TVRSJ Vol.22 No.2 pp.177-187, 2017

基礎論文 Paper

Camera Localization under a Variable Lighting Environment using Parametric Feature Database based on Lighting Simulation

Tomohiro Mashita[*1], Alexander Plopski[*2], Akira Kudo[*3],
Tobias Höllerer[*4], Kiyoshi Kiyokawa[*1], Haruo Takemura[*1]

Abstract – **Localizing the user by using a feature database of a scene is a basic and necessary step for presentation of localized augmented reality (AR) content. Due to the time and effort in preparing such a database, only a single appearance of the scene is commonly stored. The appearance depends on various factors, e.g., position of the sun and cloudiness. Observing the scene under different lighting conditions results in a decrease in the success rate and the accuracy of localization.**
To address these problems, we propose to generate a feature database from the simulated appearance of the scene model under different lighting conditions. We also propose to extend the feature descriptors used in the localization with a parametric representation of their changes under varying lighting conditions. We compare our method with the standard representation and matching based on L_2-norm in a simulation and real-world experiments. Our results show that our simulated environment is a satisfactory representation of the scene's appearance and improves feature matching from a single database. The proposed feature descriptor achieves a higher localization ratio with fewer feature points and a lower processing cost.

Keywords : **augmented reality, camera localization, feature descriptor, lighting simulation**

1 Introduction

Augmented Reality (AR) content is commonly spatially registered relative to a reference target. Although fiducial markers have been commonly used in AR applications, over the past decade vision-based localization and tracking algorithms have shifted towards marker-less environments. Localization refers to initial pose estimation and tracking to estimate the user pose in a continuous stream of information. Tracking of the camera has been mostly solved over the years with robust algorithms based on sparse 3D features [7, 15], depth-sensing cameras [6], and dense [16] and semi-dense [18] reconstruction of the environment. However, even the best tracking algorithm is useless if the initial localization is incorrect.

To estimate the user's pose, state-of-the-art mobile devices are equipped with a variety of sensors such as a camera, compass, gyroscope, accelerometer, and GPS sensor. However, the raw data provided by such sensors are not accurate enough for user localization. For example, the error in the position estimated from a GPS sensor is commonly more than 1 m. Visual search and matching algorithms are therefore employed to further refine the information provided by the localization sensors. A typical vision-based localization algorithm estimates the camera's 3D position and orientation by matching values in a database that typically stores sets of 3D points and feature vectors in an image. Namely, the feature points that most closely match those in a query image are searched in the database, and then the camera position and orientation are calculated from the corresponding 3D points of the matched feature points.

Mobile devices have only limited computational resources and bandwidth. Therefore, localization is performed against a database of feature vectors that describe the appearance of an environment. Such a database describes the static appearance of the scene and cannot account for large variations in appear-

[*1]Osaka University
[*2]Nara Institute of Science and Technology
[*3]Hitachi, Ltd.
[*4]University of California Santa Barbara

ance due to changing lighting effects, e.g., very different sun positions, cloudiness outdoors, and indoor illumination. The accuracy and the rate of localization decrease with the changing appearance of the features.

Creating databases that are capable of addressing such changes is a tedious process, because one has to not only determine the necessary subset but also record the representative data. Depending on the target environment and the variety of observable variations, the resulting database might become very large, which increases the time needed to match an image against it and requires months of recording.

In this paper we address the abovementioned problems through a dual approach. We propose foregoing the repetitive data acquisition in favor of simulating the appearance of a scene under varying lighting conditions. These conditions (outdoors and indoors) are known, because only a discrete set of light origins and degrees of cloudiness is possible. We also propose to match features based on the Mahalanobis distance, instead of the commonly used L_2 distance, to better represent feature vector changes under different illumination conditions. This dual approach is an application of the pattern classification scheme in feature matching. The data acquisition by simulation provides a correct association between the 3D point and the feature point in an image. That is, whereas the commonly used L_2 matching is a simple nearest-neighbor method, our method parametrically represents the variation of appearance in feature space.

The main contributions of our paper are

1. Instead of recording the appearance of the target scene under various lighting conditions, we generate the database through rendering of the scene under virtual illumination conditions.

2. We compare our method to a standard localization approach and show that it can achieve a better accuracy rate with fewer features.

2 Related Works

The contributions of our paper are primarily related to camera localization and feature descriptors.

2.1 Outdoor Camera Localization

Traditional camera localization uses artificial markers, which have been rigidly installed into the environment and whose position has been calibrated beforehand [17].

Ventura et al. [22] proposed localization as a part of simultaneous localization and mapping (SLAM)-based tracking. The first two keyframes of the tracking are uploaded to a server, which determines the respective 7-DOF transformation from the local to the geo-located model. The SLAM tracking is updated with the retrieved information and further keyframes are used for pose refinement.

Kurz et al. [10] targeted environments with many repetitive features, such as windows in a façade. To limit the number of false positive matches, they limited the number of features that are matched against. The authors determined the initial 3D position of the feature by intersecting its backprojection with the scene model, where the pose is given from the sensors. The feature is then matched only to database features whose positions are within the proximity of the reconstructed 3D position. The authors reported that their method achieved higher accuracy than naïve feature matching and orientation-aware feature matching [1]. Additionally, their approach greatly reduces the number of descriptor comparisons required in the matching step.

