# Real-Time Feature Descriptor Matching via a Multi-Resolution Exhaustive Search Method

Chi-Yi Tsai, An-Hung Tsao, and Chuan-Wei Wang
Department of Electrical Engineering, Tamkang University, New Taipei City, Taiwan
Email: chiyi_tsai@mail.tku.edu.tw; 601460024@s01.tku.edu.tw; 600470073@s00.tku.edu.tw

***Abstract*—Feature descriptor matching plays an important role in many computer vision applications. This paper presents a novel fast linear exhaustive search algorithm combined with a multi-resolution candidate elimination technique to deal with this problem efficiently. The proposed algorithm is inspired from the existing multi-resolution image retrieval approaches, but releasing the requirement on a norm-sorted database with pre-computed multi-resolution tables. This helps to increase the applicability of the proposed method. Moreover, the computations of candidate elimination are fully performed using a simple $L_1$ distance metric, which is able to speedup the entire search process without loss of accuracy. This property leads to an accurate feature descriptor matching algorithm with real-time performance, which will be validated in the experiments by testing with the matching of SURF descriptors.**

***Index Terms*—descriptor matching, linear exhaustive search, $L_1$ norm pyramid, multi-resolution examination**

## I. Introduction

Feature descriptor matching is one of the critical stages in various computer vision applications, especially in image retrieval, image stitching, 3D pose estimation, and multi-view reconstruction, etc. As a result, the study of image feature detection, description and matching has become an active research topic in the last years. This paper focuses on the last issue, which typically can be treated as a nearest neighbor search problem in multiple dimensions [1] and thus can be resolved by using a conventional nearest neighbor search technique [2], [3]. Developing an efficient and accurate feature descriptor matching algorithm is important as it greatly affects the efficiency and accuracy of the following vision algorithms. This issue hence motivates us to design a new feature descriptor matching algorithm achieving fast search speed with high matching accuracy.

Current feature descriptor matching methods can be roughly divided into distance metric and space partitioning approaches. The former employs a distance metric to measure the similarity between a query descriptor and a sample descriptor in a given data set, and the later employ a tree data structure to efficiently speedup the search of nearest neighbors. The simplest distance metric method is well known as the linear exhaustive search (LES) method, so-called the brute-force method, which usually employs Euclidean distance metric to search for a nearest neighbor. The LES method often provides the best matching result, but it becomes inefficient as the size of database or the dimension of feature descriptor increasing dramatically.

By contrast, the space partitioning approaches are efficient for searching multiple nearest neighbors in a large database. However, it is well known that the performance of space partitioning methods is degraded exponentially as the dimension of feature descriptor growing up, called the curse-of-dimensionality [4]. In practice, the feature descriptor is usually of high dimensionality, i.e., the dimension of the popular SIFT [5] and SURF [6] descriptors are 128 and 64, respectively. This problem highlights the importance to develop an efficient and accurate descriptor matching algorithm for feature matching in high dimensional spaces. To achieve this purpose, this paper presents a fast multi-resolution exhaustive search algorithm to efficiently search the best match in a high dimensional space. The proposed algorithm is inspired from the existing multi-resolution image retrieval approaches [7], [8], but releasing the requirement on a norm-sorted database with pre-computed multi-resolution tables. This helps to increase the applicability of the proposed method. Moreover, the proposed multi-resolution candidate elimination method is able to detect and remove all non-candidates from a large candidate list via a simple $L_1$ distance computation, allowing speeding up the entire search process without loss of accuracy. This property leads to an efficient and accurate feature descriptor matching algorithm achieving real-time performance in practical applications, which will be validated in the experiments by testing with the matching of SURF descriptors.

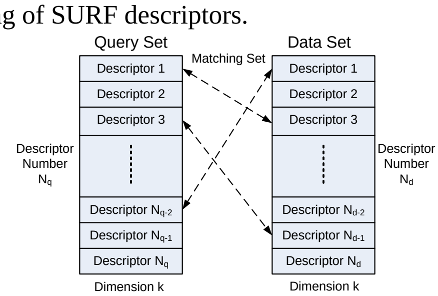



Figure 1. The feature descriptor matching problem considered in this study. That is, given a query and a data set, the goal is to find a matching set that records the descriptor-matching pairs in both given sets.

