International Conference on Artificial Reality and Telexistence
Eurographics Symposium on Virtual Environments (2016)
D. Reiners, D. Iwai, and F. Steinicke (Editors)

Simulation based Camera Localization under a Variable Lighting Environment

T. Mashita[1], A. Plopski[2], A. Kudo[1], T. Höllerer[3], K. Kiyokawa[1], and H. Takemura[1]

[1] Cybermedia Center, Osaka University, Japan
[2] Graduate School of Information Science, Nara Institute of Science and Technology, Japan
[3] Department of Computer Science, University of California, Santa Barbara, US

Abstract

Localizing the user from a feature database of a scene is a basic and necessary step for presentation of localized augmented reality (AR) content. Commonly such a database depicts a single appearance of the scene, due to time and effort required to prepare it. However, the appearance depends on various factors, e.g., the position of the sun and cloudiness. Observing the scene under different lighting conditions results in a decreased success rate and accuracy of the localization.
To address this we propose to generate the feature database from a simulated appearance of the scene model under a number of 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 a 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 over a single database. The proposed feature descriptor achieves a higher localization ratio with fewer feature points and a lower process cost.

Categories and Subject Descriptors (according to ACM CCS): I.4.7 [Image Processing and Computer Vision]: Feature Measurement—Feature representation

1. Introduction

Augmented Reality (AR) content is commonly spatially registered relative to a reference target. Although fiducial markers are a common occurrence in AR applications, over the past decade vision-based localization and tracking algorithms shifted towards markerless environments. Hereby, localization refers to an initial pose estimation and tracking to the estimation of the user pose in a continuous stream of information. Tracking of the camera has been mostly solved over the years with robust algorithms that are based on sparse 3D features [KM07, MAMT15], depth-sensing cameras [IKH11], dense [NLD11] and semi-dense [SEC14] reconstruction of the environment. However, even the best tracking algorithm is useless if the initial localization is incorrect.

State-of-the-art mobile devices are equipped with a variety of sensors, e.g., camera, compass, gyroscope, accelerometers, and GPS sensor, that can be used to estimate the user's pose. However, the raw data provided by such sensors is not accurate enough for user localization, e.g., the error in the position estimated from the GPS is commonly off by more than 1m. Visual search and matching algorithms are therefore employed to further refine the information provided by the localization sensors.

Mobile devices have only limited computational resources as well as limited bandwidth. Therefore, localization is performed against a database of feature vectors that describe the appearance in the environments. Such a database describes a static appearance of the scene and cannot account for large variations in the appearance due to changing lighting effects, e.g., largely different sun position, cloudiness outdoors, and different lights being turned on and off indoors. The accuracy and rate of the localization decreases with changing appearance of the features.

Creating databases that are capable of addressing such changes is a tedious process, as 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 may become very large, which in turn increases the time to match an image against it, and require months to record.

In this paper we address the mentioned problems through a dual approach. We propose to forego the repetitive data acquisition in favor of simulating the appearance of the scene under varying lighting conditions. These conditions are known, outdoors and indoors, as there is only a discrete set of possible light origins and degrees of cloudiness. We also propose to match features based on the Mahalanobis distance, instead of the commonly used L_2 distance, to better represent how feature vector change under different illumination conditions. This dual approach is an application of the pattern clas-

DOI: 10.2312/egve.20161437

sification scheme in the feature matching because the data acquisition by simulation provides correct association between 3D point and 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 propose a new feature descriptor and matching method that accounts for appearance changes under varying lighting conditions.
3. The compare our method against 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 that have been rigidly installed into the environment and whose position has been calibrated beforehand [RA00].

Ventura et al. [VARS14] propose to regard localization as a part of Simultaneous Localization and Mapping(SLAM)-based tracking. The first two keyframes of the tracking are uploaded to a server that determines the respective 7DOF 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. [KMPK14] target environments with many repetitive features, e.g., windows in a façade. To limit the number of false positive matches they propose to limit the number of features matched against. Hereby, the authors determine an initial 3D position of the feature by intersecting its backprojection with the scene model, given the pose from the sensors. The feature is then matched only to features in the database whose position is within the proximity of the reconstructed 3D position. The authors report that their method achieves higher accuracy than naïve feature matching and orientation aware feature matching [AMS12]. Additionally, their approach greatly reduces the number of descriptor comparisons required in the matching step.

