A Novel Filtering Approach for Robust and Fast Keypoint Matching in Mobile Environment

Michael Shell, *Member, IEEE,* John Doe, *Fellow, OSA,* and Jane Doe, *Life Fellow, IEEE*

***Abstract*—The abstract goes here.**

***Index Terms*—IEEEtran, journal, LATEX, paper, template.**

I. Introduction

IMAGE matching is a fundamental ~~technique~~ in a variety of computer vision applications, including simultaneous localization and mapping [1], [2], object recognition [3], panorama stitching [4], [5], augmented reality [6], [7], and visual odometry [8], [9]. To enhance the matching quality, many related techniques have been proposed, ~~such as keypoint based local matching, histogram based global matching, supervised learning, unsupervised clustering based method~~. Among them, ~~interest point~~ detection and matching has created great interest since it can provide relatively high matching quality against severe occlusion or transformation of give objects with low computing power [10].

The overall flowchart of keypoint matching is ~~shonw~~ in Fig. 1. Offline learning is prerequisite to online matching process. ~~While~~ offline learning phase, ~~reference images are~~ analyzed and and stored ~~as a keypoint database~~. ~~At~~ online learning phase, ~~a new captured image is analyzed~~ and compared with ~~the~~ database to find a nearest reference image. ~~Each phases are performed detection, description, and matching. In offline learning phase, train image(or reference image) is analyzed and keypoint database is constructed.~~ To analyze train images, ~~at first~~, keypoints are detected from the images. Then, from those keypoints, ~~local textures are analyzed and described~~. In ~~this procedure, to provide robustness against rotation, scale, perspective transform, descriptors are calculated,~~ Then, to be used in online phase, efficient matching structures, as databases, are constructed, such as kd tree [11], hashing

이후 이렇게 만들어진 database를 이용하여 online matching 과정에서는 입력된 query 영상에서 corresponding keypoint pair를 찾게 된다. 이를 위하여 입력된 query 영상에서 특징점을 검출하고, 검출된 특징점의 descriptor를 생성하여 database에 저장된 descriptor와 가장 유사한 특징점을 계산한다.

~~In this paper, we proposed a keypoint filtering algorithm which evaluates keypoints and stores better matchable subset, to provide not only fast speed, but also robust matching quality.~~ Conventional keypoint matching methods stores almost every keypoints which are detected by keypoint detection process by ~~geometrical~~ characteristics. ~~However,~~ keypoint detection processes are designed to extract keypoints repeatable and robust against image transformation. ~~So,~~ detected keypoints are independent to the follow matching procedures, and ~~can not~~ guarantee the quality of matching. Therefore, some keypoints are not distinguishable, and they tend to cause inter-keypoint confusion and miss matching. (bad keypoint 이미지 추가) Also, those keypoints are stored in database and are compared every frame while matching, it decreases matching speed. To overcome these problem, ~~while offline learning,~~ detected keypoints are evaluated with respect to proposed matching quality criteria and filtered by the ~~evaluation~~ score. With this filtering method, only ~~better matching keypoints subset~~ is stored ~~as keypoint~~ database, ~~so~~ it provides more ~~precise~~ matching ~~results as well as more~~ fast matching speed.

~~This paper is structured~~ as follows: In Section 2, we discuss literature on interesting point detection and matching systems, as well as conventional keypoint filtering algorithms. Section 3 defines the proposed keypoint score function, and we validated the function with matching ground truth data. Then, we compared image patches between match-friendly keypoints and other keypoints set. In Section 4, we executed experiments to prove the proposed keypoint filtering method in various dataset and algorithms and compared over several evaluation metrics. Finally, Section 5 presents the conclusion.

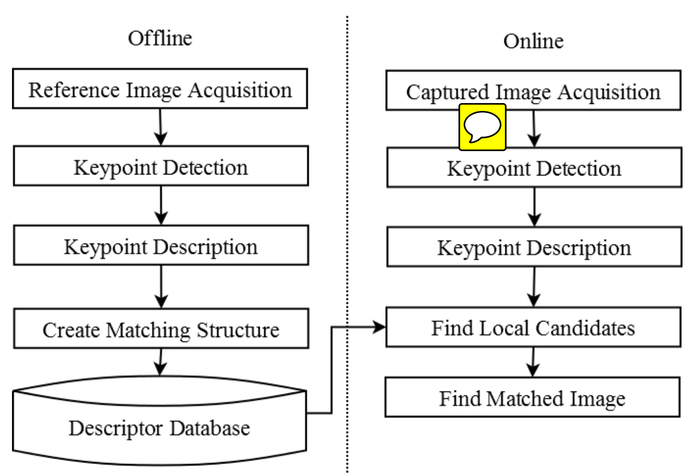


Fig. 1. ~~Process of Feature Matching~~

M. Shell is with the Department of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, 30332 USA e-mail: (see http://www.michaelshell.org/contact.html).

J. Doe and J. Doe are with Anonymous University.

Manuscript received April 19, 2005; revised December 27, 2012.