Manuscript received October 25, 2012; revised March 10, 2013.


doi:10.4304/jsw.8.9.2197-2201

## II. PROBLEM STATEMENT

Fig. 1 shows the feature descriptor matching problem considered in this study, which is also called the nearest neighbor search problem. Given a query set $\Omega_Q = \{\mathbf{X}_1, \mathbf{X}_2, \cdots, \mathbf{X}_{N_q}\}$ and a data set $\Omega_D = \{\mathbf{Y}_1, \mathbf{Y}_2, \cdots, \mathbf{Y}_{N_d}\}$, where $\mathbf{X}_i, \mathbf{Y}_j \in \Re^k$, $i = 1 \sim N_q$ and $j = 1 \sim N_d$, denote the $k$-dimensional feature descriptors in query and data sets, respectively. The goal is to find a matching set $\Omega_M = \{(x_1, y_1), (x_2, y_2), \cdots, (x_m, y_m)\}$, where $x_n, y_n \in \aleph^+$ with $n = 1 \sim m$ are nonzero positive values indexing the matching points in both given sets. For instance, Fig. 1 shows three pairs of matching points in the query and data sets, and the matching set is thus given by $\Omega_M = \{(1, 3), (3, N_{d-1}), (N_{q-2}, 1)\}$. To find the matching set, one of the most widely used methods is the LES method (so-called the brute-force method) that searches for a matching point by evaluating the similarity between a query descriptor and all unmatched descriptors in the data set. The similarity between two descriptors is usually measured by Euclidean distance (or $L_2$ distance) defined as

$$d_2(\mathbf{X}, \mathbf{Y}) = \|\mathbf{X} - \mathbf{Y}\|_2 = \sqrt{\sum_{i=1}^{k} (x_i - y_i)^2} \qquad (1)$$

where $\mathbf{X}$ and $\mathbf{Y}$ are two descriptors in the query and data sets, respectively. Fig. 2 shows the pseudo-code of the traditional LES algorithm. Note that as suggested from the previous works [9], we also employ a ratio test on descriptor distances at the last step of matching process to reject ambiguous matches. Although the LES algorithm provides a simple way to find the matching set, the run-time of this method is of order $O(kN_qN_d)$, leading to an inefficient algorithm as the dimension $k$ and length $N_q$ or $N_d$ growing dramatically.

```
The Traditional Linear Exhaustive Search Algorithm
Input: A query set ΩQ={X1,X2,…,XNq} and a data set
       ΩD={Y1,Y2,…,YNd}
Output: A match set ΩM
Initialize: Clear the match set ΩM={0}; Set a matching
counter c=0; Set a distance ratio α=0.65;
for i=1:Nq do
  d1=d2=FLT_MAX; xc=i;
  for j=1:Nd do
    dist=d2(Xi,Yj);
    if dist<d1 then
      d2=d1; d1=dist; yc=j;
    else if dist<d2 then
      d2=dist;
    end if
  end for
  if d1<αd2 then
    Push (xc,yc) into match set ΩM; c=c+1;
  end if
end for
Return: match set ΩM
```

FLT_MAX: Maximum value of a floating-point variable

Figure 2. Pseudo-code for the traditional linear exhaustive search algorithm.

Unlike the traditional LES algorithm, another commonly used search method is the space partitioning method that employs tree-like structures, such as k-d trees, to iteratively reduce the search space, efficiently increasing the nearest-neighbor search performance in a large data set. However, the author in [10] pointed out that the LES method is faster than the space partitioning one when $k \geq \log N_d$. Since such high dimensionality (i.e., $k \geq 25$) usually occurs in practical applications, this issue highlights the importance of developing a fast exhaustive search method to deal with descriptor matching in high dimensional spaces. This study thus proposes a novel fast multi-resolution exhaustive search method to achieve this purpose efficiently and effectively.

## III. THE PROPOSED FEATURE MATCHING ALGORITHM

This section presents the proposed fast exhaustive search algorithm, which is developed by combining the LES algorithm with a $L_1$-norm based multi-resolution candidate elimination technique.