Arth et al. [2] used machine learning to detect facades in the taken image. The user is localized through matching of the extracted facades with a 3D map of the surroundings. They reported that their method usually achieved localization errors within the range of 1-4 m and orientation errors of less than 3°. However, because their method requires prior sensor information and at least two visible facades, it cannot be easily applied indoors or in scenes where these requirements are not met.

The appearance of the outdoor environment varies across seasons. Naseer et al. [14] proposed a SLAM method that is robust for seasonal variations. Their method estimates the similarity between two images on the same route in different seasons and optimizes a network between two sequences defined by the similarity and matching hypotheses. Additionally, as discussed in this paper, long-term visual localization is one of the difficult challenges due to natural or man-made changes.

2.2 Feature Descriptor

Over the past years a variety of descriptors have been developed to provide an efficient way to represent and compare detected features.

SIFT [12] and SURF [3] descriptors of detected corners have proven to be robust against orientation, scale and partial illumination changes. These descriptors have also been applicable in a variety of localization [5, 21] and tracking [7] solutions. With the rise of mobile computing, modified descriptors that include additional sensor information have been shown to improve matching results and to reduce the number of comparisons needed to match a feature with a prerecorded database. Kurz et al. [8] proposed gravity-aligned feature descriptors (GAFD), where the gravity vector of the hand-held device helps distinguish between similar features with different global orientations, such as the corners of a window. Kurz et al. [10] used the scale of the feature that was retrieved from a known model to reduce the number of features to be matched against.

Our work follows the above studies in that an extension of the commonly used features is applied to further improve the robustness of the matching. However, it differs from the previous studies in that the extension is based on the variance of the feature's appearance instead of additional sensor information.

2.3 Database Acquisition

To evaluate localization methods, researchers have proposed and developed various methods to generate ground-truth information as well as to acquire a representative feature database.

Ventura et al. [21] reconstructed the surroundings through structure-from-motion, and manually set the position, scale and orientation of the reconstruction. They used all reconstructed points to localize the user from images taken by an omni-directional camera. Similarly, Irschara et al. [5] reconstructed a point-cloud model of a scene from a large image database. Additionally, they generated virtual views of the scene and keep the smallest subset that covers the targeted viewing area.

Kurz et al. [10] used a laser scanner to recover a dense point-cloud representation of the environment. By projecting the recovered model into virtual cameras distributed throughout the scene, they generated virtual views of the scene. They recovered a representative feature subset according to the method of [9].

Shinozuka et al. [20] proposed a feature table method that has keypoint variations caused by highlights and viewpoint changes. Although the keypoints variations stored in the table were generated from a 3D model, the tracking accuracy was improved.

A feature vector calculated from a synthesized image has some differences due to the difference between a real and a virtual environment. Simon [19] lessened the depth blur issue by adding depth blur to rendered images.

As has been discussed by several authors [10][5][19] and [20], artificial variations simulated by computer graphics improve the performance of camera localization. Our method resembles these studies in that simulated dense 3D models are used to generate the feature database. Unlike previous studies, we apply lighting variations to simulate the appearance of a scene under varying illumination conditions. Moreover, we generate a compact parametric feature database that records the statistical parameters of the feature vector distribution instead of all features from various conditions.

3 Feature Matching with Simulation-based Database and Mahalanobis Distance

In this section we describe in detail the main contributions of our paper, which are a feature matching methodology for databases that store multiple feature vectors of the same reference point, namely a 3D point in a scene, and a scheme for acquisition of feature vectors under varying lighting conditions and viewpoints. To synthesize a scene under varying illumination, our proposed method requires the 3D model, texture, and global position of the scene.

3.1 Feature Matching

Under different lighting conditions, the feature vector of a reference point can vary considerably. Irschara et al. [5] used a single point but multiple, sufficiently different feature vectors. However, this approach inflates the database and limits the number of features that can be represented. The varying appearance of a reference point can be seen as a cluster of feature vectors, and feature matching can be assumed as a typical multi-class classification problem. To efficiently classify a newly detected feature, we propose to use the Mahalanobis distance. The Mahalanobis distance accounts for the covariance of each cluster. Matsuzawa et al. [13] showed its effectiveness in image classification with the SIFT feature. Additionally, this stochastic representation of a cluster interpolates appearances not obtained.