II. Related Works

A. Interest Point Detectors

1) Geometry based Detectors: 영상에서 다양한 방향에서의 gradient의 변화량을 측정함으로써 corner, edge, region을 구분하는 방법을 통하여 Interest Point 를 검출하는 방법들이 제안되었다. 이러한 방법은 Repeatability가 높은 점을 검출할 수 있다는 장점이 있으나, Image derivation이 필요하여 연산량이 높고, 노이즈에 취약하다는 한계가 있었다. 가장 먼저 Harris and Stephan [12] 은 SSD의 Hessian Matrix를 계산하고, 이 행렬을 eigen decomposition 하여 corner, edge, flat을 판별하였다. 이러한 연산을 전체 영상에 걸쳐 sliding window 방식으로 연산하여야 하기 때문에 연산량이 높은 문제가 존재하였다. Mikolajczyk and Schimid

2) Template based Detectors: 많은 연산량을 필요로 하는 Geometry based Detector 를 대신하여 영상에서 간단한 연산만으로 특징점을 검출할 수 있는 Template based Detector 들이 개발되고 있다. 가장 먼저

B. Keypoint Descriptors

1) Vector-Value based Descriptors: 기존의 SIFT [13]나 SURF [14]와 같은 vector value-based description 방법은 높은 인식율을 제공해 주었지만, orientation과 scale 등의 distortion에 robust한 descriptor를 생성하기 위하여 복잡한 연산을 수행하여야 하기 때문에 연산이 복잡하게 수행되었다.

2) Binary-Value based Descriptors: 최근에는 BRIEF [15], ORB [16], BRISK [17], FREAK [18]과 같은 다양한 Binary value-based descriptor들이 개발되고 있다. 이러한 Binary descriptor들은 그림 **??**와 같이 특징점을 중심으로 다양한 형태의 패턴을 이용하여 두 점 사이의 밝기 값을 비교하여 binary code로 표현하는 방법이다. 단순 비교 연산만으로 descriptor를 계산하기 때문에 vector-based descriptor에 비하여 연산 속도가 상당히 빠르며, 최근에는 생성 패턴을 기준으로 orientation이나 scale 등을 normalize 하기 때문에 다양한 distortion에서도 상당히 강인한 성능을 보여주고 있다. 특히 smart space와 같이 제한된 성능의 환경에서는 vector-value descriptor 기반의 복잡한 연산 보다는 단순 비교연산만으로도 처리가 가능한 binary descriptor를 사용하는 연구가 많아지고 있다.

C. Keypoint Matching

다음 방법은 matching data structure를 효율적으로 설계하여 nearest neighbor match를 빠르게 수행하도록 하고 있다. 기존의 brute force matching 방법은 query image의 모든 keypoint들을 reference image의 모든 keypoint들과 비교하는 방식으로 가장 속도가 오래 걸리지만, 가장 정확한 nearest neighbor를 검출할 수 있다는 장점이 있다.

1) Partitioning Trees: [19]에서는 kD tree 기반의 approximation 방법이 제안되었다. 이 방법은 특징의 차원이 비교적 적은 SIFT나 SURF와 같은 vector-value description 방식에서는 좋은 성능을 보여주지만, 최근에 사용되는 binary descriptor에서는 dimension이 높아 성능향상을 기대하기 어렵다는 문제가 있다. 또한, Random Forest [20] 또는 Random Fern [21]은 Binary Description 방식을 Tree 구조 또는 List 구조에 적용하여 matching structure를 구성하였다. 이러한 매칭 방식들은 인식의 속도를 향상시키고, 좀 더 정확한 approximation 값을 얻기 위하여 일반적으로 offline training 단계에서 계산된 descriptor들을 이용하여 추가의 연산을 적용하여 효율적인 matching structure를 생성한다. 하지만, 이러한 방법들을 사용한 추가적인 구조체가 상당히 복잡하고 용량이 크기 때문에 모바일 환경에서 사용하기에 매칭 구조체가 과도하게 무거워진다는 문제점이 존재한다.

2) Hashing: 가장 많이 알려진 Hashing 기반의 Nearest Neighbor 검색 기법은 Locality Sensitive Hashing [22]이다. 이 방법은 많은 수의 해쉬 함수를 이용하여 특징점을 저장하고, 같은 bucket에 저장된 특징점에 한하여 Linear Search를 수행하기 때문에 비교 연산의 횟수를 $O(N)$에서 $O(k)$로 감소시키는 장점이 있다. LSH [23]와 같은 Hashing 기반의 structure를 이용하여 matching 을 가속화하는 방법도 제안되었다 [16]. 이러한 방법은 offline training 단계에서 적절히 특징점들이 고르게 분포하도록 적절한 hash function set을 구성하는 것이 중요하다.

3) Nearest Neighbor Graph Techniques:

4) Automatic Configuration of NN Algorithms:

D. Keypoint Filtering

학습 과정에서 특징점을 평가하여 저장하는 특징점 필터링 방식은 활발하게 연구되지는 않았다. 이러한 기법은 일반적으로 특징점 검출 알고리즘에서 수행되었다.