### A. *Multi-Resolution Candidate Elimination*

In this subsection, a novel multi-resolution elimination method inspired from the inequality property of $L_1$ norm pyramid [7], [8] is presented to efficiently speedup the matching process in high dimensional spaces. We first define the $L_1$ distance between two descriptors such that

$$d_1(\mathbf{X}, \mathbf{Y}) = \|\mathbf{X} - \mathbf{Y}\|_1 = \sum_{i=1}^{k} |x_i - y_i| \qquad (2)$$

where $\mathbf{X} \in \Omega_Q$ and $\mathbf{Y} \in \Omega_D$. According to [7], the computation of $L_1$ distance can be extended to a multi-resolution case, allowing improving the exhaustive search performance in a multi-level manner. Suppose that the dimension of the feature descriptor satisfies $k = 2^L$ for any nonzero positive value $L$. A $L_1$ norm pyramid associated with a feature descriptor $\mathbf{X} \in \Re^k$ can be defined as a sequence of multi-resolution descriptors $\{\mathbf{X}^{(0)} \; \mathbf{X}^{(1)} \; \cdots \; \mathbf{X}^{(L-1)} \; \mathbf{X}^{(L)}\}$ with $\mathbf{X}^{(L)} = \mathbf{X}$, where $\mathbf{X}^{(l)} = [x_1^l \; x_2^l \; \cdots \; x_{2^l}^l]^T$ for $0 \leq l \leq L-1$ denotes a half-resolution version of $\mathbf{X}^{(l+1)}$ formed by successively summing two neighboring elements of $\mathbf{X}^{(l+1)}$ such that $x_i^l = x_{2i-1}^{l+1} + x_{2i}^{l+1}$ for $1 \leq i \leq 2^l$ (Fig. 3). Then, an important inequality related to the $L_1$ distance between two descriptors can be derived based on the triangle inequality such that

$$d_1(\mathbf{X}, \mathbf{Y}) \geq d_1^{(L-1)}(\mathbf{X}, \mathbf{Y}) \geq \cdots \geq d_1^{(1)}(\mathbf{X}, \mathbf{Y}) \geq d_1^{(0)}(\mathbf{X}, \mathbf{Y}) \qquad (3)$$

where $d_1^{(l)}(\mathbf{X}, \mathbf{Y}) = \|\mathbf{X}^{(l)} - \mathbf{Y}^{(l)}\|_1$ is the $L_1$ distance between the low-resolution version of $\mathbf{X}$ and $\mathbf{Y}$. The inequality property in Eq. (3) provides a significant clue to speedup the exhaustive search operation. That is, given a query descriptor $\mathbf{X}$ and a minimum distance of matching criteria $d_{\min}$, it is possible to eliminate all matching candidates in $\Omega_D$ whose $L_1$ distance in low-resolution domain satisfies $d_1^{(l)}(\mathbf{X}, \mathbf{Y}) \geq d_{\min}$. Since the computational cost of $L_1$ distance at lower levels is much less than that of $L_2$ distance in full-resolution domain, the run-time of the exhaustive search can be improved dramatically via this multi-resolution elimination method. We thus combine this multi-resolution elimination method with the LES method to form an efficient exhaustive search algorithm with low computational complexity.

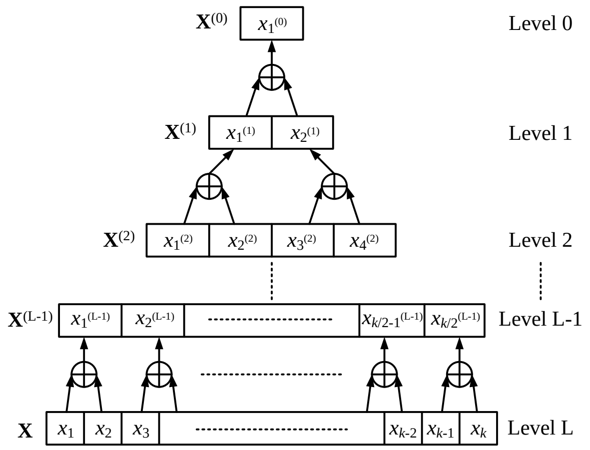



Figure 3. The $L_1$ norm pyramid of a feature descriptor **X** with dimension $k = 2^L$.