A cluster P is composed of m feature vectors $\boldsymbol{x}_i$, i=1...m, which describe the feature's appearance under different viewing directions and lighting conditions. The mean of the cluster $\boldsymbol{\mu_P}$ and its covariance matrix $\boldsymbol{\Sigma_P}$ are defined as

$$\boldsymbol{\mu_P} = \frac{1}{m}\sum_{k=1}^{m}\boldsymbol{x_k}, \quad (1)$$

$$\boldsymbol{\Sigma_P} = \frac{1}{m}\sum_{k=1}^{m}(\boldsymbol{x_k}-\boldsymbol{\mu_P})(\boldsymbol{x_k}-\boldsymbol{\mu_P})^{\mathsf{T}}. \quad (2)$$

The distance of a feature vector $\boldsymbol{x}$ to P is defined as

$$dist^{mah}(\boldsymbol{x},P)=\sqrt{\frac{1}{m}(\boldsymbol{x}-\boldsymbol{\mu_P})^T\boldsymbol{\Sigma_P}^{-1}(\boldsymbol{x}-\boldsymbol{\mu_P})}. \quad (3)$$

In some cases, the feature vectors contributing to a cluster display no width in some directions. Because these directions do not help to classify features, we apply principal component analysis (PCA) to each cluster to reduce the size of the feature vector. The result is more compact feature vectors whose elements have strong descriptive power. As a side effect, this also reduces the processing time required to determine the distance between a detected feature and a cluster.

For each cluster we store its parameters $\mathbf{P}$, $\boldsymbol{\mu_P}$, and $\boldsymbol{\Sigma_P}$. Additionally, we store a projection matrix that maps a feature space onto the respective dimensional principal component space, where the axes of the principal component space are selected in order of singular value.

3.2 Feature Vector Acquisition

Although the feature vectors for our feature matching approach could be acquired from multiple reconstruction sessions or geo-allocated images taken under different conditions, we propose to use a more easily available and general approach.

Due to improvements in computational power and reconstruction algorithms, we assume that a detailed model of the targeted environment will be easily obtained in the future. Combined with the realistic rendering already used in various game engines, our approach can be used to capture images of the scene under desired conditions. In this paper, we localize the user to outdoor environments; however, the described approach can be applied indoors as well.

We follow Lalonde et al. [11] and assume that the illumination can be described as a combination of light emitted by the sky and the sun, where the sky is modeled as ambient light and the sun as directional light. The position of the sun is described by the azimuth angle ϕ_s and zenith angle θ_s that depend on various factors, such as time of day, season, longitude, and latitude.

ϕ_s and θ_s can be determined from the longitude l_o, the latitude l_a, the solar time t and the declination δ. The solar time is defined as

$$t = t_s + 0.17\sin(\frac{4\pi(J-80)}{373}) - 0.129\sin(\frac{2\pi(J-8)}{355}) + 12\frac{SM-l_a}{\pi}, \quad (4)$$

where t_s is the time of day (24 hours), J is the day according to the Julian calendar, and SM is the first meridian. The declination is defined as

$$\delta = 0.4093\sin(\frac{2\pi(J-81)}{368}). \quad (5)$$

For a known l_o, ϕ_s and θ_s are given as

$$\theta_s = \frac{\pi}{2} - \sin^{-1}(\sin l_o \sin\delta - \cos l_o \cos\delta \cos\frac{\pi t}{12}), \quad (6)$$

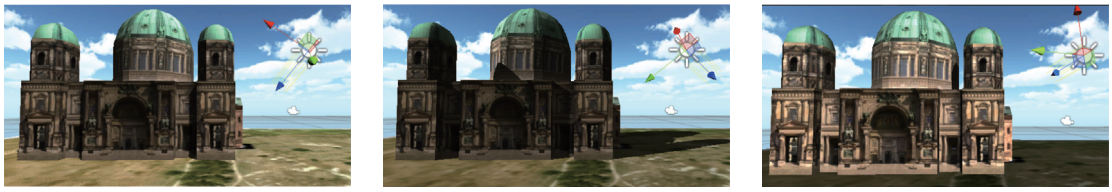
$$\phi_s = \tan^{-1}(\frac{-\cos\delta\sin\frac{\pi t}{12}}{\cos l_o \sin\delta - \sin l_o \cos\delta\cos\frac{\pi t}{12}}). \quad (7)$$

We can apply these parameters to relight the scene model to capture images from different viewpoints and to recover the feature vectors for each scenario. Because the pose of the virtual cameras and the model are known, a detected feature point can be assigned to its 3D counterpart and all feature vectors can be bundled to create the cluster described in Sec. 3.1.

4 Evaluation

We conducted three types of evaluations: an evaluation of the feature descriptor's robustness for lighting variation, a comparison between the proposed method and standard feature matching in a simulation environment, and an evaluation in a real outdoor environment by using a paper craft model. All computations were performed on a Macbook Pro with a 2.8 GHz Intel core i5 and 8 GB of 1600 MHz DDR3. We rendered all virtual views with Unity3D in its sunlight model[1]. For our synthesized experiments our chosen model was the Berlin Cathedral in the City of Sights dataset [4].

[1] http://wiki.unity3d.com/index.php/SunLight



No. 1 No. 2 No. 3

図 1 Examples of lighting variations.

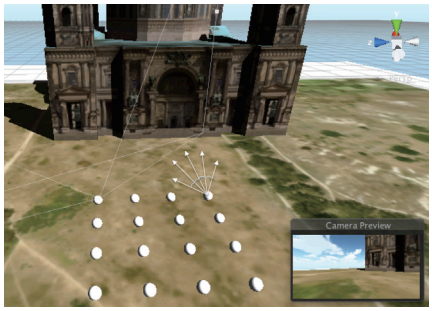


図 2 Camera positions and orientations for the simulation. The input images are generated from 16 positions and 5 directions in 15 degree steps.