Arth et al. [APV*15] use machine learning to detect facades in the taken image. The user is then localized through matching of the extracted facades with a 3D map of the surroundings. They report that their method usually achieves localization errors within the range of 1-4m and orientation errors of less than 3°. As their method requires prior sensor information and at least two visible facades it cannot be easily applied indoors or scenes where these requirements are not met.

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 [Low] and SURF [BETVG08] descriptors of detected corners have proven to be robust against orientation, scale and partially illumination changes. These descriptors have also found application in a variety of localization [IZFB09, VH] and tracking [KM07] solutions. With the rise of mobile computing, modified descriptors that include the additional sensor information have shown to improve matching results and reduce the number of comparisons needed to match the feature with a prerecorded database. Kurz et al. [KMPK13] propose Gravity-aligned feature descriptors (GAFD), where the gravity vector of the hand-held device helps distinguish between similar features with different global orientation, e.g., the corners of a window. In [KMPK14] the authors use the scale of the feature that was retrieved from a known model to reduce the number of features to be matched against.

Our work is in the spirit of the above work in that an extension of the commonly used features is applied to further improve the robustness of the matching. However, we differ from previous work 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 acquire a representative feature database.

Ventura et al. [VH] reconstruct the surroundings through Structure-from-Motion and manually set the position, scale and orientation of the reconstruction. They use all reconstructed points to localize the user from images taken by an omni-directional camera. Similarly, Irschara et al. [IZFB09] reconstruct a point-cloud model of the scene from a large image database. They additionally generate virtual views of the scene and keep the smallest subset that covers the targeted viewing area.

Kurz et al. [KMPK14] use 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 the authors generate virtual views of the scene. They recover a representative feature subset according to the method of [KOB12].

Our method resembles [KMPK14] and [IZFB09] in that a simulation and a dense 3D model is used to generate the feature database. Contrary to their works we do not assume a static model that is simply viewed from different poses, but model the appearance of the scene under varying illumination conditions.

3. Feature Matching with Simulation based Database and Mahalanobis Distance

In this section we describe in detail the main contributions of our paper, a feature matching methodology for databases that include multiple feature vectors of the same reference point, namely a 3D

point in the scene, and a scheme for acquisition of feature vectors under varying lighting conditions and viewpoints.

3.1. Feature Matching

Under different lighting conditions, the feature vector of a reference point can vary considerably. Irschara et al. [IZFB09] represent a single point but multiple, sufficiently different, feature vectors. However, this 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 with a given variance of the feature parameters and feature matching as a classification of a best-fit cluster. To efficiently classify a newly detected feature, we propose to use the Mahalanobis distance. The Mahalanobis distance accounts for the covariance of each cluster and Matsuzawa et al. [MRT*15] shows its effectiveness in an image classification with the SIFT feature. Additionally, this stochastic representation of a cluster interpolates not obtained appearances.

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

$$\boldsymbol{\mu}_P = \frac{1}{m}\sum_{k=1}^{m}\boldsymbol{x}_k, \tag{1}$$

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

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

$$dist^{mah}(\boldsymbol{x},P) = \sqrt{\frac{1}{m}(\boldsymbol{x}-\boldsymbol{\mu}_P)^T\boldsymbol{\Sigma}_P^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_P)}. \tag{3}$$

In some cases, the feature vectors contributing to a cluster display no width in some directions. As these directions do not help classifying features, we apply Principal Component Analysis (PCA) to each cluster to reduce the size of the feature vector. This results in a 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 thus 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 takes under different conditions, we propose to use a more easily available and general approach.

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

We follow [LEN12] 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 the 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. Hereby, the solar time is defined as

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

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

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

For a known l_o, ϕ_s and θ_s are given as

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

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

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

4. Evaluation

We conducted three types of evaluation consisting of an evaluation of feature descriptor's robustness for lighting variation, comparison between proposed method and usual feature matching in a simulation environment, and an evaluation in an outdoor real environment using paper craft. All computations were performed on a Macbook pro with 2.8 GHz Intel core i5 and 8GB 1600 MHz DDR3. We rendered all virtual views with Unity3D and its sunlight model[†]. For our synthesized experiments our model of choice was the Berlin Cathedral of the City of Sights dataset [GGV*10].

