Computer Vision and Image Understanding 115 (2011) 649–667

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Robustly registering range images using local distribution of albedo

Diego Thomas *, Akihiro Sugimoto

National Institute of Informatics, Tokyo 101-8430, Japan

ARTICLE INFO

Article history:
Received 19 February 2010
Accepted 1 November 2010
Available online 14 January 2011

Keywords:
Range image
Registration
Albedo
Region growing
Rigidity constraint
Lambertian

ABSTRACT

We propose a robust method for registering overlapping range images of a Lambertian object under a rough estimate of illumination. Because reflectance properties are invariant to changes in illumination, the albedo is promising to range image registration of Lambertian objects lacking in discriminative geometric features under variable illumination. We use adaptive regions in our method to model the local distribution of albedo, which enables us to stably extract the reliable attributes of each point against illumination estimates. We use a level-set method to grow robust and adaptive regions to define these attributes. A similarity metric between two attributes is also defined to match points in the overlapping area. Moreover, remaining mismatches are efficiently removed using the rigidity constraint of surfaces. Our experiments using synthetic and real data demonstrate the robustness and effectiveness of our proposed method.

1. Introduction

The 3D modeling process of real objects has attracted increased interest during the past decade for applications in augmented reality, cinema, computer games, or medicine. Because it is labour intensive to create a detailed 3D model of a real object using various modeling software, automating the whole modeling process has attracted considerable interest in recent years. This process can be divided into five steps: (1) data acquisition, (2) reconstruction of 3D images, (3) 3D registration, (4) merging, and (5) inverse rendering.

Recent acquisition devices, like modern laser range scanners, can retrieve both the 3D shape and color image of an object from a fixed viewpoint; the acquired 3D image in this case is called a range image (Fig. 1). Therefore the second step of the 3D modeling process can be omitted for range images. However, as some parts of an object are occluded from a fixed viewpoint, multiple viewpoints are required to obtain the full 3D shape of the object. Therefore, 3D images of partially overlapping parts of the object, acquired from different viewpoints have to be aligned. This process is called 3D registration. There are two categories for 3D registration. The first, called coarse registration, is to find rough alignment between two 3D images, starting from sufficiently different poses [1]. The second, called fine registration, is to find accurate alignment between two 3D images, starting from rough alignment. We refer to range image registration when using range images [2].

The most common approach to registering range images is to find correspondences in points between two overlapping range images and then accordingly estimate the transformation in aligning the two range images. Several methods for registering range images can be found in the literature that use geometric features for computing correspondences in points. However, we assume that the range images to be registered have simple textured shapes (like cylinders) and are thus devoid of salient geometric features. Consequently, photometric features in addition to geometric features are required to compute correspondences in points.

Reflectance properties as a photometric feature are promising because of their independence of the pose of the object relative to the sensor. Retrieving these properties has provided a major research area in physics-based vision called reflectance from brightness (with a known shape and illumination). Cerman et al. [3] recently proposed a method, which we call ICP using albedo (ICPA), using the reflectance properties (which is the albedo for Lambertian objects) of the object surface in the standard iterative closest point (ICP) process. The illumination conditions have to be precisely known a priori so that the reflectance of the surface of an object can be accurately retrieved from its shape and brightness. Consequently, the direct use of albedo values as a matching constraint, as achieved by ICPA, requires global illumination to be accurately estimated, which is difficult to attain in practice under real illumination conditions.

We introduce a region-based approach to using reflectance attributes, namely the albedo, for robust fine registration of Lambertian objects under rough estimates of illumination. Because retrieving the albedo on the surface of a Lambertian object is sensitive to estimates of illumination, the albedo of a point cannot be directly used under rough estimates of illumination. We thus

* Corresponding author.
E-mail addresses: diego_thomas@nii.ac.jp (D. Thomas), sugimoto@nii.ac.jp (A. Sugimoto).

1077-3142/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.cviu.2010.11.016

(a) *can00* (b) *can20*

Data *can*.

(c) *box00* (d) *box20*

Data *box*.

(e) *hand00* (f) *hand20*

Data *hand*.

(g) *candy00* (h) *candy20*

Data *candy*.

Fig. 1. Examples of range images with color images used in this paper. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

employ the local distribution of albedo for registration. Our proposed method uses adaptive regions to model the local distribution of albedo on the object surface, which leads to robust extraction of attributes against illumination estimates. These regions are grown using a level-set method, allowing us to exclude outliers and then to define more reliable attributes. We define a robust metric, using the principal component analysis (PCA) of each region to find correspondences in points. This is a stable and powerful metric to maximize the number of correct matches, even under rough estimates of illumination. Moreover, we reject remaining mismatches by enforcing the rigidity constraint on surfaces and then estimate transformation using the weighted least squares (WLS) method. Our method has advantages with rough estimates of illumination and with large amounts of noise. These advantages allow us to use simple models of illumination to register range images. Our experiments using synthetic and real data demonstrate that our method is robust. We assume in this paper that the surfaces' textures present sufficient saliency to constrain the matching of two overlapping range images. We do not consider uniform nor 'salt and pepper' textures, nor repetitive patterns. We also assume

that the objects do not present self-occlusions, shadows nor interreflections.

2. Related work

During the past few decades, many approaches to registering range images have been discussed. The most well-known approach to fine registration is the iterative closest point (ICP) [4,5]. This method iterates two steps: it matches each point of the first range image with its closest point on the other range image, and estimates the transformation between the two range images using the correspondences in the matched points. The ICP converges monotonically to a local minimum and therefore requires rough alignment. Many discriminative geometric features have been proposed [6–10] to achieve robust and accurate alignment. Other approaches based on graduated assignment such as the ICP Markov Chain (ICPMC) [11] use maximization of entropy to define a probabilistic model for correspondences between points. However, even though such methods can deal with complex 3D shapes, they do not work well for simple textured shapes (like cylinders), which are devoid of salient geometric features.

To overcome the problem with shapes devoid of salient geometric features, many approaches using photometric features have also been discussed. Godin et al. [12] proposed using the dense attributes of range image elements as matching constraints. Weik [13] proposed using texture intensity gradients and intensity differences. Johnson and Kang [14] proposed dealing with textured 3D shapes by using color. Okatani and Sugimoto [15] proposed using chromaticity for registration. Brusco et al. [16] proposed incorporating texture information in the concept of spin-images. Pulli et al. [17] proposed new mismatch error to improve registration using both color and geometric information. However, because the color or chromaticity of a Lambertian surface depends on the pose of the object and illumination conditions, these methods perform poorly when the change in pose significantly affects the appearance of the object.

However, the albedo is a photometric property that is unaffected by the pose of the object, the illumination conditions, or the viewpoint, and is thus useful for matching. Cerman et al. [3] proposed using the albedo difference to match points to register range images. However, this point-based approach is sensitive to data noise and requires detailed knowledge on illumination conditions. Therefore it cannot be applied in practice to real data.

In other approaches, Lowe [18] proposed a scale-invariant feature descriptor called the scale-invariant feature transform (SIFT) that makes use of differences in Gaussian functions to define the features for key point. Some variations of SIFT have been proposed to speed up the algorithm while maintaining the same accuracy. For example, Bay et al. [19] proposed a descriptor called speeded-up robust features (SURF) that makes use of an integral image to speed up computation and comparison. Tola et al. [20] also proposed a local descriptor that can be quickly computed and even be used in poor quality images. However, as these techniques did not directly use available 3D information but used 2D information projected from 3D, they were sensitive to texture deformation caused by projection. In fact, the same texture captured from different viewpoints produces differently deformed 2D textures, which makes the use of these techniques problematic. Moreover, these approaches focused more on computational efficiency than on accuracy.

In contrast to previous work, the method we propose can handle changes in photometric appearance even under rough estimates of illumination. It is robust to data noise and can thus be easily applied to real data. Moreover, it makes use of the albedo distribution as well as normals and 3D point positions, which leads to accurate and robust results.

3. Proposed method

Our proposed method uses the local distribution of albedo on the surface to define discriminative attributes that are robust to data noise. We define a similarity metric to efficiently match points in the overlapping part of two range images and use the rigidity constraint of surfaces to refine matching. The transformation aligning two range images is then computed using the WLS approach.