1) Thresholding: Harris score 나 Hessian score 등을 이용하여 특징점의 강인한 검출 정도를 측정하여 필터하는 방법이 사용되었다.

2) Non-maximum Suppression: Spatial relation을 고려하여 non-maximum suppression이 수행되었다.

3) Distictiveness: 특징점의 distictiveness를 고려한 필터방법이 몇가지 제안되었다.

III. Proposed Method

본 논문에서는 matching quality 를 기준으로 detected keypoint 를 evaluate하여 filtering하는 방법을 제안한다. 이를 통하여 matching quality가 높은 점들만을 학습함으로써 online matching 과정에서 높은 matching preciseness와 빠른 속도를 얻을 수 있다. 본 장에서는 matching quality 에 따라서 keypoint를 evaluation 하기 위한 criteria 를 정의하고, ground truth data 를 기준으로 이러한 criteria 가 동작됨을 증명한다. 또한, 이러한 기준에 의하여 분류된 keypoint 들의 image patch 를 비교하여 matching에 좋은 특징점과 그렇지 않은 특징점의 xxx 측면에서의 차이점을 비교하였다.

A. Statement of the Problem

In general, keypoint matching methods 일반적으로 키포인트 기반의 매칭 방법은 미리 학습된 키포인트 데이터베이스 K^R와 입력된 영상을 분석하여 생성된 키포인트 집합 K^I를 비교하여, 가장 유사한 키포인트 pair 집합 $C = \{(k_i^R, k_j^I) | \underset{k_i \in K^R}{\operatorname{argmin}} \underset{k_j \in K^I}{\operatorname{argmin}} |k_i^R - k_j^I|\}$ 을 계산하는 과정이다. 기존의 키포인트 매칭 방법은 검출된 키포인트 집합 K^R을 그대로 사용하였으나, 본 논문에서는 키포인트 평가 함수($s(k)$)를 제안하여 이러한 평가 함수에 의하여 필터링된 집합 $K' = \{k | s(k) is \quad high\} \in K^R$ 을 계산하고, 이러한 필터링 된 부분집합 K'는 필터링 되지 않은 K^R에 비하여 더 높은 인식성능을 보여줌을 증명하고자 한다. 조금만 더 늘여쓰자

TABLE I
KEYPOINTBASED IMAGE MATCHING SYSTEMS

Reference	Detector	Descriptor	Matching	Comments
Bleser and Stricker (2008) [24]	FAST	patch, warped		
Carrera et al. (2007) [10]	Harris	SURF		
Chekhlov et al. (2007) [25]	Shi-Tomasi	SIFT-like		
Cheng et al. (2006) [8]	Harris	patch		
Davison et al. (2007) [2]	Shi-Tomasi	patch, warped		
DiVerdi et al. (2008) [26]	Shi-Tomasi	Optical flow & SURF		
Eade and Drummond (2006) [27]	FAST	patch, warped		
Klein and Murray (2007) [6]	FAST	patch, warped		
Lee and Höllerer (2008) [28]	DoG	Optical flow & SIFT		
Lepetit and Fua (2006) [20]		Randomized Trees	Randomized Trees	
Muja and Lowe (2012) [11]	DoG	SIFT	FLANN	
Nistér et al. (2004) [9]	Harris	patch		
Özuysal et al. (2007) [29]		Ferns	Ferns	
Park et al. (2008) [30]		Ferns	Ferns	
Se et al. (2002) [31]	DoG	scale, orientation		
Skrypnyk and Lowe (2004) [32]	DoG	SIFT	kD Tree	
Taylor et al. (2009) [33]	FAST	trained histograms		
Wagner et al. (2009) [34]	FAST	patch & reduced SIFT		
Wagner et al. (2010) [5]	FAST	patch, warped		
Wiliams et al. (2007) [35]	FAST	Randomized lists		

B. Keypoint Score Function

General keypoint matching generates the feature database, the subjects of comparison, by way of the offline training of the reference images before online matching procedure. In particular, as shown in the foregoing study of matching data structure, a method that composes a matching structure by applying a various computations to offline process to increase the online recognition speed and recognition rate is proposed. Such established kaypoints matching methods used simply all of the keypoints detected in feature detection module for training. However, since the feature detection algorithm is performed independently from the description algorithm, the descriptor is not able to ensue the matching performance for the detected features. 이유를 들라

Thus, in this paper, the keypoints of the subjects of training in the offline training process were assessed and only the keypoints providing rigid real-time recognition performance were selected. Accordingly, both recognition rate and speed can be enhanced by only saving discriminant (good) features in the database based on proposed method.

1) Definition of Good Keypoints: The proposed filtering method selects only good keypoints by analyzing the characteristics of detected keypoints and measuring the degree of the effectiveness for image matching.

There are several factors which good keypoints should follows:

First, good keypoints need to be stably detected as good points in an environment where targeted images change in various ways. In fact, a wide range of transformation, such as the rotation, size, noise and lighting of the targeted images. The good keypoints for recognition are detected stably.