4.1 Descriptor Robustness under Lighting Variations

Under different types of illumination, the appearance and the feature vector will vary. To evaluate its impact on the localization, we performed a simple test with the commonly used descriptors SIFT and SURF. We use three different lighting conditions to generate virtual scenes. In all conditions, we change only the position of the sun and keep the intensity and color constant. We show examples of images for each condition in Fig. 1. In conditions No. 1 and No. 2, the sun is illuminating the model from the side. In condition No. 3, the sun is illuminating the building from the front, and results in a brighter appearance of the model.

We follow Kurz et al. [10] to create a database for each condition. We record images from 16 different locations and under 5 different orientations, as shown in Fig. 2. We follow Kurz et al. [9] to select 2000 of the most representative features, which are stored in the databases for the respective lighting conditions. We refer to the SIFT feature databases as $D_{SIFT}i$ and the SURF feature databases as $D_{SURF}i$, where i is the respective lighting condition. Additionally, we create databases $D_{SIFT}ALL$ and $D_{SURF}ALL$, which consist of all three lighting conditions.

Results and Discussion

In our evaluation we used all 80 training images from which we constructed the databases. We determined the camera pose for an input image for all databases with the OpenCV function “cv::SolvePnPRansac”. An estimation is assumed to be correct if the position is offset by less than 0.5 m from the ground truth. The width of the building is set to 40 m. The performance of a database is evaluated by the ratio of the correct localization defined above to the number of query images. We show the results in Tables 1 and 2. We also show the results of the matching for the SIFT features for one camera pose in Fig. 3, where a good feature match is determined by a reprojection error of less than 20 pixels in a 742 × 373 pixel image.

As expected, the localization is more likely to fail for images taken under different lighting conditions. It is especially notable that in condition No. 3, the accuracy of the databases constructed under conditions No. 1 and No. 2 is greatly reduced. This is partially due to a larger number of detected features, as the front of the building is more visible. The additional features lead to a higher number of false matches and thus incorrect localization. $D_{SIFT}ALL$ and $D_{SURF}ALL$, which include all three lighting conditions, show better correct localization ratios for all inputs. These results indicate that a database of all the various lighting conditions yields robust localization for lighting variations. However, the number of accumulated records in the database increases in proportion to the lighting variations. Thus, the processing costs also increase. In other words, a trade-off exists between robustness and processing costs.

4.2 Localization in a Virtual Environment

To provide an objective evaluation of our proposed approach, we synthesize an image dataset composed of 200 different lighting conditions, i.e., different sun positions and illumination colors, as described in Sec. 3.2. For each lighting condition we take 50 images from

表 1 Ratio of correct localization with SIFT (%).

Input	$D_{SIFT}1$	$D_{SIFT}2$	$D_{SIFT}3$	$D_{SIFT}ALL$
Env. 1	91.25	71.25	86.25	95.00
Env. 2	90.00	85.00	81.25	91.25
Env. 3	64.75	48.75	87.75	85.00

表 2 Ratio of correct localization with SURF (%).

Input	$D_{SURF}1$	$D_{SURF}2$	$D_{SURF}3$	$D_{SURF}ALL$
Env. 1	90.00	66.25	85.00	85.00
Env. 2	78.75	78.75	77.50	83.75
Env. 3	51.25	30.00	77.50	76.25

different camera poses. Some examples are shown in Fig. 4.

We randomly select 100 lighting conditions from which we train our proposed matching and construct a comparison feature database. The remaining 100 conditions are used as an evaluation dataset.

We observed that the SIFT descriptor seems to be robust against varying lighting conditions, and so we use it as the feature descriptor of choice. For each lighting condition we select L representative reference points according to Kurt et al. [9]. We combine the points into a database D_{SIFT} and also use the points to train our classifier.

Results and Discussions

We compare the localization based on the matching results of our method and L_2-norm matching with D_{SIFT}. The matches are computed with the OpenCV function "cv::BruteForceMatcher". Again, we define a localization as successful if the positional error deviates from the ground truth by less than 0.5 m. We train our classifier with different combinations of parameters, as shown in Table 3. We show the impact of the number of principal axes P on the localization in Fig. 5. As shown, the localization rate forms a plateau at around 12-16 axes. We show the impact of the number of reference points L for 16 principal axes in Figs. 6 and 7.

Our method performs better than D_{SIFT} for a small number of features. The training dataset D_{SIFT} outperforms our method for more than 500 reference features and the evaluation dataset outperforms our method for more than 900 reference features. We believe that this is due to the increasing number of detected features that are not stored in our database, because our database contains only the most representative features that are observed under different lighting conditions. As a result, we observe an increasing number of false matches for these features, and the false positives in turn impact the localization results. On the other hand, the L_2-norm matching approach overfits the data and benefits from a large number of reference points.

4.3 Evaluation in a Real Environment

To evaluate how our method performs in real conditions, we constructed a paper-craft model of the Vienna concert hall and the ground plane from the City of Sights dataset. When recording real data, we used a compass and level gauge to align it with its virtual counterpart. The model was placed outdoors (Fig. 8) and data was recorded at different times of the day and under different lighting conditions. Table 4 shows the time and conditions for the recordings. We recorded the model with an iPhone 5S with the video mode set to 720 p and three images per frame. From each recording we randomly selected 100 frames for the evaluation.