3.1. Overview of proposed method

The registration process is iteratively carried out by successively estimating rigid transformation, until a convergence criterion is satisfied or a maximum of iterations is completed. Matches are obtained by evaluating the similarities between attributes of points. These attributes are defined by adaptive regions representing the local distribution of albedo on the surfaces of objects. That is, each region represents the shape of the texture patterns around a point. The region grows inside homogeneous areas and stops at the border of pattern boundaries. The transformation is then estimated by minimizing the distances between matching points. Fig. 2 has a flowchart of our proposed approach.

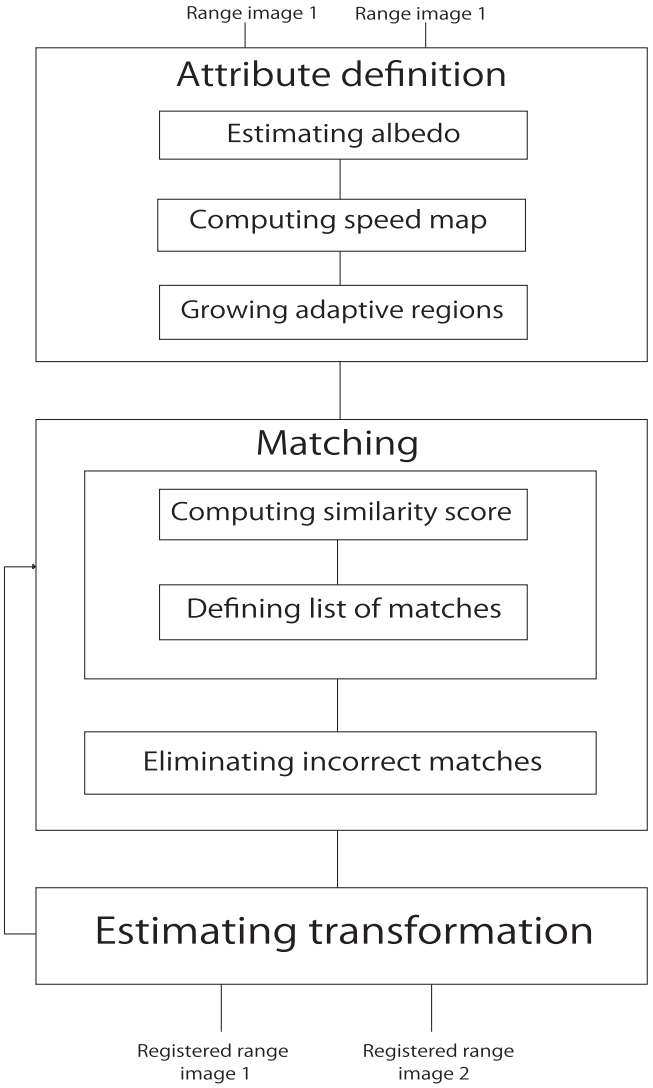


Fig. 2. The flowchart of our proposed method.

3.1.1. Generation of adaptive region

Because point-based matching methods are sensitive to data noise, we propose using information from a local distribution of albedo. We define attributes, viz., a region, for each point of two range images that represent the local distribution of albedo around the point. The region is adaptively defined using a level-set approach and transformed into their local coordinate system defined by the principal axis.

3.1.2. Evaluation of similarities using the albedo

We define a similarity metric to match points based on their attributes to estimate correspondences in points between two range images. This similarity metric represents the photometric similarities of attributes weighted by their geometric similarity. We define a list of possible matches for each point based on similarity scores and then dynamically create a list of matches.

3.1.3. Elimination of incorrect matches

We eliminate possibly incorrect matches by using the rigidity constraint of the surfaces. That is, the distance between two points on the surface does not depend on the pose of the object. Consequently, the distance between two points in one range image and their two corresponding points in the other range image should be the same. If not, the rigidity constraint is violated. We test the rigidity constraint for a match with all other matches and if the number of violations exceeds a tolerance threshold, the match is eliminated.

3.1.4. Estimation of rigid transformation

The matching list that is obtained is used as the input of a WLS algorithm to compute the current rigid transformation aligning two range images.

This framework allows simple textured range images to be robustly and accurately registered even with large amounts of data noise and rough estimates of illumination. We explain our method in more detail in what follows.

3.2. Generation of adaptive region

We define a region for each point of the two range images to obtain reliable attributes for each to find correspondences between points. The main idea here is to obtain a reliable representation of the local distribution of albedo. Therefore, these regions should be adaptively defined depending on the local distribution of albedo around the point of interest. Level-set methods, which are widely used for segmentation, appear to effectively model complex shapes in textured images and are robust to data noise. Therefore, we adaptively grow regions using a level-set method.

3.2.1. Level-set method

A region is defined by a contour that we define with a level-set method (fast marching algorithm [21]). A contour is defined as the zero level-set of a higher dimensional function called the level-set function, $\psi(X,t)$ (see Fig. 3). The level-set function is then evolved under the control of a differential equation. The evolving contour can be obtained at any time by extracting the zero level-set $\Gamma(t) = \{X \,|\psi(X,t) = 0\}$.

We use a simple form of the level-set equation:

$$\frac{d}{dt}\psi = -P(x)\|\nabla\psi\|, \tag{1}$$

where P is a propagation (expansion) term. This propagation term of the level-set function is next defined in terms of a speed image. In our approach, the propagation of the contour is defined using the gradient of the albedo such that the propagation term is high in uniform areas and low close to pattern boundaries. We define the zero level-set as the contour propagated at a certain time T (for example, $T = 0.2$ s).

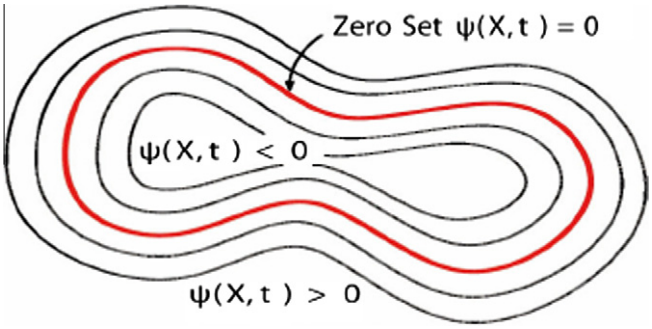


Fig. 3. Concept of zero set in a level-set.

3.2.1.1. Speed image. A speed image represents how fast a contour can propagate for every point of the range image. This speed should depends on the homogeneity of the albedo for every point compared with their local neighborhood. The speed image in the proposed method is computed from the gradient magnitude image of albedo that is obtained by applying the Gradient Magnitude filter to the albedo image.[1] Mapping should be done in such a way that the propagation speed of the front is very low with high albedo gradients while it speeds up in low gradient areas. This arrangement makes the contour propagate until it reaches the edges of albedo patterns and it then slows down in front of these edges.

We employ the Sigmoid function, S, to compute the speed image since it provides numerous control parameters that can be customized to shape an accurate speed image. In fact, it has a mechanism for focusing attention on a particular set of values and progressively attenuating the values outside that range.

$$S(I) = \frac{1}{1 + e^{-\left(\frac{I-\beta}{\alpha}\right)}}, \tag{2}$$

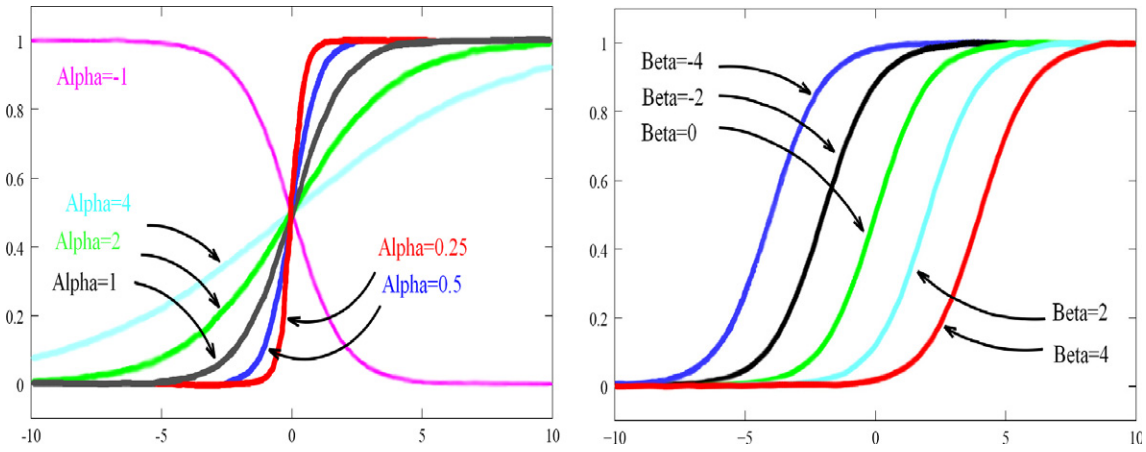
where I is the gradient magnitude image of albedo, α defines the width of the gradient intensity range, and β defines the gradient intensity around which the range is centered (see Fig. 4). As suggested by Ibanez et al. [21], α and β are found as follows. For the gradient magnitude image, let us denote the minimum value along the contour of the albedo patterns as $K1$. We denote the average value of the gradient magnitude in the homogeneous regions of the albedo image as $K2$. Then, β is $\frac{K1+K2}{2}$ while α is $\frac{K2-K1}{6}$. Fig. 5 shows a concrete example of the albedo, gradient and speed images obtained using a synthetic data.