The detection of stable keypoints can be measured by *Repeatability* condition. *Repeatability* is calculated by the ratio between the total number of converted images and the number of cases where the converted keypoints are existent in the converted images.

$$p_{repeatability}(p_i) = \frac{n_i^{overlap}}{N} \quad (1)$$

where $n_i^{overlap}$ is calculated by the frequency of the existence of converted keypoint(p_i) in the set of keypoints($T(p_i) \in K'_t$) of converted images $T_t(I)$; N is the total number of converted images ; and all keypoints have single value.

Second, Good keypoints need to be well-matched with identical keypoints even though targeted images change in various ways(*Similarity* condition). With regard to a certain keypoint(p_i) of reference images, genuine distribution' and imposter distribution' for the corresponding keypoint can be measured by calculating the matching between the descriptors of all the sets of keypoints(p_i) in images($T_t(I)$) converted in various ways during the training process. At this time, to reduce the failure in matching the corresponding keypoints and the descriptors in the converted images, the genuine distribution needs to have small value, being far enough away from match distance threshold. To this effect, it was measured using the mean of genuine distribution. As shown in Equation (2), the keypoints with the decreasing the genuine distribution are better, so the evaluation function was calculated by normalizing the mean of the genuine distribution and subtracting its value from 1.

$$p_{similarity}(p_i) = 1 - \frac{\mu_{gen,i} - \min_i \mu_{gen,i}}{\max_i \mu_{gen,i} - \min_i \mu_{gen,i}} \quad (2)$$

Third, the trained keypoints and other keypoints shall not be matched(*Separability*), which is associated with the imposter distribution of each keypoint. Of the keypoints extracted from the images converted in various images, the distribution of the matching with other keypoints rather than the converted keypoints themselves are referred to as imposter distribution.

Thus for a specific keypoint to show the low success rate of matching with other keypoints rather than themselves, in is necessary that the genuine distribution and imposter distribution are well classified. To this effect, in this paper, *Fisher's Discriminant Ratio [36]* was used. It measures the distance between two classes by the mean and distribution of sample in 1-dimensional, two class problems. Since the second Similarity condition ensures the genuine distribution is small enough, the nonexistence of the matching with the keypoints in the imposter distribution is ensured if the importer distribution is far enough away compared with the genuine distribution. Separability value also requires the normalization process as shown in equation (4).

$$FDR(p_i) = \frac{(\mu_{gen,i} - \mu_{imp,i})^2}{\sigma^2_{gen,i} + \sigma^2_{imp,i}} \tag{3}$$

$$p_{separability}(p_i) = \frac{FDR(p_i) - \min_i FDR(p_i)}{\max_i s_i} \tag{4}$$

The score functions of each keypoint can be defined using 3 criteria calculated as above. The 3 conditions are dependent, so can be defined as shown in Equation (5).

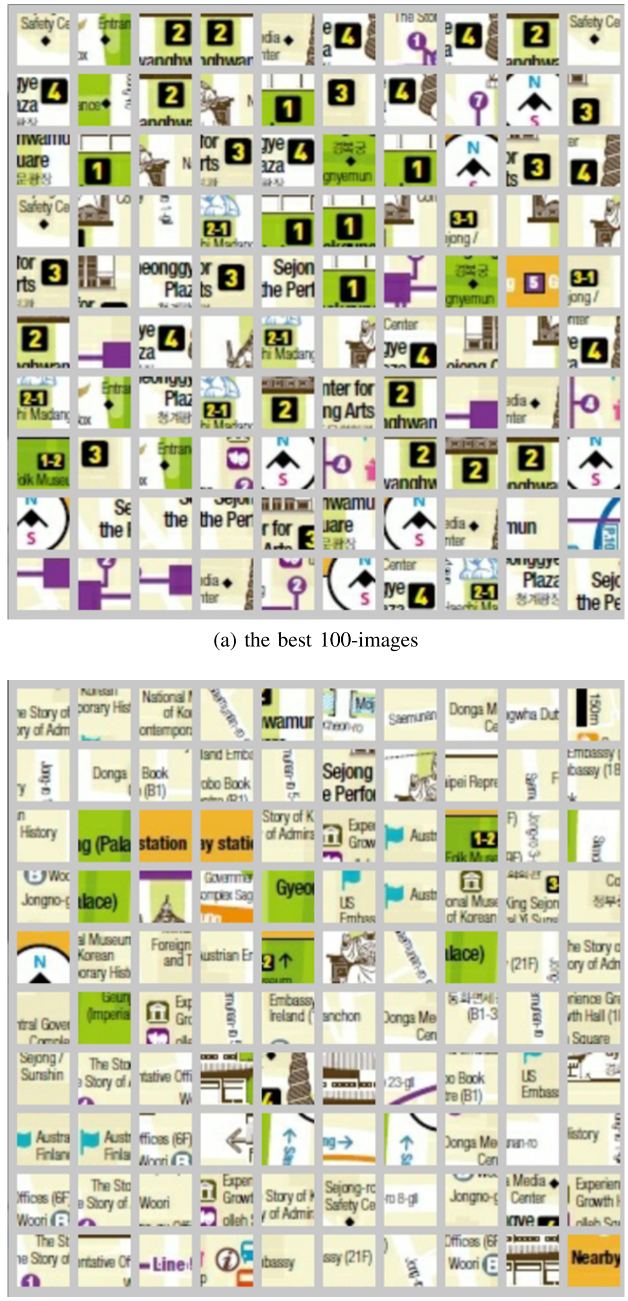
$$gf(p_i) = p_{prepeatability}(p_i) p_{similarity}(p_i) p_{separability}(p_i) \tag{5}$$