The virtual illumination was simulated by calculating the sun lighting directions described in Sec. 3.2. In actuality, the lighting was simulated every 10 days and every one hour. Figure 9 shows examples of the images in the real and simulated environments.

To obtain the reference dataset used as the ground truth for evaluation, we conducted dense feature sampling and a large number of iterations. In actuality, 5000 feature points in each lighting condition and 10000 iterations of RANSAC were conducted. We used L_2-norm for matching. Figure 10 shows some localization results for each condition of the real environment shown in Table 4. We excluded condition No. 6 from the evaluation and the reference dataset, because the localization failed for most frames of this dataset. We believe that this failure is due to the

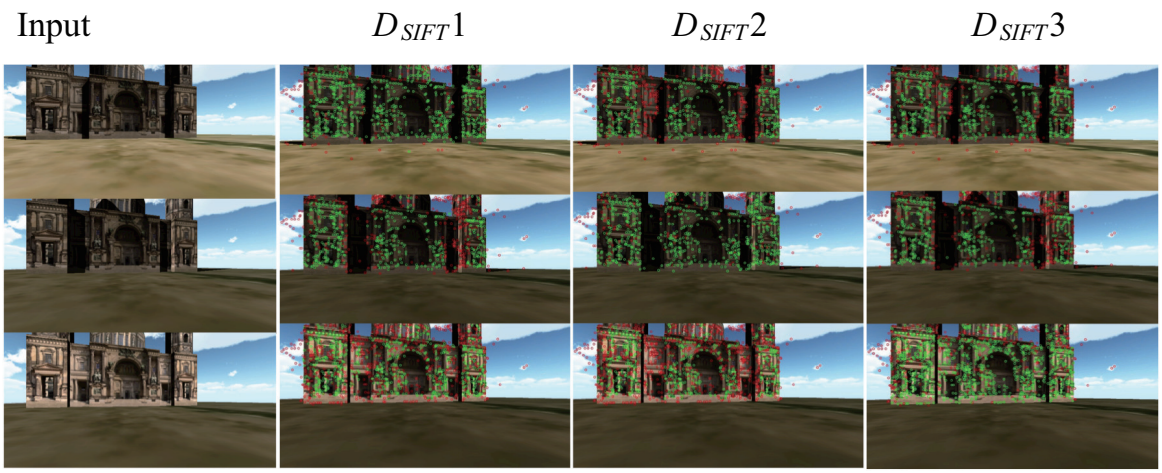


図 3　Matching results for the SIFT feature databases accumulated under different lighting conditions. The green points show the feature points for correct matching and the red points show those for mismatching.

表 3　Parameter settings

Number of principal axes [p]	8, 10, ...16..., 30
Number of reference points [L]	50, 100, ...200, ..., 1000
Number of feature points in an image	500
Number of iterations of RANSAC	500

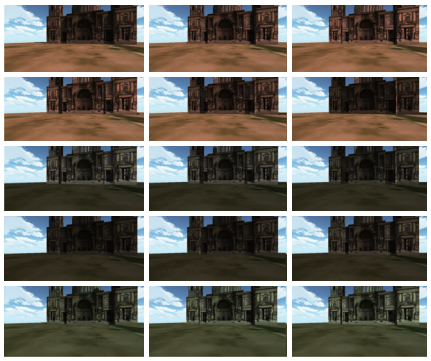


図 4　Examples of lighting variations

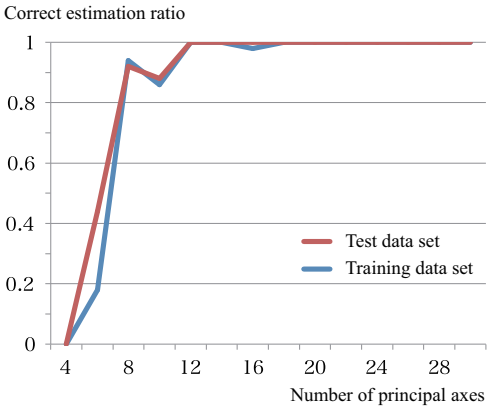


図 5　Number of principal axes and correct localization ratio.

small number of good feature points for the front of the building, which was in the shadow.

Results and Discussions

To determine if it is beneficial to simulate the color of the light, we generated two datasets, labeled D_{white} and D_{color}. The color of the light was assumed as white in D_{white} and was simulated for each condition in D_{color}. The other parameters were set according to Table 5. We show the results of the evaluation of dataset No. 3 in Fig. 12. We found only a small difference in the overall performance, and the difference was observable primarily in the higher dimension of P. Our observations show that white colored light generates feature values that are better distributed in a limited dimension of P but are robust for lighting variations. On the other hand, features generated with color simulation are better distributed in higher dimensions of principal axes. However, inaccuracy of the light color simulation does not improve the overall localization rate. An improved color simulation could prove beneficial for D_{color} in the future, but we used D_{white} in this evaluation.

