANSAC found the correct correspondences of matching pairs without any overhead.

4.2 Matching Performance vs. Image Resolutions

Fig. 6. Resolution and success rate.

Fig. 7. Resolution and execution time.

The next experiments examine the relationship among image resolutions, the numbers of extracted features and execution times, using 76 pairs of images. Fig. 6 shows the plot of matching success rate. A value n in x-axis means that a $1/n$ down-sampled image was used. The original images, represented by ‘ $n=1$ ’, are expected to produce the best success rate, but this expectation was incorrect, especially when there were small number of extracted features. We guess the reason is that high-resolution images include more features, but the valid corresponding features on the opposite image are sometimes removed due to the upper limit on the extracted feature counts, lowering the success rate. As a result, case with one fourth of the resolution, such as 1000x750 pixels, achieved the highest rate in this experiment. On the other hand, Fig. 7 shows that the execution time drastically increases in accordance with the resolution. The balance of the resolution and execution cost is an important factor in the floor fingerprint verification.

4.3 Gravity-Based Rectification

The next experiment evaluates the effects of rectification of perspective distortion using gravity sensors. Fig. 8 shows the relations between the pitch angles (in radian) of the smartphone and the correct feature ratios. The

Fig. 8. Correct rates for rectified images.

features are counted as correct if geographically corresponding points have similar feature vectors, that is, if more than 156 feature bits are identical in 256 total bits of the ORB descriptor. The original images are marked as circles, and the rectified images are marked as crossings.

In most cases, the rectified images have better correct ratios than the original ones. Especially, the gravity approach transforms road surfaces very well, while the original images with larger angles tend to fail. Carpet relatively have lower correct ratios than the others, since the surfaces are configured in 3 dimensional structures, then local features show many differences between the vertical view and the skewed view. In cases of lower angles than 0.1 radian, the rectified images sometimes have lower correct ratios. This is because the transformation produces additional distortions that change the configuration of local-interest features.

4.4 Feasibility of Location Estimation

We conducted experiments to evaluate the accuracy of the location estimation using floor images taken in the corridors of our campus building. We prepared the test set of 116 photos within a rectangular corridor floor area of 2m by 88m, taken with a smartphone hand-held horizontally. The floor is made of flat, stone-type tiles with gravel texture. These are used as reference images.

We prepared another set of 27 query images, using a rotated and/or tilted smartphones (Nexus 6P). The image resolution is 4000x2992. The environmental conditions of the images are not good, as the ceiling light or outdoor light are reflected. The query image, taken from different angle of the same area, has different view of ceiling light reflections, which is expected to make the identification more challenging. In contrast to our initial presumption, however, the result was very good without errors. We could correctly

identify the locations of the photos in all query cases. The total execution time for matching all images in database was 56.6 sec on average. It corresponds to 0.488 sec for one-to-one matching.

We also tried linoleum floors, but the success rate was relatively low, approximately 60%. While stone-type floor had clear and distinctive features on its surface, texture on linoleum surface was less apparent, which we consider the reason for the low success rate.

5. Discussion

We have outlined our proposed approach toward user's location estimation based on the floor fingerprint obtained from the floor texture image captured with the user's smartphone, and reported the results from our preliminary experiments.

As we have stated in the Section 1, there have been reported works related to self-localization of autonomous robots based on pattern recognition of fine-grain chips mixed into floor materials⁽³⁾. This approach makes use of intentionally embedded markers, and assumes relatively stable and known image capturing conditions by a camera installed on a robot. In matching, they use Polestar Algorithm, based on solely the mutual distance relationships among marker points. On the other hand, our approach make use of fine-grained floor texture naturally generated in the manufacturing process, rather than intentionally embedded markers. Appearance of this kind of naturally generated fine-grained floor texture is easily affected by environmental conditions such as lighting. We, therefore, make use of ORB local feature descriptors to extract the inter-point relations as features and search the image based on RANSAC, to attain high-degree accuracy. As standard ORB detects stronger features distributed unevenly, we needed to balance the distribution suitable to the floor pattern detection.

Gravity based image transformation works very well in the experiments. In addition to Kurz's research, we presented how to keep the center and the scale of the original images after warping. When the image is taken from relatively low angle, the correct match ratios tend to decreases due to image distortions. We have seen that our gravity based image transformation works well in most of the cases with pitch angles over 0.1 radian.