4.1. Descriptor Robustness under Lighting Variation

Under different illumination the appearance and the feature vector will vary. To evaluate its impact on the localization, we performed a simple test where we test 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 an example of an image for each condition in Fig. 1. In condition No. 1 and No. 2

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

No. 1 No. 2 No. 3

Figure 1: *Examples of the variation of lighting.*

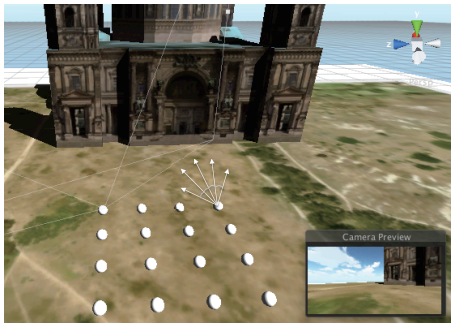

Figure 2: *Camera positions and orientations for the simulation. The input images are generated from 16 positions and 5 directions in 15 degree steps.*

the sun is illuminating the model from the side. In condition No. 3 the sun is illuminating the building from the front, which results in a brighter appearance of the model.

We follow [KMPK14] to create a database for each condition. Hereby we record images from 16 different locations and under 5 different orientations, shown in Fig. 2. We follow [KOB12] to select 2000 most representative features, which are used as the database for the respective lighting condition. 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.

4.1.1. Results and Discussions

In our evaluation we used all 80 training images from which we constructed the databases. We determined the camera pose of 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. Hereby, the width of the building is set to 40 m. 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 re-projection error of less than 20 pixels.

As expected, the localization was more likely to fail on 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 better visible. The additional features lead to a higher number of false matches and thus incorrect localization.

4.2. Localization in Virtual Environment

To provide an objective evaluation of our proposed approach we synthesize an image dataset that is composed of 200 different lighting conditions, with different sun positions and illumination colors, as described in Sec. 3.2. For each lighting condition we take 50 images from different camera poses. Some examples are shown in Fig. 4.

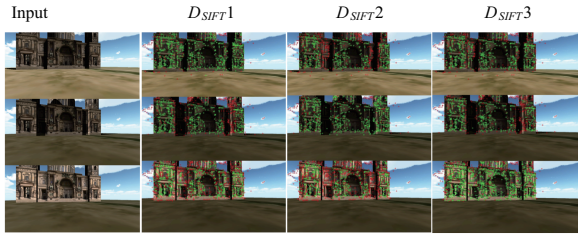

Figure 3: *Matching results of the SIFT feature databases accumulated under different lighting conditions: Green points shows the feature points of correct matching and red points shows mismatching.*

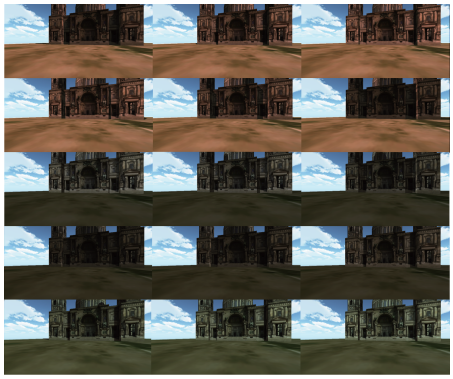

Figure 4: *Examples of lighting variations*

We randomly selected 100 lighting conditions from which we trained our proposed matching and constructed a comparison feature database. The remaining 100 conditions were used as an evaluation dataset.

As we observed that the SIFT descriptor seems to be robust against varying lighting conditions we used it as the feature descriptor of choice. For each lighting condition we selected L representative reference points according to [KOB12] that we combine into a database D_{SIFT} and also use to train our classifier.

4.2.1. Results and Discussions

We compare the localization based on matching results of our method and L_2-norm matching with D_{SIFT}. Hereby, the matches