3.2.2. Generation of regions

For each point p, we define a time image T_p. For each pixel x of T_p, $T_p(x)$ represents the time required for the level-set function to propagate from p to x. Starting from point p, a region is grown by using the four-neighborhood and by adding points into the region, such that T_p on these points is less than a threshold (e.g., 0.2 sec) (see Fig. 6). The maximum size of the region is enforced, which allows us to discriminate points in excessively homogeneous areas.

This region grows in homogeneous areas and stops in front of the contour of albedo patterns. Consequently, while the size of

[1] We remark that in the case of general shape models, points may not be aligned in a grid manner and the Gradient Magnitude Filter cannot be directly applied. However, it is still possible to compute the gradient magnitude image as far as the neighborhood relationship between points is provided. In fact, we can compute the gradient magnitude of a point as the sum of the differences in albedo between the point and its adjacent points. The obtained gradient magnitude image represents the local homogeneity of albedo and can thus be effectively mapped to the speed image.

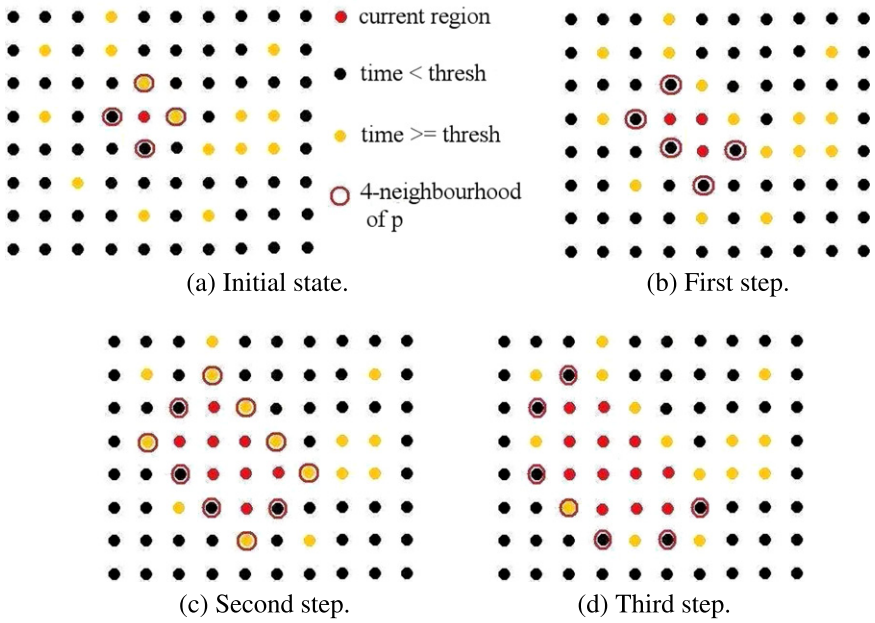


(a) α defines width of window. (b) β defines center of window.

Fig. 4. Effects of various parameters in Sigmoid function.

(a) Albedo image. (b) Gradient image. (c) Speed image.

Fig. 5. Example of albedo, gradient and speed images obtained using synthetic data.



(a) Initial state. (b) First step.

(c) Second step. (d) Third step.

Fig. 6. Adaptively defined region using four-neighbourhood.

the region increases, the homogeneity of the region is preserved. Moreover, the growth of the region is adapted to the distribution of albedo and to data noise in the neighborhood of each point.

We then transform each region into their local coordinate system so that the comparison between two regions becomes independent of the pose of the object. That is, we transform a region into the coordinate system defined by the normalized principal axis computed for this region. Given point x and its region $R(x)$, the PCA is carried out to obtain the eigenvalues and eigenvectors of the covariance matrix of $R(x)$. The three eigenvectors are then normalized to define a new basis $B = (e_1, e_2, e_3)$ where e_1 corresponds to the normalized eigenvector with the greatest eigenvalue and e_3 corresponds to the eigenvector with the smallest eigenvalue. $R(x)$ is then transformed to the new basis B to obtain the final attribute of x. We remark that the transformation is done independently of the color of points inside the region.

As a result, a reliable region is adaptively generated depending on each point. The local distribution of albedo of 3D points inside this region defines an attribute for each point. The attributes obtained in this way enhance robustness in evaluating similarity to find correspondences. Figs. 7 and 8 show a concrete example of the different steps in the region growing process. Fig. 7 shows the initial state. Two corresponding points in two overlapping range images are regarded as seed points for the growth of the regions, and a maximum size of the regions is enforced. Fig. 8 shows the regions at different time. Namely after the first, the third, the sixth and the tenth iteration of the region growing process. As expected, we observed that the regions of two corresponding points grow in a similar way.

3.3. Restrictions on search area

Poor estimates of the albedo are obtained with rough estimates of illumination. These will particularly be far less useful around the border of the range images. Consequently, we do not take into consideration points near the borders of the range images to reduce their influence. We denote the restricted area of the range image, P, as $C(P)$.

Moreover, we dynamically control a search area (area where a possible match for p will be searched), $\Omega(p)$, for each point $p \in P$ in the other range image Q, whose center is the projection of p on Q (i.e., the point of Q closest to p in the sense of the Euclidean distance). $\Omega(p)$ is defined such that the closer to convergence registration becomes, the smaller $\Omega(p)$ becomes. This control enhances the stability and accuracy of registration. $\Omega(p)$ is defined independently of $C(Q)$.

Points in large homogeneous areas are not discriminative enough to be used in registration. Such points are detected using the size of their regions. Indeed, the regions of such points are close to the maximum given beforehand. Therefore we do not take into account points whose regions are larger than 95% of the maximum size of regions.

3.4. Evaluation of similarities using the albedo

We define a similarity metric between two points using their attribute to find correspondences across two range images.

Letting p be a point in $C(P)$ and q be a point in $\Omega(p)$, we denote the regions corresponding to p and q by $R(p)$ and $R(q)$, respectively. For each point $m \in R(p)$, we define its corresponding point $n(m)_q \in R(q)$ (Fig. 9). The corresponding point $n(m)_q$ is defined by

$$\arg\min_{x \in R(q)} \left(\left\| T(\overrightarrow{pm}) - \overrightarrow{qx} \right\|_2 \right). \tag{3}$$

For each pair $(m, n(m)_q)$, we define a weight $\omega_{(m,q)}$ such as

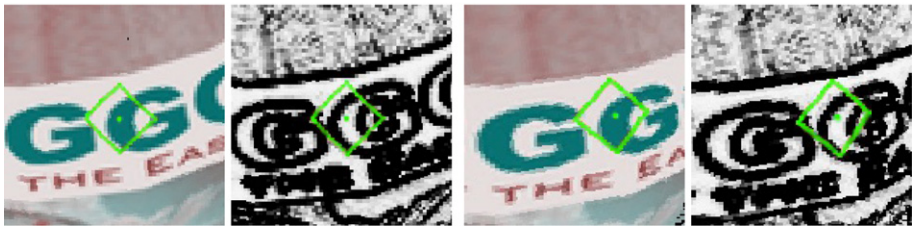
$$\begin{aligned} \omega_{(m,q)} &= 0 \quad \text{if} \quad \left\| T\left(\overrightarrow{pm}\right) - \overrightarrow{qn(m)_q} \right\|_2 > thresh, \\ \omega_{(m,q)} &= 1 \quad \text{if} \quad \left\| T\left(\overrightarrow{pm}\right) - \overrightarrow{qn(m)_q} \right\|_2 \leqslant thresh, \end{aligned} \tag{4}$$

where *thresh* is a distance threshold (e.g., 0.4 mm if the resolution of range images is 0.5 mm). We can similarly define the corresponding point and weight for each point in $R(q)$.