For application of floor fingerprint technology to practical location estimation, we need to compare the query image with a large number of reference images in the database to find a match. In our experiments using 116

reference images, the whole identification spent one minute, although we have already used GPGPUs as possible. Still, however, as it takes one minute in the current implementation, we need to take more effective measures of acceleration, including the external resources such as cluster computing and cloud computing, to carry out advanced, practicality experiments in the real field such as in a gigantic shopping malls.

As for the identification accuracy, there are on the other hand difficult floor types, such as linoleum tiles. Our finding is that the search for good sets of features and parameters, and also appropriate preprocessing filters that can effectively enhance features, is an important research target.

6. Conclusions

This paper presented our research on "floor fingerprinting" technology which finds the identity of a floor surface based on its textures, and its application to location estimation based on a photo image of the floor. We have shown that, although the floor textures generally appear to have insufficient amount of features, the average deviation filter works effectively to enhance their features. When the image is distorted by the perspective view, we have shown the gravity based transformation can rectify the distortion sufficiently well. We have proposed an identification method in which we first conduct ORB-based image matching, then we find an image-to-image correspondence using RANSAC. We have demonstrated, through our experimental results that we can successfully achieve identification of wood-type, stone-type, and linoleum floors, but we also observed an innegligible number of failure cases especially with linoleum floors. As applicability to location estimation, while achieved no errors with stone-type floors, we had only 60% success rate with linoleum floors, which is our future work. We plan to continue our work to improve the processing speed with such methods as the use of external parallel computing resources, in order to make this applicable to real-time location estimation.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number JP17K00138.

References

- (1) Jimenez, A. R, Seco, F., Prieto, C. and Guevara, J.: "A comparison of pedestrian dead-reckoning algorithms using a low-cost MEMS IMU," IEEE International Symposium on Intelligent Signal Processing, WISP 2009, pp. 37-42, 2009.
- (2) Frese, U., Wagner, R. and Roefer, T.: "A SLAM Overview from a User's Perspective," KI-Kuenstliche Intelligenz, Vol. 24, No. 3, pp. 191-198, 2010.
- (3) Murasawa, K., Fukase, Y., Takemoto, Y. and Kimura, S.: "Localization System with Dots Pattern on Floor (Part II) (in Japanese)", the 27th Annual Conference of the robotics society of Japan, pp. 1F3-06, 2009.
- (4) Ishiyama, R., Nakamura, Y. Monden, A., Huang, L. and Yoshimoto, S.: "Melon Authentication by Agri-biometrics - Identifying Individual Fruits using a Single Image of Rind Pattern," VISAPP (1), pp. 698-704, SciTePress, 2012.
- (5) Takahashi, T. and Ishiyama, R.: "FIBAR: Fingerprint Imaging by Binary Angular Reflection for Individual Identification of Metal Parts," Proc. of the 5th Int. Conf. on Emerging Security Technologies, pp. 46-51, 2014.
- (6) Vincent, E. and Laganieri, R.: "Detecting Planar Homographies in an Image Pair," Proceedings of the 2nd International Symposium on Image and Signal Processing and Analysis, pp. 182-187, 2001.
- (7) Rublee, E., Rabaud, V., Konolige, K. and Bra, G.: "ORB: An Efficient Alternative to SIFT or SURF," International Conference on Computer Vision, 2011.
- (8) Alcantarilla, P., Nuevo, J. and Bartoli, A.: "Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces," Proceedings of the British Machine Vision Conference, 2013.
- (9) Martin A. F. and Robert C. B.: "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," Commun. ACM, Vol. 24, No. 6, pp. 381-395, 1981.
- (10) Kurz, D. and Benhimane, S.: "Gravity-Aare Handheld Augmented Reality," 10th IEEE International Symposium on Mixed and Augmented Reality (ISMAR), 2011.
- (11) Kurz, D. and Benhimane, S.: "Handheld Augmented Reality involving gravity measurements," Computers & Graphics, Vol. 36, Issue 7, pp. 866-883, 2012.
- (12) OpenCV libraty, <http://opencv.org/>, reffered on May 29, 2017.