C. Proof of Criteria

1) Validation Design: To validate the proposed keypoint evaluation criteria, we examined a relationship between criteria and correct matching count of each keypoints. At first, to provide robust image matching, we synthesized image dataset by various image transformation. Then, based on this dataset, we counted correct matching count for each keypoint, and this correct matching count is a basis of matching quality. 이러한 correct matching count가 높은 특징점은 fixed image dataset 에서 더 높은 matching quality를 보여준다고 볼 수 있기 때문에 본 논문에서 제안하는 Matching에 더 적합한 keypoint로 볼 수 있다. 반대로 correct matching count가 낮은 특징점은 특징점이 반복적으로 검출되지 않거나, 모호성이 높아 inter-keypoint miss-match가 많이 발생하는 특징점으로 matching 에 적합하지 못한 keypoint로 볼 수 있다. 따라서, 이러한 correct match count 와 제안하는 keypoint evaluation score function (see, Eq. 5) 간의 상관관계를 관찰함으로써 제안하는 score function 의 적절성을 검증할 수 있다.

2) Dataset: 검증에 사용된 이미지는 서울 관광 가이드북 [37]의 *Seoul Tour Map* 16장을 사용하였다. 우리는 이러한 이미지를 대상으로 rotate(0.5 – 2.0-folds, at the interval of 0.1-fold), scaling($0° - 360°$, at the interval of 10 intervals), and blurring (Gaussian blur, $r \in \{0, 3, 5, 7\}$ pixels) 의 transform 을 적용하여 총 36,864 장의 dataset을 생성하였다. 이 중 랜덤으로 training Set 16,114 장, Test set 16,142 장을 선택하여 실험을 진행하였다.

3) Images Patches: 그림 2와 같이, correct matching count 를 기준으로 상위 10개의 keypoint 와 하위 10개의 keypoint 들의 특징을 비교하였다. 상위 10개에 대한 패치는 비교적 단순한 사각형 형태에서 많이 검출되었다. Genuine과 Impostor Histogram의 값을 정규화하여 표현된 Normal Distribution의 분포를 보면 Genuine과 Impostor 분포가 확연하게 구분되는 것을 확인할 수 있다. 반면, 하위 10개에 대한 패치는 글자 또는 단순한 패턴이 반복되는 형태에서 많이 검출되었다. Genuine과 Impostor Histogram의 값을 정규화하여 표현된 Normal Distribution의 분포를 보면 Genuine과 Impostor 분포가 많은 부분 겹쳐있어 구분이 어려운 것을 확인할 수 있다. 인식에 좋은 특징점은 큰 숫자 패치와 같이 단순한 색상으로 패턴이 큰 숫자 표지와 같은 특징점이 인식에 좋은 성능을 보여주었으며, 반대로 작은 설명 글씨와 같은 특징점들은 인식 성능이 좋지 못하였으며, 이러한 점들을 제거하고 학습을 수행하는 것이 좋다.



(a) the best 100-images

(b) the worst 100-images

Fig. 2. The Best/Worst 100-Images with Regard to Correct Matching Count

IV. Experiments

The improvement of recognition performance after the training using the keypoints filtering algorithm proposed earlier was measured. As for the experimental images, 16 images of Seoul Guide Map Pamphlet was selected. These images were deformed by way of rotating(0.5 $\sim$ 2.0-folds, at the interval of 0.1-fold) scaling ($0° \sim 360°$, at the interval of 10 intervals) and blurring (Gaussian blur, $r = 0, 3, 5, 7$ pixels). As a result, 32,256 images were obtained. Of them, 16,114 training images and 16,142 test images were selected at random. First, the keypoints of training images were detected and then the score function($gf(p_i)$) was calculated using the detected keypoints.

The keypoints database is composed of the set of all keypoints($K_{all}, n(K_{all}) = 3000$) that did not consider the score function and the set of the keypoints($K_{50}, K_{100}, K_{300}, K_{500}$) composed of top 50, 100, 300 and 500 keypoints filtered in the score function.

First the improvement of recognition speed was measured. As for the proposed method, since the keypoints were reduced to be saved in the training phase, the number of the keypoints, the subjects of comparison, decreases, which in turn increases the computing speed.