The similarity function between two points p and q is then defined as the weighted sum of the differences of the albedo of corresponding pairs:

(a) Range image 1. (b) Range image 2.

Initial seed points in two range images (in red).



(c) Range image 1. (d) Range image 2.

Maximum size of the regions (in vivid green).

Fig. 7. Initial state for the growth of the regions of two matched points in two range images.

(a) Range image 1. (b) Range image 2.

The regions at the first iteration (in red).

(c) Range image 1. (d) Range image 2.

The regions at the third iteration (in red).

(e) Range image 1. (f) Range image 2.

The regions at the sixth iteration (in red).

(g) Range image 1. (h) Range image 2.

The regions at the tenth iteration (in red).

Fig. 8. The growth of the regions at different time.

$$L(p,q) = \frac{size(R(p)) + size(R(q))}{\left(\sum_{m\in R(p)}\omega_{(m,q)} + \sum_{m\in R(q)}\omega_{(m,p)}\right)^2} \times \left\{ \sum_{m\in R(p)} \omega_{(m,q)} \left\| \overrightarrow{alb(m)} - \overrightarrow{alb(n(m)_q)} \right\|_2^2 + \sum_{m\in R(q)} \omega_{(m,p)} \left\| \overrightarrow{alb(m)} - \overrightarrow{alb(n(m)_p)} \right\|_2^2 \right\}, \quad (5)$$

where $size(R(\cdot))$ is the number of points in $R(\cdot)$ and $\vec{alb}(m)$ is the albedo vector of point m, computed using the Lambertian model of reflectance for each color channel:

$$\overrightarrow{alb(m)} = \frac{\overrightarrow{c(m)}}{\overrightarrow{norm(m)}^{\top} M \overrightarrow{norm(m)}}, \quad (6)$$

where $\overrightarrow{norm(m)}$ is the normal of the surface at point m, M is the illumination matrix and $c(m)$ is the RGB vector of point m.

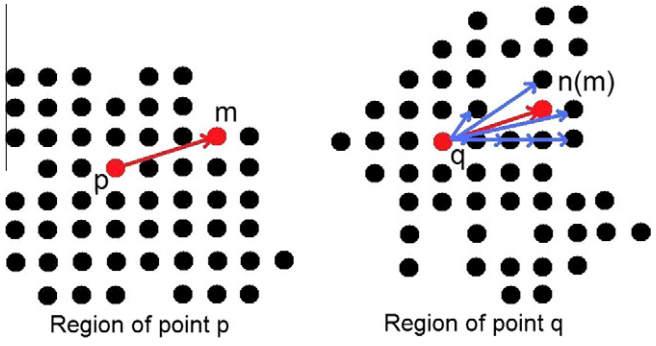


Fig. 9. Searching for corresponding point of m.

If p and q are matches and two regions $R(p)$ and $R(q)$ represent the same part of the object viewed from different viewpoints, then $m \in R(p)$ and $n(m)_q \in R(q)$ are two sampled points with small

distance (the distance between the two points is smaller than the resolution of the range images) viewed from different viewpoints. Thus, their albedo is likely to be similar. Therefore, the function, L, becomes small for points p and q. In contrast, L increases for points with different regions. As we can see, support by corresponding points inside the region define the similarity between two points of interest. This leads to similarity being robustly and stably evaluated.

Note that if $\sum_{m\in R(p)}\omega_{(m,q)}$ or $\sum_{m\in R(q)}\omega_{(m,p)}$ is less than $0.6\times size(R(p))$, *resp.* $0.6\times size(R(q))$, then the pair, (p,q), is not considered to be a possible match. This is because if (p,q) is a correct match, we can expect that there will be a sufficient number of matches between $R(p)$ and $R(q)$. Moreover, considering computational time, if the bounding boxes of $R(p)$ and $R(q)$ differ (up to the resolution of the range images), then the pair, (p,q), is not considered to be a possible match.

3.5. Matching

We dynamically create a matching list based on similarity scores computed as explained above. We search for a set of matches such that each point has at most one corresponding point and that the sum of the scores between all matches is minimized.

We compute a list of possible matches for each point sorted in the ascending order of similarity scores. Taking into consideration computational time, we enforce a maximum tolerance threshold for possible matches. The matching list is then iteratively and dynamically constructed. The match with the lowest similarity score at each iteration is chosen and added to the matching list. The two matching points are then removed from all the lists of possible matches and these lists are updated accordingly (resorted). We iterate this process until no more possible matches remain to obtain the final matching list.

At the end of this step, we have a reliable and consistent list of matches that does not contain any isolated points. Indeed, the region grown from an isolated point is empty and this point will not be a candidate for any match.

3.6. Elimination of incorrect matches

The list of matches that is obtained cannot be always directly used as input in the step to estimate transformation. This is because large amounts of noise or repetitive patterns in the albedo distribution may cause incorrect matches. We therefore remove such incorrect matches to enhance the robustness of estimating transformation further. To evaluate the accuracy of matches, we use the rigidity constraint of surfaces. This is because the rigidity constraint does not depend on the intensity or normals and it is therefore robust against data noise.

For two corresponding pairs, (p,q) and (p',q'), in the range images P and Q, we consider point pairs (p,p') and (q,q'), which represent the same points viewed from different viewpoints. Assuming that surfaces are rigid, we can see that distances $\left\|\overrightarrow{pp'}\right\|_2$ and $\left\|\overrightarrow{qq'}\right\|_2$ should not differ too much. That is, we define d by representing the difference between $\left\|\overrightarrow{pp'}\right\|_2$ and $\left\|\overrightarrow{qq'}\right\|_2$:

$$d=\left|\left\|\overrightarrow{pp'}\right\|_2-\left\|\overrightarrow{qq'}\right\|_2\right|. \tag{7}$$

If (p,q) and (p',q') are correct matches, then d should be smaller than a threshold, *Tdist* (e.g., 1.0 mm, for a resolution of 0.55 mm). This gives us the rigidity constraint (see Fig. 10).

Each pair in the list of matches is evaluated with all the other pairs in the list. If the number of pairs that violates the rigidity constraint exceeds a certain percentage, *Perc* (e.g., 50%), of the pairs, then the current pair is considered to be an incorrect match and is removed from the list.

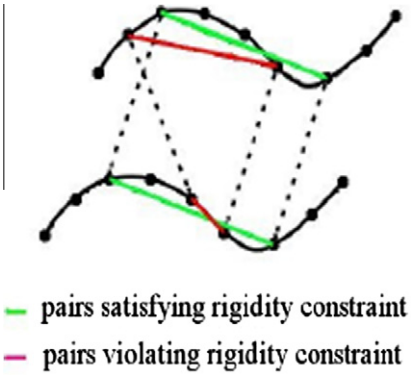


Fig. 10. Principle of rigidity constraint.

Assuming that the majority of the obtained pairs are correct matches, this method efficiently removes incorrect matches from the list of pairs obtained in Section 3.4. To handle cases where the majority of matches are incorrect, we dynamically update the parameters, *Tdist* and *Perc*, in such a way that the elimination step is tolerant to mismatches at the beginning of registration, and stringent against mismatches at the end of registration. This is because, at the beginning of registration, we may have a large number of mismatches and only a rough estimate of the transformation is sufficient. In contrast, as registration becomes more accurate, there are less mismatches and we aim at eliminating as many mismatches as possible to refine the estimation of transformation.

Our proposed approach is simpler than RANSAC in implementation. In fact, the RANdom SAmple Consensus (RANSAC) algorithm iterates the following three steps until the probability of finding the correct transformation becomes sufficiently large. (1) Three matches are randomly chosen among all matches in the list. (2) The rigid transformation aligning the three matches is estimated. (3) All the other matched points in one range image are transformed to the other range image using the estimated transformation and the consistency of matched points with the list is evaluated. The transformation that is the most consistent with the list of corresponding points is identified as the solution, and matches inconsistent with the solution are identified as outliers. On the other hand, our proposed technique to eliminate incorrect matches requires only to verify the validity of the rigidity constraint for each pair of matches. We remark that our technique needs $O(n^2)$ operations to remove incorrect matches.

3.7. Estimation of rigid transformation

We use the WLS method [22] to estimate transformation as accurately as possible. It weights each pair with the Euclidean distance between two corresponding points during the least squares minimization. These weights represent the feasibility of the correspondence of paired points. Minimization is iterated while updating the weight of each pair. The resulting transformation obtained with this method is more accurate than that with the standard least squares method.