### *B. Minimum Distance Determination*

As mentioned in the previous subsection, the multi-resolution elimination method requires a minimum distance $d_{\min}$ as the matching criteria to eliminate any matching candidate in $\Omega_D$ for a given feature descriptor $\mathbf{X}$ in $\Omega_Q$. To determine the value of $d_{\min}$, the authors in [7] first select an initial best match $\mathbf{Y}_{ibm}$ whose norm $\|\mathbf{Y}_{ibm}\|$ is close to $\|\mathbf{X}\|$. Next, the value of $d_{\min}$ is initially computed as $d_{\min} = d_1(\mathbf{X}, \mathbf{Y}_{ibm})$ and is updated when a better match is found in the rest of candidates. This method is simple, but only effective in a norm-sorted database constructed in an off-line preprocess. In many real-time applications, however, the data set $\Omega_D$ is obtained on-line and is not norm-sorted. This problem therefore motivates us to develop a new minimum distance computation method to compute an appropriate value of $d_{\min}$ for the exhaustive search in a non-sorted data set.

Suppose that all multi-resolution descriptors of both query and data sets are built and stored in $2L$ multi-resolution tables. Let $\mathbf{T}_l^Q$ and $\mathbf{T}_l^D$, $0 \le l \le L-1$, denote the multi-resolution tables for the query and data sets, respectively. Unlike the previous method in [7], the proposed method aims to determine an appropriate minimum distance based only on the lowest resolution tables $\mathbf{T}_0^Q$ and $\mathbf{T}_0^D$, significantly reducing the computational burden of the minimum distance determination. Moreover, as the initial value of $d_{\min}$ determined, its value will keep as a constant for all query descriptors during the candidate elimination process, leading to faster exhaustive search performance but loss of matching accuracy. To maintain the matching accuracy, we remain the ratio test on two $L_2$ distances at the last step as suggested in the LES method. This is another main difference between the proposed method and the previous works.

The concept of the proposed minimum distance computation method is simple. First, we take a subset of $\Omega_Q$ as a training set. The value of $d_{\min}$ is then computed as the average of the $L_1$ distance at Level 0 between each descriptor in the selected subset and all descriptors in $\Omega_D$. More specifically, the value of $d_{\min}$ is obtained by

$$d_{\min} = \frac{1}{\bar{N} N_d} \sum_{i=1}^{\bar{N}} \sum_{j=1}^{N_d} d_1^{(0)}(\mathbf{X}_i, \mathbf{Y}_j) \text{ with } \bar{N} = \lfloor \beta N_q \rfloor \quad (4)$$

where $\lfloor x \rfloor$ means the largest integer less than or equal to $x$, and $\beta$ denote a nonzero positive number satisfying $0 < \beta \le 1$ to define the subset of $\Omega_Q$. Empirically, we found that setting the value of $\beta$ as 0.25 may obtains the best candidate elimination performance with less computational burden. We thus choose $\beta$ = 0.25 as the default setting for the proposed minimum distance computation method.

```
The Proposed Fast Multi-resolution Exhaustive Search Algorithm
Input: A query set ΩQ={X1,X2,…,XNq} and a data set
       ΩD={Y1,Y2,…,YNd}
Output: A match set ΩM
Initialize: Clear the match set ΩM={0}; Set a matching counter
  c=0; Set a distance ratio α=0.65;
Build multi-resolution tables T0^Q~TL-1^Q for the query set Q;
Build multi-resolution tables T0^D~TL-1^D for the data set D;
Determine the minimum distance dmin based only on the tables
  T0^Q and T0^D;
for i=1:Nq do
 d1=d2=FLT_MAX; xc=i;
 for j=1:Nd do
  if d1^(l)(Xi,Yj)>dmin, l=0~L-2 , then
   continue;
  else if d1^(L-1)(Xi,Yj)>dmin then
   dist=d2(Xi,Yj);
   if dist<d2 then
    d2=dist;
   end if
   continue;
  else
   dist=d2(Xi,Yj);
   if dist<d1 then
    if d1<d2 then
     d2=d1;
    end if
    d1=dist; yc=j;
   else if dist<d2 then
    d2=dist;
   end if
  end if
 end for
 if d1<αd2 then
 Push (xc,yc) into match set ΩM; c=c+1;
 end if
end for
Return: match set ΩM
```

Figure 4. Pseudo-code for the proposed multi-resolution exhaustive search algorithm.

### *C. The Proposed Algorithm*

Fig. 4 shows the pseudo-code of the proposed fast multi-resolution exhaustive search algorithm, which combines the LES method with the multi-resolution candidate elimination method. After initialization, the multi-resolution tables $\mathbf{T}_l^Q$ and $\mathbf{T}_l^D$ with $0 \le l \le L-1$ are built for all descriptors in both query and data sets. The value of $d_{\min}$ is then computed by using Eq. (4). The following exhaustive search procedure is similar to the LES method, except that we apply the multi-resolution examination process at the lower levels before the $L_2$ distance examination at the highest level. This helps to achieve high search speed without loss of accuracy, even in high dimensional spaces. The search performance of the proposed method will be validated in the experiments.