We additionally performed an evaluation of the impact of the number of principal axes for $L = 200$, which showed comparable results in both methods. We found that our method performs best for databases constructed with 14-18 principal axes. The results for all datasets are shown in Fig. 11.

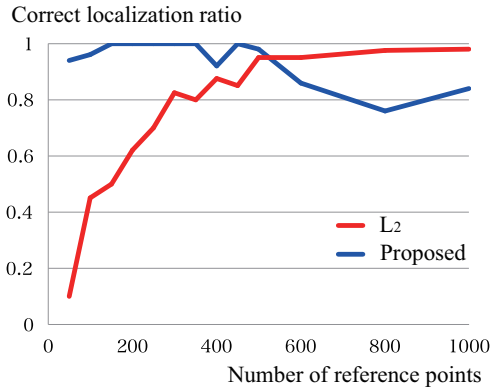


図 6 Correct localization ratios for the training data set.

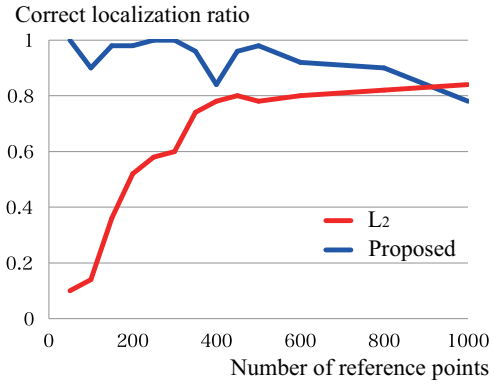


図 7 Correct localization ratio for the test data set

Similar to the simulation, we compared our classifier with the parameters from Table 5 and L_2-norm matching. For this comparison, we used the combined dataset consisting of Nos. 1-5. Similar to the simulation results, the localization rate with L_2-norm increases with the number of feature points. As shown in Fig. 13, L_2-norm outperforms our method for more than 200 feature points. However, the localization of our method remains relatively constant and independent of the number of used feature points. Additionally, our method performs faster than L_2 norm. We show the processing time in Fig. 14. Based on these observations, we found that L_2-norm matching exchanges processing time with localization correctness, and the proposed classifier from well-selected feature points relaxes the trade-off between processing cost and localization stability due to the large number of feature points.

As we discussed with regard to the results for colored lighting, some cases of low simulation accuracy cause low localization performance. In the photorealistic simulation of a real scene, we should consider many factors in the scene, including shape, reflectance of all materials, light sources, and imaging system of the camera. In this evaluation, the difference between real and virtual environments due to these factors seems to be relatively small, because we used a papercraft model as a real environment. If more factors impact the realism of the synthesized images, we must consider the improvement of the accuracy in the simulation.

図 8 Paper craft model set in an outdoor environment.

表 4 Time and weather for real environment.

No.	Date, Time, Weather
1	Jan-04-2016, 14:00, Clear sky
2	Jan-25-2016, 12:30, Cloudiness
3	Jan-04-2016, 15:00, Clear sky
4	Jan-28-2016, 07:30, Clear sky
5	Jan-25-2016, 14:00, Clear sky
6	Jan-25-2016, 09:30, Clear sky

5 Conclusion

In this study, we propose a localization method robust for varying lighting environments. Our method consists of simulation-based database construction and feature matching using the Mahalanobis distance. In the database construction, various virtual types of illumination are simulated and many feature points are accumulated. The stochastic parameters for the Mahalanobis distance, which represents the variations of lighting, are accumulated in the database.

The results show that the proposed method performs with a lower processing time and a higher correct localization ratio than the usual localization method based on feature matching with L_2-norm. However, the lighting color simulation does not improve the localization performance. Future work to reduce processing times includes development of more efficient feature matching and database separation

表 5　Parameter settings.

Number of principal axes [P]	8, 10, ...16..., 30
Number of reference points [L]	50, 100, ...200, ..., 500
Lighting color	White, Colored
Number of features in an image	500
RANSAC iterations	500

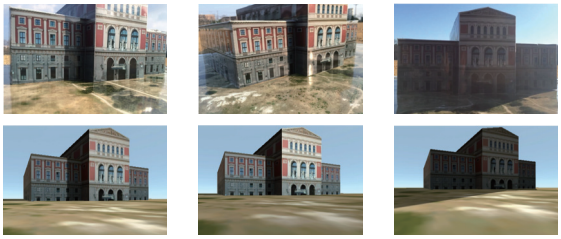


図 9　Examples of input images in real and virtual environments. The upper row shows images taken with the camera and the lower row shows virtual images of the scene generated under similar lighting conditions.

図 10　Localization results for reference data. The lines overwritten in the images are the edges of the estimated building's position.

based on contexts such as time and weather. For the lighting simulation, more accurate illumination for the simulation is necessary to achieve more accurate localization.

Acknowledgment

This work was supported in part by JSPS KAKENHI Grant Numbers JP16H02858 and JP16K16100.

References

[1] C. Arth, A. Mulloni, and D. Schmalstieg. Exploiting sensors on mobile phones to improve wide-area localization. In *Proceedings of the 21st International Conference on Pattern Recognition (ICPR)*, pages 2152–2156, 2012.