Table 1: *Ratio of correct localization with SIFT in %.*

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

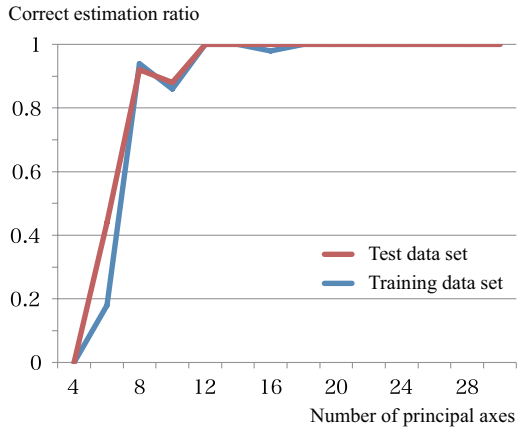

Figure 5: *Number of principal axes and correct localization ratio.*

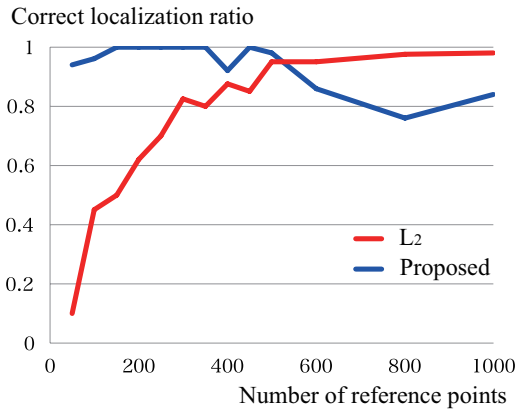

Figure 6: *Correct localization ratio on the training data set.*

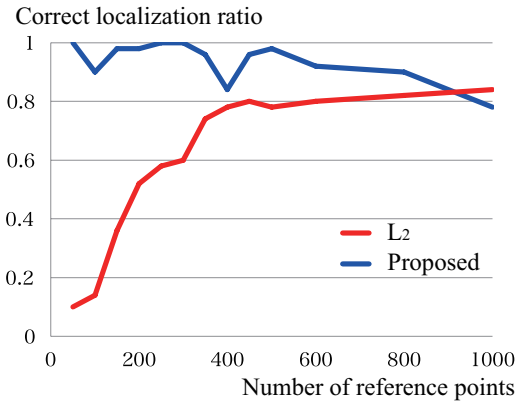

Figure 7: *Correct localization ratio on the evaluation data set*

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 combination of parameters, as shown in Table 3. We show the impact of the number of principle axes P on the localization in Fig. 5. As shown, the localization rate is plateaued around 12-16 axes. We show the impact of the number of reference points L for 16 principle axes in Figs. 6 and 7.

Our method performs better than D_{SIFT} for a small number of features. On the training dataset D_{SIFT} outperforms our method for more than 500 reference features and on the evaluation dataset for more than 900 reference features. We believe that this is due to an increasing number of detected features that are not stored in our database, as it contains only the most representative features that are observed under different lighting conditions. As a result, we observe an increasing number of false matches of these features, which in turn impacts the localization results. On the other hand, 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 of the Vienna concert hall and the Ground Plane from the City of Sights dataset. To improve the rigidness of the craft, we printed it on heavy paper and reinforce it with a card board. When recording the real data, we used a compass and level gauge to align it with its virtual counterpart. The model was placed outdoors (Fig. 8) and was recorded at different times of the day and different lighting conditions. Table 4 shows the time and conditions of the recordings. We recorded the model with an iPhone 5S with the video mode set to 720p and three images per frame. From each recording we randomly selected 100 frames that were used in the evaluation.

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

To obtain the reference dataset used as the ground truth for evaluation, We conducted dense feature sampling and a large number of iterations. In actual, 5000 feature points in each lighting condition and 10000 iterations of RANSAC were conducted. We used

Table 2: *Ratio of correct localization with SURF in %.*

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

Table 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 iteration of RANSAC	500

Figure 8: *Paper craft set in outdoor environment.*

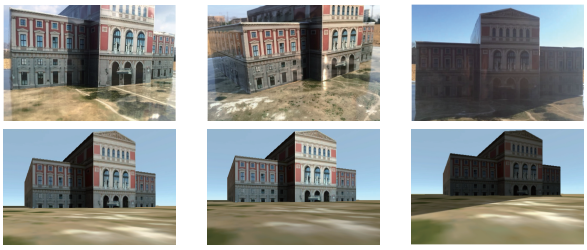

Figure 9: *Examples of input images in the real and virtual environment. The upper row shows images taken with the camera and lower row virtual images of the scene generated under similar lighting conditions.*

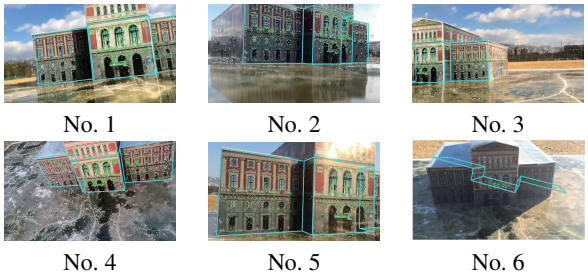

Figure 10: *Localization result of the reference data. The lines overwritten in the images are the edges of estimated building's position.*