4. Computational complexity analysis

Our proposed algorithm has its input of two range images with N points and outputs the rigid transformation aligning the two range images. Here we briefly give analysis on the computational complexity to our proposed algorithm. We refer to Fig. 2 for the

different steps of our method and give the computational complexity for each of these steps.

The first part of our proposed method (Attribute definition) takes $O(N)$ operations. We first estimate the albedo of each point of the two range images using Eq. (6), which takes $O(N)$ operations. In order to compute the speed images, gradient images and sigmoid images are computed, which is also done in $O(N)$ operations. Let Cst denote a given maximum number of points for a region. Growing the region of a point takes $O(Cst)$ operations (because we do not compute the propagation time outside the maximum region). Therefore, computing the region for each point of the two range images takes $O(NCst)$ operations. Because Cst is fixed during the process, the computational complexity of the Attribute definition is finally $O(N)$.

The second part of our method (Matching) takes $O(N^3\log(N))$ operations. For a point p, the corresponding point is searched in $\Omega(p)$, where $\Omega(\cdot)$ is defined in Section 3.3. Let ω be the size of $\Omega(\cdot)$. For one possible match, the similarity score is computed in $O(Cst^2)$ operations. Thus computing all similarity scores takes $O(N\omega Cst^2) = O(N\omega)$ operations (Cst is constant). The following loop is then executed until no possible match exists: (1) The list of possible matches is sorted ($O(N\omega\log(N\omega))$ operations). (2) The best match is taken and the two corresponding points are removed from the list ($O(N\omega)$ operations). (3) Go to (1) with $O((N-2)\omega)$ possible matches. The computational complexity of this loop is $O\left(\sum_{i=0}^{i=N/2} i\omega\log(i\omega)\right)$. We observe that

$$\sum_{i=0}^{i=N/2} i\omega\log(i\omega) \leqslant N\omega\left(\log\left(\prod_{i=0}^{i=N/2} i\right) + N\log(\omega)\right) \leqslant N\omega\left(\log(N^N) + N\log(\omega)\right) \leqslant N^2\omega\log(N\omega). \quad (8)$$

As a consequence, the loop is executed in $O(N^2\omega\log(N\omega))$ operations. Even though in practice ω is monotonically decreasing at each iteration, ω remains constant dominated by N in the worst case. Thus $\omega = O(N)$ and the computational complexity of the loop becomes $O(N^3\log(N^2)) = O(N^3\log(N))$. We then eliminate incorrect matches with $O(N^2)$ operations. Thus the computational complexity of the "Matching part" is finally $O(N^3\log(N))$.

The computational complexity of the Weighted Least Square algorithm is $O(N^2)$ and thus the total computational complexity of our proposed algorithm is: $O(N) + O(N^3\log(N)) + O(N^2) = O(N^3\log(N))$.

In this paper, we focus on accuracy of the transformation estimation and leave reducing the computational cost for future work. However, some ideas are available to reduce the computational cost of our algorithm in implementation. The computation of each point's attribute or similarity score is done independently and thus our method can be highly parallelized. Theoretically, with an infinite number of processors, the computational complexity of the "Attribute definition" part could go from $O(N)$ to $O(1)$ if parallelized. With modern graphic cards, a large number of processors available would drastically improve the computational time of our technique. It would also be interesting to use a random algorithm to select possible matches, as proposed in [12] for example. This would decrease the number of possible matches and thus speed up the algorithm.

5. Experiments

We evaluated our method using synthetic and real data and compared it with ICPA and ICP using both chromaticity and geometric features (which we call ICP-CG). This comparative study is thus useful for determining the effectiveness of different methods of registering overlapping range images of Lambertian surfaces devoid of salient geometric features. We selected these two methods for two main reasons:

- ICPA is, to the best of our knowledge, the most recent method that uses the albedo for registering overlapping range images. Because the albedo is invariant to changes in object poses under fixed illumination and fixed viewpoints, it is useful for computing correspondences in points between overlapping range images.
- ICP-CG[2] is the standard algorithm for registering overlapping range images. Because chromaticity is tolerant to some extent against changes in illumination, using chromaticity in addition to geometric features improves the accuracy of registration.

Because the objects used in our experiments were simple in shape (like cylinders), registration using only geometric characteristics did not work well. Consequently, standard root mean square (RMS) point-to-point Euclidean error was not relevant for evaluating the registration results in our case. This paper discusses our evaluation of the registration results by comparing the transformation we obtained with the ground truth transformation.

We use an angular measure of errors for rotation like Barron et al. [23] and the Euclidean error for translation. Let (R_g, T_g) be the ground truth transformation and (R_e, T_e) be the estimated transformation with R_g, R_e rotations and T_g, T_e translations. A rotation, $R = cos\left(\frac{\alpha}{2}\right) + \vec{u}\,sin\left(\frac{\alpha}{2}\right)$, is represented using quaternions, where α is the angle of rotation and $\vec{u}$ is the unit vector representing rotation axis. Let res be the resolution and d the depth of range images, we define err, which is the error of the obtained transformation as:

$$err = \frac{\Theta d + \|T_g - T_e\|}{res}, \quad (10)$$

where Θ is the angle between the normalized ground truth rotation $\frac{R_g}{\|R_g\|}$, and the normalized estimated rotation, $\frac{R_e}{\|R_e\|}$. The err represents the error in terms of neighboring points. It is thus an objective and informative criterion to evaluate the accuracy of the different methods of registration. All results from these experiments have been presented with the estimated albedo image. We compute the resolution res of a range image as the average of the 2D distances between adjacent points projected onto the 2D grid induced by the laser scanner (the depth is not considered). Note that the resolution is not computed based on neighboring relationship in the 3D space.

5.1. Definition of parameters

ICPA and ICP-CG require three parameters: *max_it* (the maximum iterations for registration), *conv_thresh* (the threshold for convergence), and *percentage* (the percentage of matches to be eliminated).

Our proposed method requires five parameters: *max_it*, *conv_thresh*, and *thresh* (the threshold to grow the regions), *max_size* (the maximum size of a region), and *tol_thresh* (the threshold for tolerance to similarity). The two thresholds, *Tdist* and *Perc*, are

[2] The ICP-CG algorithm is an ICP algorithm that uses Euclidean distance and chromaticity similarity to find corresponding points. At iteration k, the current transformation $Trans$ aligning the two range images is estimated. For each point p of the range image P, the projection q of point $Trans(p)$ on the range image Q is found and then the closest point to p in the sense of chromaticity distance is found in the region $\Omega(p)$ as the corresponding point of p, where $\Omega(p)$ is defined in Section 3.3. If $\sigma_R(x)$, $\sigma_G(x)$ and $\sigma_B(x)$ are the red, green and blue channels for the chromaticity of a point x, then the chromaticity distance $dchrom(p, q)$, between two points p and q is:

$$dchrom = \sqrt{(\sigma_R(p) - \sigma_R(q))^2 + (\sigma_G(p) - \sigma_G(q))^2 + (\sigma_B(p) - \sigma_B(q))^2}. \quad (9)$$

The obtained list of corresponding pairs is then sorted and only a certain percentage (70% in the experiments) of the pairs with the best score is kept.

Table 1
Description of synthetic data.

Nb_Points	Resolution	Expected_rot (angle; axis)	Expected_trans
30,650	0.01 mm	(18.00; 0.01, 0.99, −0.03)	(−0.02, 0.00, −0.01)

(a) First image. (b) Second image.

Fig. 11. Input synthetic data.

(a) Superimposed input images. (b) Registration results with albedo image.

Fig. 12. Results obtained with our proposed method in ideal case.

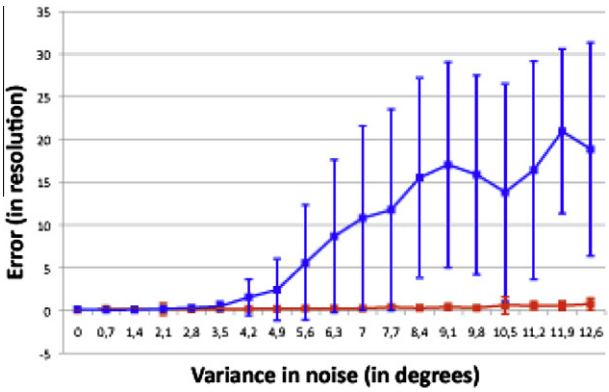


Fig. 13. Results for various different illuminations (red: our method, blue: ICPA).

dynamically defined for the elimination step depending on the current state of registration. At the beginning of registration, *Tdist* = 8 × "resolution of the image" and *Perc* = 70%, and at the end of registration, *Tdist* = 2 × "resolution of the image" and *Perc* = 30%. The tolerance *tol_thresh* was set to 0.01 for synthetic data and to 0.5 for real data. We remark that the restricted search area in Section 3.3 was equally applied to our method, ICPA and ICP-CG.