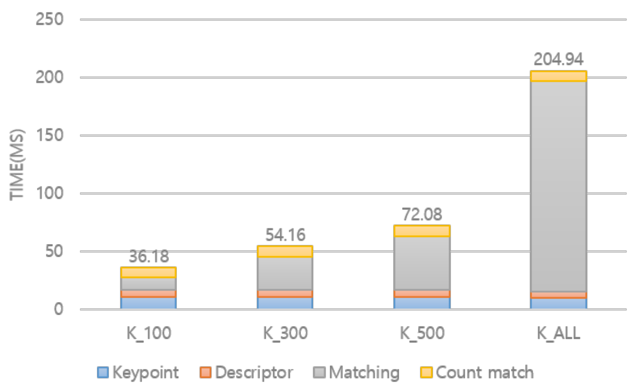


Fig. 3. Time Comparison Among Conventional Full Database and Proposed Filtered Database

As shown in Figure **??**, the computing speed improves in proportion to the number of the set of keypoints. In particular, the time spent for training deceases to $1/n$ compared to the training of whole keypoints when training was performed with 100 keypoints. In a lightweight implementation environment like smartphone, the reduction of computation provides rapid interaction. The proposed method is expeted to increases the speed and to improve the overall recognition performance. The test image recognition performance was measured using keypoints database. As for the match method for measuring recognition performance, we used the match method [38] which prevent false matches.

The results of the measurement of recognition rate using the above method were demonstrated in Figure 4. In comparison with the keypoints database using the whole of keypoint sets(K_{all}), K_{500} and K_{300} showed slight degradation of recognition rate whereas K_{100} and K_{50} showed the improvement of performance.

When performing keypoint filtering, the bad keypoints causing miss-match are eliminated, which in turn increases the reliability of the match results. To prove this, the precision [39] in the feature-level was calculated. The precision can be calculated as the ratio between the number of the correspondence pairs obtained after matching and the correct matches, indicating the insignificant proportion of mass-match and significant proportion of correction match in the match results. The increase of the ratio between correct match and match results subsequently affects the performance of robust pose estimation.

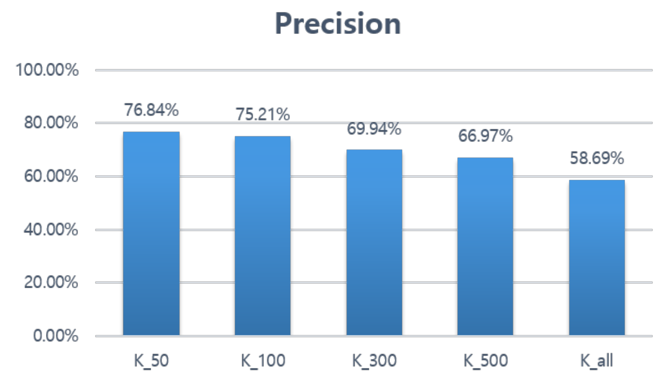


Fig. 5. Precision of filtered keypoint database

The results of precision are demonstrated in Table II and Table 5. The filtered keypoints sets showed higher precision compared to the whole of keypoints set (K_{all}). The number of the detected keypoints decreased but the ratio of correct match increased, which showed high precision. Such results are able to improve the speed and performance of robust pose estimation.

V. Conclusion

The conclusion goes here.

Appendix A
Proof of the First Zonklar Equation

Appendix one text goes here.

Appendix B

Appendix two text goes here.

Acknowledgment

The authors would like to thank...

TABLE II
PRECISION OF FILTERED MATCHING

	K_{50}	K_{100}	K_{300}	K_{500}	K_{all}
Avg. Match Result	10.098	15.618	26.747	31.409	44.859
Avg. Correct Match	7.759	11.747	18.705	21.033	26.326
Precision	76.8%	75.2%	69.9%	67.0%	58.7%