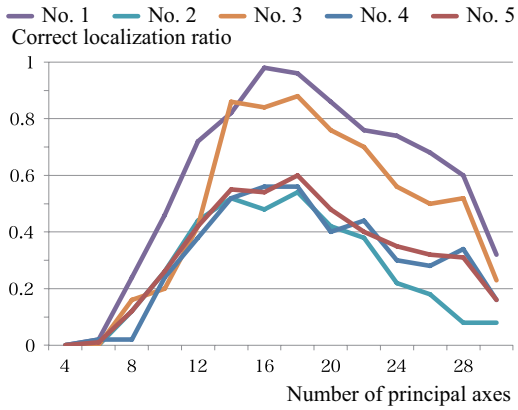

Figure 11: *Variations of correct localization ratio in each condition.*

L_2-norm for matching. Figure 10 shows some localization results in each condition of the real environment shown in Table 4. We have excluded condition No. 6 from the evaluation and the reference dataset as the localization failed for most frames of this dataset. We believe that this is due to the front of the building being in the shadow, which lead to a small number of good feature points.

4.3.1. Results and Discussions

To determine if it is beneficial to simulate the color of the light we generated two datasets D_{white} and D_{color}, where in D_{white} the color of the light was assumed as 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 that there was only a small difference in the overall performance and it 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 principle axes. However, inaccuracy of the light color simulation does not improve the overall localization rate. An improved color simulation may prove beneficial for D_{color} in the future, but we use D_{white} in this evaluation.

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

Similar to the simulation we compare our classifier with the parameters from Table 5 and L_2-norm matching. For this comparison We used the combined dataset consisting of No. 1-5. Similar to the simulation results, the localization rate with the L_2-norm increases with the number of feature points. As shown in Fig. 13 it

Table 4: *Time and weather of the 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

Table 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

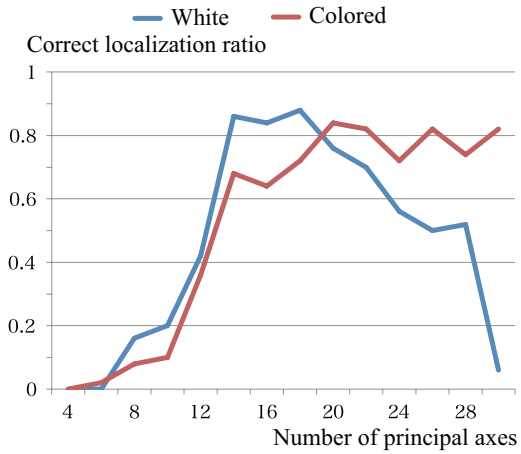

Figure 12: *Comparison between color varied light and white light*

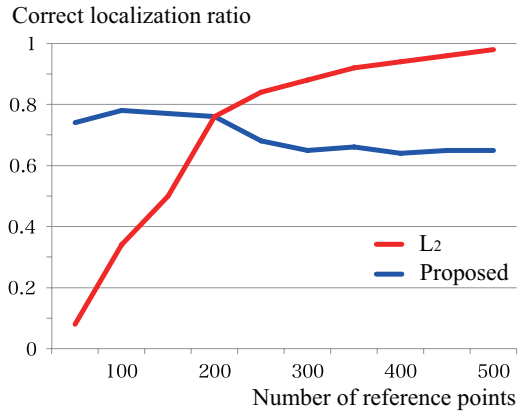

Figure 13: *Relationships between the number of feature points and correct localization ratio.*

outperforms our method for more than 200 feature points. However, the localization of our method remains relatively constant 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, whereas L_2-norm matching exchanges processing time with localization correctness, the proposed classifier from well selected feature points relaxes the trade-off between processing cost and localization stability due to large number of feature points.

5. Conclusion

In this study, we proposed a localization method robust for varying lighting environment. Our method consists of the simulation based database construction and feature matching on the Mahalanobis distance. In the database construction, various virtual illuminations are simulated and lots of feature points are accumulated. The stochastic parameters for the Mahalanobis distance which represents variation of lighting are accumulated to the database.

The results show that proposed method performs lower process-

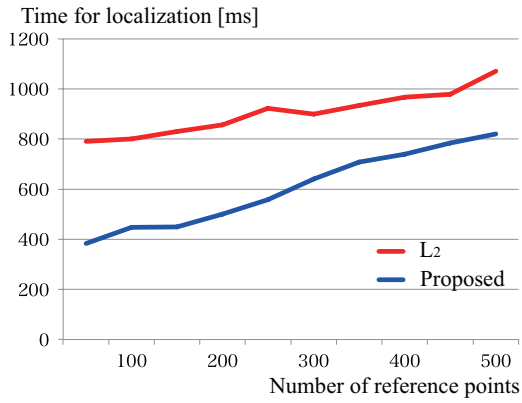

Figure 14: *Relationships between the number of feature points and localization time.*

ing time and higher correct localization ratio than usual localization method based on feature matching with L_2-norm. However, lighting color simulation does not improve localization performance. Future works to reduce processing times includes development of a more efficient feature matching and database separation based on the context such as time, weather, and so on. Regarding lighting simulation, more accurate illumination for the simulation is necessary to achieve more accurate localization.