5.2. Evaluation with synthetic data

We conducted experiments with synthetic data to test the robustness of the proposed method against data noise and changes in illumination. The synthetic data were obtained with 3D modeler software (3D Studio Max) (see Table 1). The exact albedo image and the exact illumination, modeled with a direction and a color vector, were known. The illumination represents a single distant light-source point. Assuming the object to be Lambertian, we rendered the brightness at the surface using the Lambertian reflection model. That is, given the albedo ρ and normal $\vec{norm}$ at a surface point, and light-source direction $\vec{l}$ and light-source brightness L, the brightness, I, at this point is computed as: $I = \rho L \vec{norm} \cdot \vec{l}$. We estimated the albedo using an approximation of the exact illumination to test the robustness of our proposed method (see Figs. 11 and 12).

We manually established a rough pre-alignment of the two range images before applying our method, which allowed us to

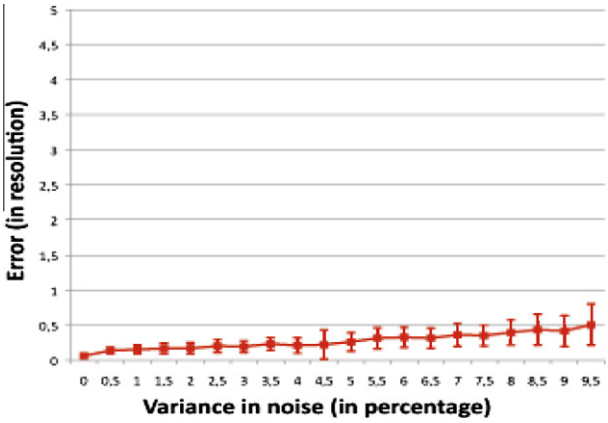


Fig. 14. Results for noise in intensities with our method.

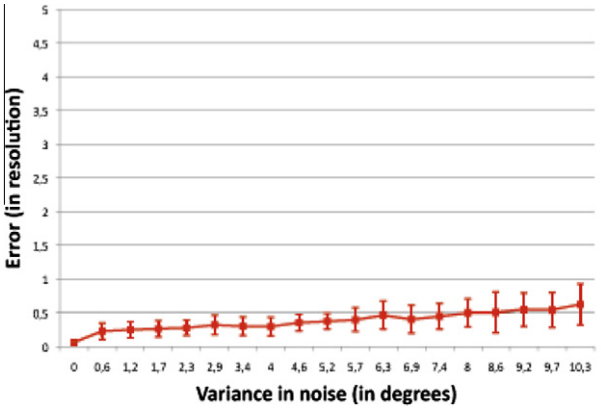


Fig. 15. Results for noise in normals with our method.

simulate the case where the input data were captured from two viewpoints rotationally differentiated by 18.00 degrees around the axis $(0.01, 0.99, -0.03)$. We used the same sets of parameters for all synthetic experiments: *max_it* = 10; *conv_thresh* = 0.02 - radian; for ICPA, *percentage* = 30%; for the proposed method, *thresh* = 0.05 s, *max_size* = 0.1 mm and *tol_thres* = 0.1.

To verify the effects against estimated illumination, we randomly rotated the direction vector of exact illumination. That is, let the angle between the perturbed direction vector and the ground truth direction vector of illumination be (θ, ϕ), in which ϕ is a uniform random number from 0 degree to 360 degrees. We evaluated our method with different values of θ, and therefore different estimates of illumination. The value of θ was changed from 0 to 13 degrees by 0.7 degrees. For all values of θ, we applied our method 30 times under the same initial conditions.

Fig. 13 plots the quantitative evaluation of registration results in terms of averages and variances in error in results obtained with different estimates of illumination. The results obtained with ICPA have also been shown for comparison. Since ICP-CG failed in registration because of geometrical symmetries in the shape of the object, we did not carry out comparative experiments with ICP-CG. As expected, we find that our method is in average more

(a) Intensity image. (b) Albedo image. (c) Our method. (d) ICPA.

(e) Intensity image. (f) Albedo image. (g) Our method. (h) ICPA.

(i) Intensity image. (j) Albedo image. (k) Our method. (l) ICPA.

Fig. 16. Examples of registration results obtained with different types of noise. From top to bottom: results with 17 degrees of noise added to illumination, those with 10% of noise added to intensities, and results obtained with 6 degrees of noise added to normals.

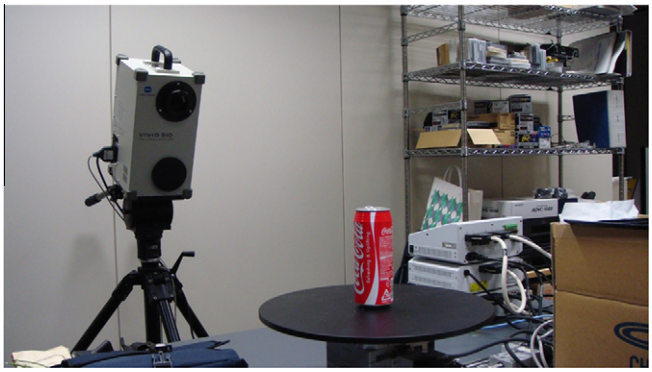


Fig. 17. Experimental set up to capture range images.

accurate and more stable than ICPA in estimating transformation on average. The results obtained with ICPA became inaccurate and unstable as soon as the estimated illumination differed slightly from being exact (for θ greater than 4.2 degrees). The results with ICPA were largely unsatisfactory when the estimates of illumination were not close to being exact. In contrast, our proposed approach gained satisfactory results for θ up to 12 degrees. Note that, in addition to small errors, the variance was also small, which proves the robustness and reliability of our proposed approach against noise in illumination. Therefore, we can conclude that it is more robust to changes in illumination than ICPA. Moreover, for exact estimates of illumination, our proposed method achieves registration that is as accurate as that with ICPA.

Fig. 14 plots quantitative results of registration obtained with our proposed method under various noise in intensities where the ground truth illumination was used. We applied Gaussian noise with a variance σ of several percent to the average of the image intensities. The value σ was changed from 0% to 10% by 0.5%. For each different noise intensity, we applied our method 30 times under the same initial conditions. We observe that even with noise with a variance of 9.5%, the largest errors are under the resolution of the range sensor.

We also conducted intensive experiments under noise added to normals (Fig. 15). We randomly perturbed each normal vectors. That is, let the angle between the direction of a perturbed normal and a ground truth normal be (α, ϕ), in which ϕ is a uniform random number from 0 degree to 360 degrees. We evaluated our method with different values of α, and therefore different perturbations in the normals. The value α was changed from 0 to 10 degrees by 0.6 degrees. For each different intensity of noise, we applied our method 30 times under the same initial conditions. We observe that even with noise of angle 10 degrees, the largest errors are of the same order as the resolution of the range sensor. We find that our method is stable against both geometric and photometric noise from these results.

(a) First image. (b) Second image. (c) Superimposed.

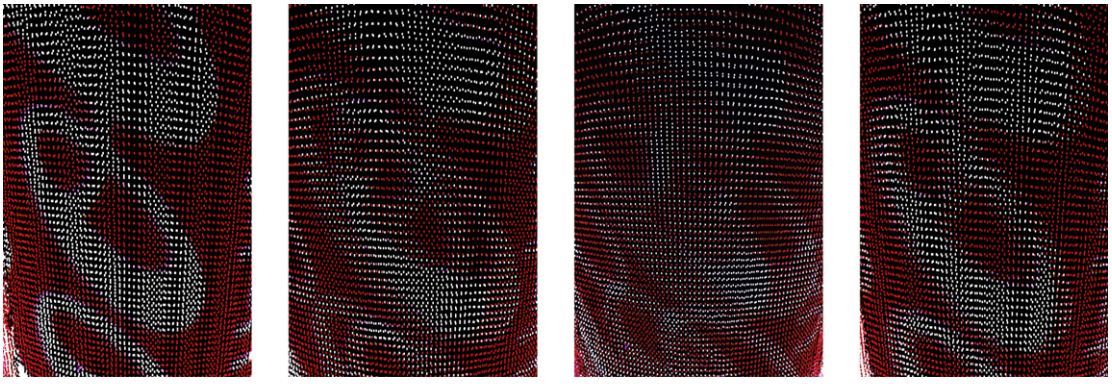
Fig. 18. Range images captured from different viewpoints.