[2] C. Arth, C. Pirchheim, J. Ventura, D. Schmalstieg, and V. Lepetit. Instant outdoor localization and slam initialization from 2.5d maps. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 21(11):1309–1318, Nov 2015.

[3] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110(3):346–359, 2008.

[4] L. Gruber, S. Gauglitz, J. Ventura, S. Zollmann, M. Huber, M. Schlegel, G. Klinker, D. Schmalstieg, and T. Höllerer. The city of sights: Design, construction, and measurement of an augmented reality stage set. In *Proceedings of IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 157–163, 2010.

[5] A. Irschara, C. Zach, J. Frahm, and H. Bischof. From structure-from-motion point clouds to fast location recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2599–2606, 2009.

[6] S. Izadi, D. Kim, and O. Hilliges. Kinectfusion: Real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th annual ACM symposium on User Interface Software and Technology*, 2011.

[7] G. Klein and D. Murray. Parallel tracking and mapping for small AR workspaces. In *In Proceedings of the 6th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, 2007.

[8] D. Kurz, P. Meier, A. Plopski, and G. Klinker. An outdoor ground truth evaluation dataset for sensor-aided visual handheld camera localization. In *Proceedings of IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 263–264, 2013.

[9] D. Kurz, T. Olszamowski, and S. Benhimane. Representative feature descriptor sets for robust handheld camera localization. In *Proceedings of IEEE International Symposium on Mixed and*

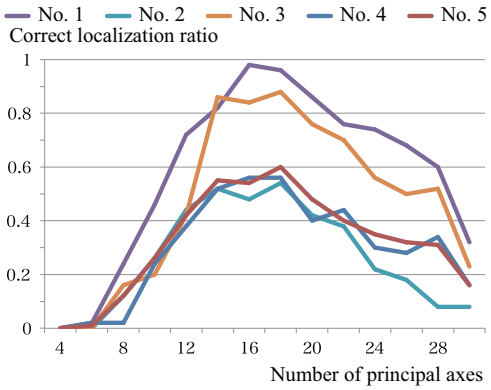


図 11　Change in correct localization ratio for different conditions.

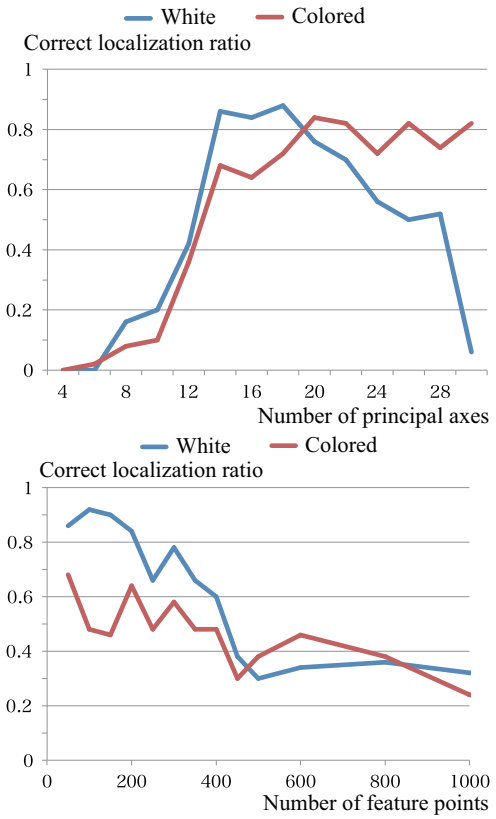


図 12 Comparison between color varied light and white light.

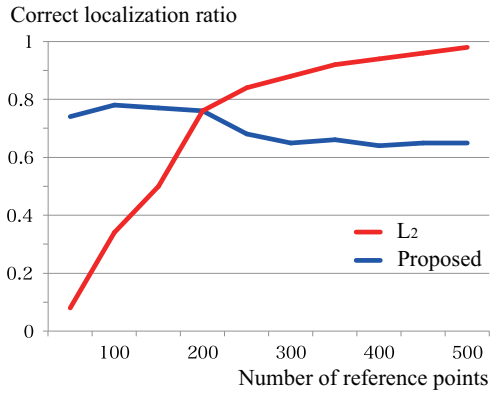


図 13 Relation between number of feature points and correct localization ratio.

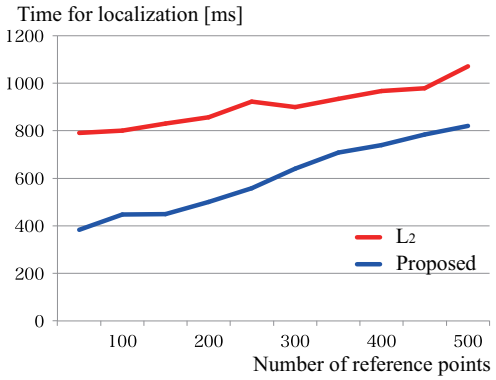


図 14 Relation between number of feature points and localization time.

Augmented Reality (ISMAR), pages 65–70, 2012.