TABLE I.

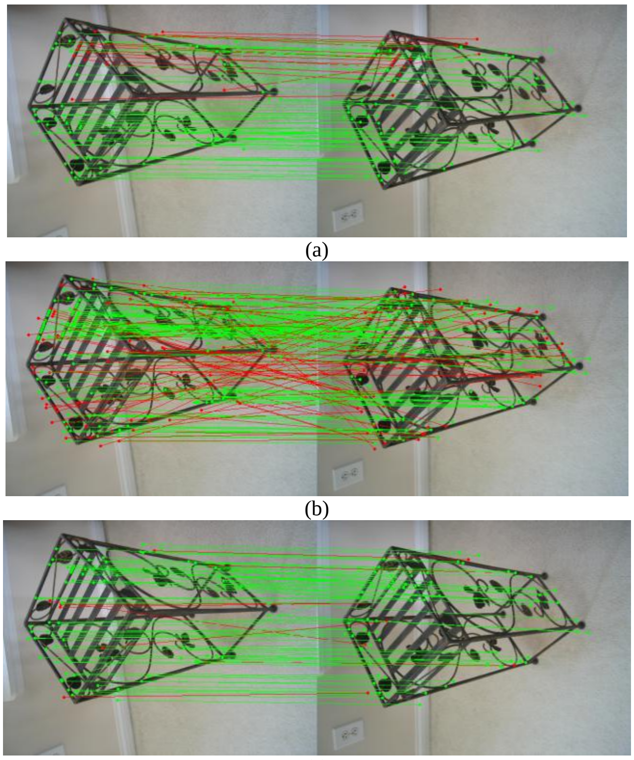
PERFORMANCE COMPARISONS OF THE PROPOSED METHOD WITH TWO EXISTING METHODS

| Image-Pair No. | Search Method | | | | | |
|---|---|---|---|---|---|---|
| | LES | | k-d tree | | Proposed method | |
| | Inlier ratio (%) | Search time (ms) | Inlier ratio (%) | Search time (ms) | Inlier ratio (%) | Search time (ms) |
| 1 | 97.66 | 27.28 | 77.98 | 102.58 | **98.05** | **2.04** |
| 2 | **91.89** | 70.96 | 54.14 | 128.50 | 91.67 | **3.18** |
| 3 | 98.87 | 33.75 | 83.51 | 205.58 | **100.00** | **1.91** |
| 4 | 94.44 | 22.45 | 62.83 | 72.16 | **95.12** | **1.30** |
| 5 | 84.44 | 48.66 | 64.50 | 142.75 | **90.91** | **2.42** |
| Avg. | 93.46 | 40.62 | 68.59 | 130.31 | **95.15** | **2.17** |

## IV. EXPERIMENTAL RESULTS

In the experiments, we combined the SURF keypoint detector [6] with the LES, k-d tree, and the proposed methods to evaluate their feature matching performance in a high dimensional space ($k$ = 64). Two performance metrics were used to evaluate the matching performance of each competing method, including an inlier ratio and a search time metrics. The former measures the ratio of the total inlier-point number to the total matching-point number, indicating the matching accuracy of the search algorithm. By contrast, the later measures the overall search time (in millisecond) spent on the matching process, indicating the computational efficiency of the search algorithm. An efficient feature matching algorithm should behave a higher inlier ratio with a small search time. Note that we applied an existing outlier removal method [11] to extract all inliers and compute the inlier ratio for each competing method.

Five test image-pairs were used in out testing, and Table I records the experimental results. It is clear from Table I that the proposed method provides the best inlier ratio and the smallest search time, followed by the LES and k-d tree methods. Moreover, the inlier ratio of the proposed method is about 95%, and its search performance is about nineteen times the LES method and about sixty times the k-d tree method. Thus, the proposed method is more suitable to meet the requirement of real time applications. Fig. 5 shows the feature matching results of the test image-pair No. 5. In Fig. 5, the green and red lines indicate, respectively, the inlier and outlier matches after the outlier removal process. We can see from Fig. 5 that the LES and k-d tree methods produce more outliers in their matching results when compared to the proposed method. The above experimental results therefore validate the search accuracy and computational efficiency of the proposed feature matching algorithm.