(a) Albedo first image. (b) Albedo second image. (c) Gradient image. (d) Speed image.

Fig. 19. Estimate of albedo, gradient, and speed images.

(a) Ground Truth. (b) Our method. (c) ICPA. (d) ICP-CG. (e) SURF.

Fig. 20. Results obtained with data *can* using different methods.



(a) Our method. (b) ICPA. (c) ICP-CG. (d) SURF.

Fig. 21. Zoom in on the square part in Fig. 20.

Table 2
Description of data *can* used for experiment.

Nb_Points	Resolution	Expected_rot (angle; axis)	Expected_trans
35,000	0.55 mm	(17.83; 0.00, −0.94, −0.34)	(−8.05, 0.41, −1.47)

Table 4
Description of data *can2*.

Nb_Points	Resolution	Expected_rot (angle; axis)	Expected_trans
94,000	0.3 mm	(17.52; 0.00, −0.94, −0.34)	(−9.16, 0.55, −1.38)

Fig. 16 has some particular examples of results obtained under different types of noise with our proposed method and with ICPA. The first example corresponds to noise of 17 degrees added to illumination, where we obtained error of 0.49 with our method and error of 24.39 with ICPA. The second example corresponds to noise of 10% added to intensity, where we obtained error of 0.62 with our method and error of 2.43 with ICPA. The last example corresponds to noise of 6 degrees added to normals, and we obtained error of 0.62 with our method and error of 5.30 with ICPA. The quantitative results confirm the improvements in accuracy with our proposed method than with ICPA.

5.3. Evaluation with real data

We conducted experiments using real objects to test the effectiveness of the proposed method. In addition to ICPA and ICP-CG, we also compared the proposed method with SURF. SURF is a fast and robust well-known detector and descriptor that is often used for matching points between overlapping images. SURF uses only 2D gray images and is most effective for images with large amount of distinctive patterns. Using approximations of Hessian matrices allows fast detection of distinctive points at multiple scales. Note that we used the

Table 3
Quantitative evaluation of registration, using data *can*.

	NbMPts	Final_rot (angle; axis)	Final_trans
Proposed method	2400	(17.83; −0.01, −0.94, −0.34)	(−8.00, 0.38, −1.52)
ICPA	25,000	(8.47; −0.02, −0.96, −0.28)	(−5.11, −1.12, −0.95)
ICP-CG	24,000	(4.01; −0.06, −0.98, −0.21)	(−2.92, −1.01, −0.57)
SURF	222	(15.33; 0.02, −0.94, −0.35)	(−6.65, 0.16, −1.04)

	Proposed method	ICPA	ICP-CG	SURF
Error	0.22	10.66	16.53	3.91
Time	3.0 mn	5.2 mn	5.4 mn	0.0 mn

source code for SURF without modifications that was provided at http://www.vision.ee.ethz.ch/~surf/.

We employed a Konica Minolta Vivid 910 range scanner, which captures the 3D shapes and textures of objects. A mechanical system was used to rotate objects (see Fig. 17). The data *can* obtained using this equipment are presented in Fig. 18. Because the position and orientation of the range scanner are unknown, it is difficult to obtain the ground truth from the experimental setup. Accordingly, we manually computed the ground truth transformation for all data to evaluate the registration results. That is, we chose about 10 corresponding points in two range images and computed the transformation that minimized the distance between all corresponding points. We employed the ground truth obtained in this way to evaluate errors using Eq. (10).

The proposed method starts with an estimate of the geometric transformation and with a rough estimate of global illumination conditions to estimate the albedo. The initial estimate of registration is obtained by just superimposing the two captured range images. Global illumination is manually estimated roughly by rotating the direction vector of illumination and changing its brightness, which corresponds to spot light. We used the same sets of parameters for all experiments: $max\_it = 10$; $conv\_thresh = 0.002$ radian; for ICP-CG and ICPA, $percentage = 30\%$; for the proposed method, $thresh = 0.2$ s, $max\_size = 5.5$ mm and $tol\_thres = 0.5$. This means that we do not need extra tuning for the parameters depending on objects.

We obtained two range images of a rotationally symmetric can that has a height of about 10cm and a diameter of about 5 cm (Fig. 18). Fig. 19 has the gradient image and the speed image computed from the estimate of albedo. We compared the results obtained with ICPA, ICP-CG, and SURF (Figs. 20 and 21) to demonstrate how effective our method is. Note that the same initial estimate was used for the four methods. The details on data are listed in Table 2 and the quantitative results are listed in Table 3. *Nb_Points* is the number of points in the range image in these tables, *Res* is the resolution of the range image, *NbMPts* represents the number of matched points at the end of the registration, *Final_rot* is the estimated rotation after registration, *Final_trans* is the estimated translation after registration, *Error* is the error between the estimated transformation and the ground truth transformation, and *Time* is the processing time of the registration. Note that for our method, ICPA and ICP-CG, we had 10 iterations to converge, and we observe that the improvement in the estimations of transformations became small after 3 iterations.

Because of the lack of geometric features, ICP-CG failed in registering the two range images. In addition, the registration result obtained with ICPA is also not satisfactory. The result obtained with SURF is more accurate than those with ICPA and ICP-CG because of the robustness and discriminative power of the SURF descriptor. However, we observed an error of about four times the resolution of the range images. This comes from the distorted texture patterns due to 2D projection. In contrast, we can see significant improvements in the registration obtained with the proposed method. It is remarkable that 15% of matched pairs were eliminated as incorrect matches in our method and the accuracy obtained was about 0.2 times the resolution of the range images. The error obtained is below the accuracy of the range sensor; it is not possible to obtain better accuracy to register these data. When the search area (see Section 3.3) was not dynamically restricted by $\Omega(\cdot)$, the results obtained were less accurate for all ICP variants. Indeed, the errors

(a) Image 1. (b) Image 2. (c) Superimposed. (d) Albedo 1. (e) Albedo 2.

Fig. 22. Range images captured from different viewpoints with zoom in, and albedo estimations.

(a) Ground Truth. (b) Our method. (c) ICPA. (d) ICP-CG. (e) SURF.

Fig. 23. Results obtained with data *can2* using different methods.

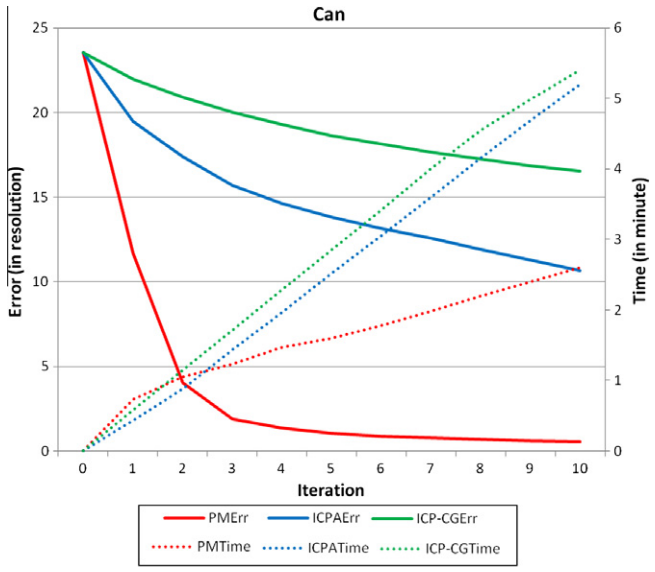


Fig. 24. Error versus computational time for each iteration during registration using data *can* (PM stands for proposed method).

here were 6.74, 12.83, and 19.96 times the resolution of the range images, with the proposed method, ICPA and ICP-CG respectively.