Acknowledgment

This work was partly supported by JSPS KAKENHI Grant Number JP16H02858 and JP16K16100.

References

[AMS12] ARTH C., MULLONI A., SCHMALSTIEG D.: Exploiting sensors on mobile phones to improve wide-area localization. In *Proceedings of the 21st International Conference on Pattern Recognition (ICPR)* (2012), pp. 2152–2156. 2

[APV*15] ARTH C., PIRCHHEIM C., VENTURA J., SCHMALSTIEG D., LEPETIT V.: Instant outdoor localization and slam initialization from 2.5d maps. *IEEE Transactions on Visualization and Computer Graphics (TVCG) 21*, 11 (Nov 2015), 1309–1318. 2

[BETVG08] BAY H., ESS A., TUYTELAARS T., VAN GOOL L.: Speeded-up robust features (surf). *Computer Vision and Image Understanding 110*, 3 (2008), 346–359. 2

[GGV*10] GRUBER L., GAUGLITZ S., VENTURA J., ZOLLMANN S., HUBER M., SCHLEGEL M., KLINKER G., SCHMALSTIEG D., HÖLLERER T.: 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)* (2010), pp. 157–163. 3

[IKH11] IZADI S., KIM D., HILLIGES O.: 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). 1

[IZFB09] IRSCHARA A., ZACH C., FRAHM J., BISCHOF H.: From structure-from-motion point clouds to fast location recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2009), pp. 2599–2606. 2, 3

[KM07] KLEIN G., MURRAY D.: 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). 1, 2

[KMPK13] KURZ D., MEIER P., PLOPSKI A., KLINKER G.: 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)* (2013), pp. 263–264. 2

[KMPK14] KURZ D., MEIER P. G., PLOPSKI A., KLINKER G.: Absolute spatial context-aware visual feature descriptors for outdoor handheld camera localization. In *International Conference on Computer Vision Theory and Applications* (2014), pp. 36–42. 2, 4

[KOB12] KURZ D., OLSZAMOWSKI T., BENHIMANE S.: Representative feature descriptor sets for robust handheld camera localization. In *Proceedings of IEEE International Symposium on Mixed and Augmented Reality (ISMAR)* (2012), pp. 65–70. 2, 4

[LEN12] LALONDE J.-F., EFROS A. A., NARASIMHAN S. G.: Estimating the natural illumination conditions from a single outdoor image. *International Journal of Computer Vision (IJCV) 98*, 2 (2012), 123–145. 3

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

[MAMT15] MUR-ARTAL R., MONTIEL J. M. M., TARDÃŞS J. D.: Orb-slam: A versatile and accurate monocular slam system. *IEEE Transactions on Robotics 31*, 5 (Oct 2015), 1147–1163. 1

[MRT*15] MATSUZAWA T., RELATOR R., TAKEI W., OMACHI S., KATO T.: Mahalanobis encodings for visual categorization. *IPSJ Transactions on Computer Vision and Applications (CVA) 7* (2015), 69–73. 3

[NLD11] NEWCOMBE R. A., LOVEGROVE S. J., DAVISON A. J.: Dtam: Dense tracking and mapping in real-time. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)* (2011), IEEE, pp. 2320–2327. 1

[RA00] REKIMOTO J., AYATSUKA Y.: Cybercode: Designing augmented reality environments with visual tags. In *Proceedings of the ACM Designing Augmented Reality Environments (DARE)* (2000), ACM, pp. 1–10. 2

[SEC14] SCHÖPS T., ENGEL J., CREMERS D.:. In *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality (ISMAR)* (2014), IEEE, pp. 145–150. 1

[VARS14] VENTURA J., ARTH C., REITMAYR G., SCHMALSTIEG D.: Global localization from monocular slam on a mobile phone. *IEEE Transactions on Visualization and Computer Graphics (TVCG) 20*, 4 (2014), 531–539. 2

[VH] VENTURA J., HÖLLERER T.: Wide-area scene mapping for mobile visual tracking. In *Proceedings of IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 3–12. 2