[10] D. Kurz, P.G. Meier, A. Plopski, and G. Klinker. Absolute spatial context-aware visual feature descriptors for outdoor handheld camera localization. In *International Conference on Computer Vision Theory and Applications*, pages 36–42, 2014.

[11] J. F. Lalonde, A. A. Efros, and S. G. Narasimhan. Estimating the natural illumination conditions from a single outdoor image. *International Journal of Computer Vision (IJCV)*, 98(2):123–145, 2012.

[12] D. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, (2):91–110.

[13] T. Matsuzawa, R. Relator, W. Takei, S. Omachi, and T. Kato. Mahalanobis encodings for visual categorization. *IPSJ Transactions on Computer Vision and Applications (CVA)*, 7:69–73, 2015.

[14] T. Naseer, M. Ruhnke, C. Stachniss, L. Spinello, and W. Burgard. Robust Visual SLAM Across Seasons. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2529 - 2535, IEEE/RSJ, 2015.

[15] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós. Orb-slam: A versatile and accurate monocular slam system. *IEEE Transactions on Robotics*, 31(5):1147–1163, Oct 2015.

[16] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison. Dtam: Dense tracking and mapping in real-time. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pages 2320–2327. IEEE, 2011.

[17] J. Rekimoto and Y. Ayatsuka. Cybercode: Designing augmented reality environments with visual tags. In *Proceedings of the ACM Designing Augmented Reality Environments (DARE)*, pages 1–10. ACM, 2000.

[18] T. Schöps, J. Engel, and D. Cremers. Semi-Dense Visual Odometry for AR on a Smartphone. In *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 145–150. IEEE, 2014.

[19] G. Simon. Tracking-by-Synthesis Using Point Features and Pyramidal Blurring. In *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality (ISMAR)* pages 85–92. IEEE, 2011

[20] Y. Shinozuka, F. de Sorbier, and H. Saito. Specular 3D Object Tracking by View Generative Learning. In *Proceedings of Irish Machine Vision and Image Processing Conference*, pages 9–14, 2014.

[21] J. Ventura and T. Höllerer. Wide-area scene mapping for mobile visual tracking. In *Proceedings of IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 3–12.

[22] J. Ventura, C. Arth, G. Reitmayr, and D. Schmalstieg. Global localization from monocular slam on a mobile phone. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 20(4):531–539, 2014.

（2016年12月5日受付）

［著者紹介］

Tomohiro Mashita （正会員）

Tomohiro Mashita graduated from Osaka University in 2001 and completed the M.S. and doctoral programs in 2003 and 2006, respectively. He was a postdoctoral fellow at Osaka University from 2006 to 2008. He was a senior research fellow at Graz University of Technology from 2012 to 2013. He is currently an Assistant Professor at Cybermedia Center, Osaka University. He is research interest includes Computer Vision, Pattern Recognition, and Human Interface. He is a member of the IEICE, the IPSJ, the VRSJ and the IEEE.

Alexander Plopski

Alexander Plopski received his B.Sc. and M.Sc. from Technical University of Munich in 2010 and 2012, respectively. In 2016, he obtained a Ph.D. in information system design from Osaka University. Since 2016, he works as an Assistant Professor at the Nara Institute of Science and Technology.

Akila Kudo

He received his BE and ME degree from Osaka University in 2014 and 2016, respectively. In 2016, he joined Hitachi, Ltd.

Tobias Höllerer

Tobias Höllerer holds an Informatik Diplom from the Technische Universität Berlin and MS and PhD degrees in computer science from Columbia University. At UCSB, he co-directs the Four Eyes Laboratory conducting research in the four I's of Imaging, Interaction, and Innovative Interfaces. Dr. Höllerer's research interests lie in the area of human-computer interaction and experimental systems, with a particular focus on augmented and virtual reality, information visualization, 3D displays and interaction, and social and adaptive user interface technologies.

Kiyoshi Kiyokawa （正会員）

He received his ME and PhD degrees in information systems from Nara Institute of Science and Technology (NAIST) in 1996 and 1998, respectively. He is currently a Professor at Graduate School of Information Science, NAIST. He worked for Communications Research Laboratory from 1999 to 2002. He was a visiting researcher at Human Interface Technology Laboratory at the University of Washington from 2001 to 2002. He was an Associate Professor at Cybermedia Center, Osaka University from 2002 to 2017. His research interests include virtual reality, augmented reality, human augmentation, 3D user interface, and CSCW. He is a member of the IEICE, the IPSJ, the VRSJ, the HIS, and the ACM.

Haruo Takemura （正会員）

He received his BE, ME, and PhD degrees from Osaka University in 1982, 1984, and 1987, respectively. In 1987, he joined Advanced Telecommunication Research Institute, International. In 1994, he joined Nara Institute of Science and Technology, as an associate professor in the Graduate School of Information Science and Technology. From 1998 to 1999, he was a visiting associate professor at the University of Toronto, Ontario, Canada. He is a Professor at Cybermedia Center, Osaka University since 2001. His research interests include interactive computer graphics, human-computer interaction, and mixed reality. He is a member of the IEICE, the IPSJ, the VRSJ, the HIS, the IEEE, and the ACM.