(a)

(b)

(c)

Figure 5. SURF descriptor matching results of test image-pair No. 5 obtained by (a) the LES method, (b) the k-d tree nearest neighbor search method, and (c) the proposed method. In the above images, the green and red lines indicate, respectively, the inlier and outlier matches after an outlier removal process [11].

## V. CONCLUSIONS AND FUTURE WORK

This paper proposes an efficient and accurate multi-resolution exhaustive search algorithm that achieves high feature descriptor matching accuracy with real-time performance. The proposed algorithm consists of a $L_1$-norm based multi-resolution candidate elimination technique, allowing efficiently removing all non-candidates in a large candidate list with less computational burden. A novel minimum distance determination method is also proposed to decide a minimum threshold value for eliminating candidates. Experimental results validate the performance of the proposed method by comparing with two existing search methods, quantitatively and effectively. In the future, combination with a SIFT detector will be further investigated.

## ACKNOWLEDGMENT

This work was supported by the National Science Council of Taiwan, R.O.C. under grant NSC 101-2221-E-032-022.

## REFERENCES

[1] S. A. Nene and S. K. Nayar, "A simple algorithm for nearest neighbor search in high dimensions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 989-1003, September 1997.

[2] J. L. Bentley, "Multidimensional binary search trees in database applications," *IEEE Transactions on Software Engineering*, vol. 5, pp. 333-340, July 1979.

[3] T. Uemiya, Y. Matsumoto, D. Koizumi, M. Shishibori, and K. Kita, "Fast multidimensional nearest neighbor search algorithm based on ellipsoid distance," *International Journal of Advanced Intelligence*, vol. 1, pp. 89-107, November 2009.

[4] R. Weber, H. Schek, and S. Blott, "A quantitative analysis and performance study for similarity−search methods in high-dimensional spaces," in *Proc. 24th Very Large Data Bases*, New York, USA, 1998, pp. 194-205.

[5] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91-110, November 2004.

[6] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Speeded-up robust features (SURF)," *Computer Vision and Image Understanding*, vol. 110, pp. 346-359, June 2008.

[7] S. C. Song, M. J. Kim, and J. B. Ra, "A fast multiresolution feature matching algorithm for exhaustive search in large image databases," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, pp. 673-678, May 2001.

[8] S. C. Song, and J. B. Ra, "Multiresolution descriptor matching algorithm for fast exhaustive search in norm-sorted databases," *Journal of Electronic Imaging*, vol. 14, pp. 1-7, October-December 2005.

[9] P. Moreels and P. Perona, "Evaluation of features detectors and descriptors based on 3D objects," *International Journal of Computer Vision*, vol. 73, pp. 263-284, July 2007.

[10] J. M. Kleinberg, "Two algorithms for nearest-neighbor search in high dimensions," in *Proc. 29th ACM Symposium on Theory of Computing*, Texas, USA, 1997, pp. 599-608.

[11] C. Olsson, A. Eriksson, and R. Hartley, "Outlier removal using Duality," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, San Francisco, CA, USA, 2010, pp. 1450-1457.

**Chi-Yi Tsai** was born in Kaohsiung, Taiwan, Republic of China in 1978. He received the B.S. and M.S. degree in electrical engineering from National Yunlin University of Science and Technology, Yunlin, Taiwan, in 2000 and 2002, respectively. He received the Ph.D. degree in electrical and control engineering, at the National Chiao Tung University, Taiwan, in 2008. He is currently an assistant professor in department of electrical engineering, Tamkang University, Taiwan. His research interests include image processing, color image enhancement processing, visual tracking control for mobile robots, visual servoing and computer vision.

**An-Hung Tsao** was born in Taipei, Taiwan, Republic of China in 1989. He received the B.S. degree in electrical engineering from Tamkang University, Taiwan, in 2011. He is currently pursuing the M.S. degree in electrical engineering at the Tamkang University, Taiwan. His research interests include image processing and computer vision.

**Chuan-Wei Wang** was born in Taipei, Taiwan, Republic of China in 1988. He received the B.S. degree in electrical engineering from Tamkang University, Taiwan, in 2011. He is currently pursuing the M.S. degree in electrical engineering at the Tamkang University, Taiwan. His research interests include point cloud registration and 3D modeling.