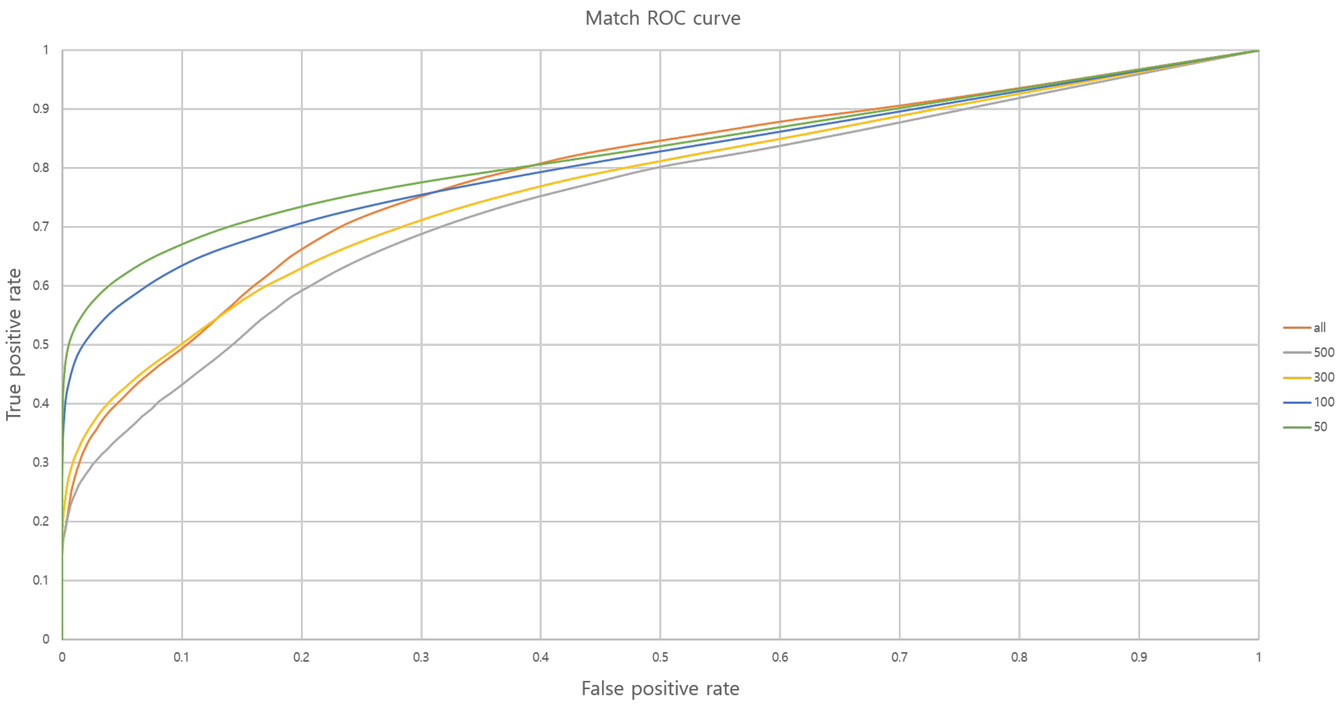


Fig. 4. ROC curve for match rate

References

[1] H. Chang, C. S. G. Lee, Y.-H. Lu, and Y. Hu, "P-SLAM: Simultaneous localization and mapping with environmental-structure prediction," *IEEE Transactions on Robotics*, vol. 23, no. 2, pp. 281–293, Apr. 2007.

[2] A. Davison, I. Reid, N. Molton, and O. Stasse, "MonoSLAM: Real-time single camera SLAM," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1052–1067, Jun. 2007.

[3] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, 2006, pp. 2161–2168.

[4] M. Brown and D. Lowe, "Recognising panoramas," in *Ninth IEEE International Conference on Computer Vision, 2003. Proceedings*, Oct. 2003, pp. 1218–1225 vol.2.

[5] D. Wagner, A. Mulloni, T. Langlotz, and D. Schmalstieg, "Real-time panoramic mapping and tracking on mobile phones," in *2010 IEEE Virtual Reality Conference (VR)*, Mar. 2010, pp. 211–218.

[6] G. Klein and D. Murray, "Parallel tracking and mapping for small AR workspaces," in *6th IEEE and ACM International Symposium on Mixed and Augmented Reality, 2007. ISMAR 2007*, Nov. 2007, pp. 225–234.

[7] D. Wagner, D. Schmalstieg, and H. Bischof, "Multiple target detection and tracking with guaranteed framerates on mobile phones," in *8th IEEE International Symposium on Mixed and Augmented Reality, 2009. ISMAR 2009*, Oct. 2009, pp. 57–64.

[8] Y. Cheng, M. Maimone, and L. Matthies, "Visual odometry on the mars exploration rovers - a tool to ensure accurate driving and science imaging," *IEEE Robotics Automation Magazine*, vol. 13, no. 2, pp. 54–62, Jun. 2006.

[9] D. Nister, O. Naroditsky, and J. Bergen, "Visual odometry," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004*, vol. 1, Jun. 2004, pp. I–652–I–659 Vol.1.

[10] G. Carrera, J. Savage, and W. Mayol-Cuevas, "Robust feature descriptors for efficient vision-based tracking," in *Progress in Pattern Recognition, Image Analysis and Applications*, ser. Lecture Notes in Computer Science, L. Rueda, D. Mery, and J. Kittler, Eds. Springer Berlin Heidelberg, Jan. 2007, no. 4756, pp. 251–260. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-540-76725-1_27

[11] M. Muja and D. Lowe, "Fast matching of binary features," in *2012 Ninth Conference on Computer and Robot Vision (CRV)*, May 2012, pp. 404–410.

[12] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proceedings of Fourth Alvey Vision Conference*, 1988, pp. 147–151.

[13] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, Nov. 2004. [Online]. Available: http://link.springer.com/article/10.1023/B%3AVISI.0000029664.99615.94

[14] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, Jun. 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1077314207001555

[15] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary robust independent elementary features," in *Computer Vision – ECCV 2010*, ser. Lecture Notes in Computer Science, K. Daniilidis, P. Maragos, and N. Paragios, Eds. Springer Berlin Heidelberg, Jan. 2010, no. 6314, pp. 778–792. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-642-15561-1_56

[16] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *2011 IEEE International Conference on Computer Vision (ICCV)*, Nov. 2011, pp. 2564–2571.

[17] S. Leutenegger, M. Chli, and R. Siegwart, "BRISK: Binary robust invariant scalable keypoints," in *2011 IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 2548–2555.

[18] A. Alahi, R. Ortiz, and P. Vandergheynst, "FREAK: Fast retina keypoint," in *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2012, pp. 510–517.

[19] J. Beis and D. Lowe, "Shape indexing using approximate nearest-neighbour search in high-dimensional spaces," in *, 1997 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1997. Proceedings*, Jun. 1997, pp. 1000–1006.

[20] V. Lepetit and P. Fua, "Keypoint recognition using randomized trees," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 9, pp. 1465–1479, Sep. 2006.

[21] M. Ozuysal, M. Calonder, V. Lepetit, and P. Fua, "Fast keypoint recognition using random ferns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 3, pp. 448–461, Mar. 2010.

[22] A. Andoni and P. Indyk, "Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions," in *47th Annual IEEE Symposium on Foundations of Computer Science, 2006. FOCS '06*, Oct. 2006, pp. 459–468.

[23] A. Gionis, P. Indyk, and R. Motwani, "Similarity search in high dimensions via hashing," in *Proceedings of the 25th International*

Conference on Very Large Data Bases, ser. VLDB '99. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1999, pp. 518–529. [Online]. Available: http://dl.acm.org/citation.cfm?id=645925.671516

[24] G. Bleser and D. Stricker, "Advanced tracking through efficient image processing and visual-inertial sensor fusion," in *IEEE Virtual Reality Conference, 2008. VR '08*, Mar. 2008, pp. 137–144.

[25] D. Chekhlov, M. Pupilli, W. Mayol, and A. Calway, "Robust real-time visual SLAM using scale prediction and exemplar based feature description," in *IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR '07*, Jun. 2007, pp. 1–7.

[26] S. DiVerdi, J. Wither, and T. Hollerei, "Envisor: Online environment map construction for mixed reality," in *IEEE Virtual Reality Conference, 2008. VR '08*, Mar. 2008, pp. 19–26.

[27] E. Eade and T. Drummond, "Scalable monocular SLAM," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, Jun. 2006, pp. 469–476.

[28] T. Lee and T. Hollerer, "Hybrid feature tracking and user interaction for markerless augmented reality," in *IEEE Virtual Reality Conference, 2008. VR '08*, Mar. 2008, pp. 145–152.

[29] M. Ozuysal, P. Fua, and V. Lepetit, "Fast keypoint recognition in ten lines of code," in *IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR '07*, Jun. 2007, pp. 1–8.

[30] Y. Park, V. Lepetit, and W. Woo, "Multiple 3d object tracking for augmented reality," in *7th IEEE/ACM International Symposium on Mixed and Augmented Reality, 2008. ISMAR 2008*, Sep. 2008, pp. 117–120.

[31] S. Se, D. Lowe, and J. Little, "Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks," *The International Journal of Robotics Research*, vol. 21, no. 8, pp. 735–758, Aug. 2002. [Online]. Available: http://ijr.sagepub.com/content/21/8/735

[32] I. Skrypnyk and D. Lowe, "Scene modelling, recognition and tracking with invariant image features," in *Third IEEE and ACM International Symposium on Mixed and Augmented Reality, 2004. ISMAR 2004*, Nov. 2004, pp. 110–119.

[33] S. Taylor, E. Rosten, and T. Drummond, "Robust feature matching in 2.3 #x00b5;s," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2009. CVPR Workshops 2009*, Jun. 2009, pp. 15–22.

[34] D. Wagner, D. Schmalstieg, and H. Bischof, "Multiple target detection and tracking with guaranteed framerates on mobile phones," in *8th IEEE International Symposium on Mixed and Augmented Reality, 2009. ISMAR 2009*, Oct. 2009, pp. 57–64.

[35] B. Williams, G. Klein, and I. Reid, "Real-time SLAM relocalisation," in *IEEE 11th International Conference on Computer Vision, 2007. ICCV 2007*, Oct. 2007, pp. 1–8.

[36] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, no. 2, pp. 179–188, Sep. 1936. [Online]. Available: http://onlinelibrary.wiley.com/doi/10.1111/j.1469-1809.1936.tb02137.x/abstract

[37] "Seoul travel guide," Feb. 2014. [Online]. Available: http://www.visitseoul.net/en/guide/ebook.do?_method=list&m=0004011019003&p=04

[38] H. Choi, G. C. Han, and I.-j. Kim, "Smart booklet: Tour guide system with mobile augmented reality," in *2014 IEEE International Conference on Consumer Electronics (ICCE)*, Jan. 2014, pp. 353–354.

[39] J. Heinly, E. Dunn, and J.-M. Frahm, "Comparative evaluation of binary features," in *Computer Vision – ECCV 2012*, ser. Lecture Notes in Computer Science, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Springer Berlin Heidelberg, Jan. 2012, pp. 759–773. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-642-33709-3_54

PLACE PHOTO HERE

Michael Shell Biography text here.

John Doe Biography text here.

Jane Doe Biography text here.