We captured two range images of the same object under the same situation using a lens with a different focal length to check the effect of the resolution of range images in our proposed method. The new lens zoomed in on the object surface, which increased the resolution of the range images. Description of the new data *can2* is listed in Table 4 and Fig. 4. Figs. 22 and 23 has qualitative evaluation of registration and Table 5 lists quantitative evaluation of registration. We observe that the results obtained with our proposed method are almost the same even for range images with different resolutions. This can be understood by the fact that we use neighborhood relationship to grow regions, which is independent to the resolution. Remark that for these data, we changed the parameter *max_size* to 3.0 mm only to reduce the computational time.

Figs. 27–29 show the results obtained with the different objects called *hand*, *box*, and *candy*. Table 6 lists the description of data before registration. Note that *hand* has a height of about 20 cm and a width of about 10 cm, *box* has a height of about 10 cm and a width of about 20 cm, and *candy* has a height of about 10 cm and a width of about 15 cm. Figs. 25 and 26 show initial estimates of registration and global illumination. Figs. 27–29 compare the results obtained with our method, ICPA, ICP-CG, and SURF. The quantitative results for these experiments are listed in Table 7. Identified incorrect matches were 35% for *hand* data, 7% for *box* data and 10% for *candy* data. Note that for the data *box*, we slightly changed the tolerance for outliers. That is, we set *Tdist* to be 4 × the "resolution of the image" at the end of registration.

Table 8 has the average of computational time required for the different steps of registration for our proposed method, ICPA and ICP-CG. We used an Intel Core 2 Duo CPU 3 GHz and all real data for the experiments. We do not show the time required for the registration using SURF. This is because when using SURF the registration is almost instantaneous compared with the other methods. Remark that the computational time were long compared to standards for registration methods. This is because we aimed at estimating the transformation aligning two range images as accurately as possible without considering time consumption. In this sense, we enforced ten iterations for the registration while in general, with only three iterations the methods almost converged. If a compromise between accuracy and processing time is searched, relaxing the threshold value for convergence will reduce the processing time while the registration accuracy decreases (we expect that an acceptable compromise will be around three iterations). In addition, we remark that the processing time of our proposed method was shorter than that of ICPA and ICP-CG. This is because in our proposed method, points without discriminative albedo (points in a large uniform area for example) are not considered in the matching step. As a consequence, the number of corresponding points obtained after matching was less with our proposed method and thus the outlier elimination step required less time.

We also show in Fig. 24 the computational time as well as error attained at each iteration during registration for our proposed method, ICPA, and ICP-CG when using data *can* (we had similar results for the other data). We observe that the error drastically decreases after the first few iterations with our proposed method. A small error means that the estimated transformation is close to the exact one. The initial estimate of transformation is

Table 5
Quantitative evaluation of registration, using data *can2*.

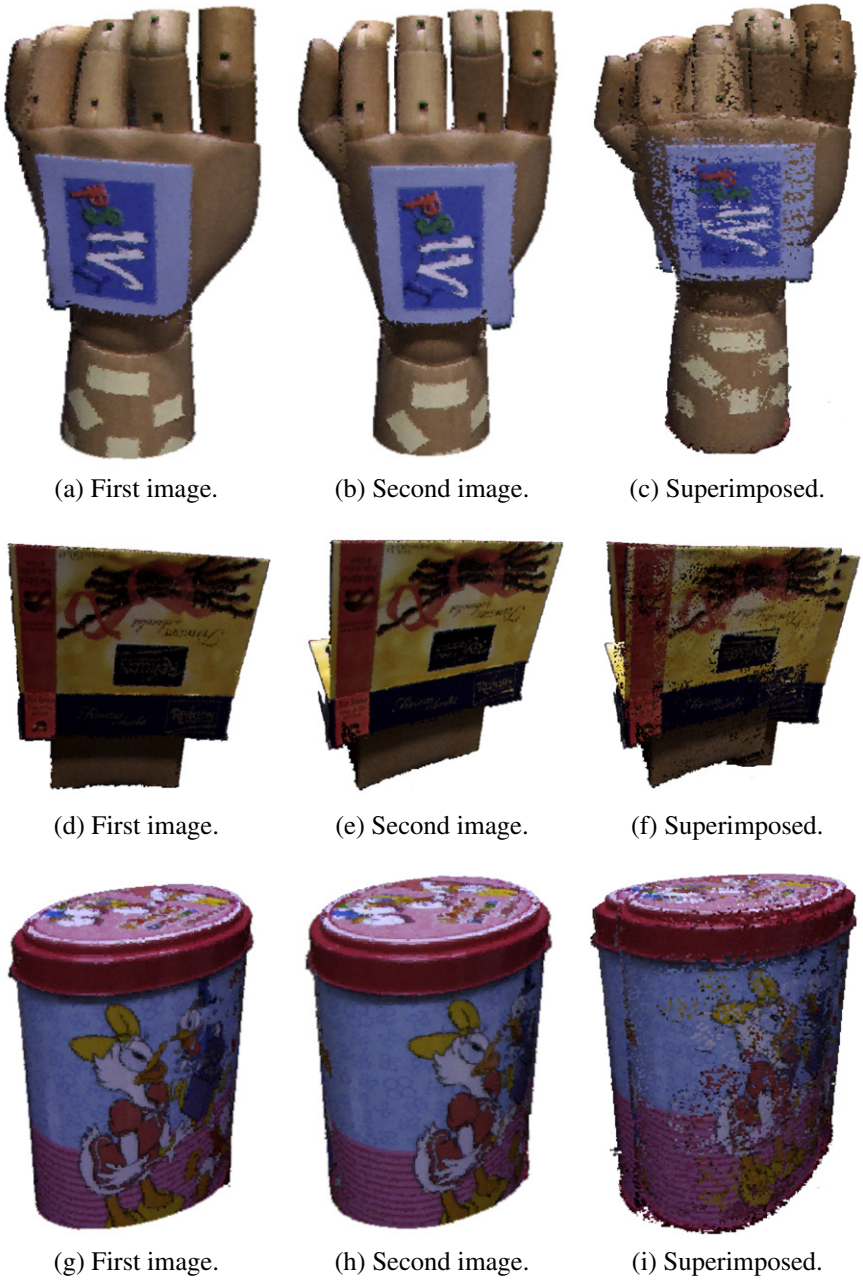
		NbMPts	Final_rot (angle; axis)	Final_trans
Proposed method		838	(17.46; −0.02, −0.94, −0.34)	(−9.06, −0.50, −1.44)
ICPA		44,000	(1.60; 0.35, 0.46, −0.82)	(−1.60, −1.36, −0.30)
ICP-CG		44,000	(6.69; −0.01, 0.96, 0.29)	(1.50, −0.15, −0.09)
SURF		330	(12.09; 05, −0.98, −0.19)	(−5.71, −0.53, −1.41)
	Proposed method	ICPA	ICP-CG	SURF
Error	0.42	40.60	55.28	35.6
Time	16.13 mn	31.0 mn	29.6 mn	0.0 mn

Table 6
Description of data *hand*, *box*, and *candy*.

	Nb_Points	Resolution (mm)	Expected_rot (angle; axis)	Expected_trans
hand	50,000	0.55	(17.66; −0.02, −0.94, −0.34)	(−7.32, 6.64, −0,80)
box	100,000	0.55	(17.62; −0.02, −0.94, −0.34)	(−4.96, −0.72, −2.06)
candy	50,000	0.55	(13.99; −0.02, −0.93, −0.36)	(−11.70, 1.58, 2.82)

far from the exact one and therefore the estimated transformation drastically changes during the first few iterations of registration with our proposed method. Consequently, the size of the search area (Ω), which decreases proportionally to the change in the estimated transformation, also drastically decreases during the first few iterations. Therefore, for each point of the two range images, the search for the corresponding point is restricted to a small area after the first few iterations, which decreases the computational time during the subsequent iterations. In contrast, the error decreases slowly with ICPA and ICP-CG during the iterations of registration and therefore the size of Ω also decreases slowly. Consequently, the computational time for each iteration of registration with ICPA and ICP-CG remains approximately constant. This explains the reason why the computational time for our proposed method decreases drastically with the first few iteration, while in contrast, that for ICPA and ICP-CG remains approximately constant. Finally, after ten iterations, the computational time of registration with our proposed method is far less than that with ICPA and ICP-CG.

Methods using only geometric features are not sufficient because of the lack of discriminative geometric features. Moreover, color information like chromaticity cannot be directly used because of changes in color due to illumination. In fact, neither ICPA nor ICP-CG attained satisfactory results due to data noise and inaccurate estimates of illumination. The SURF could not accurately register the range images, either. This is because of distortions due to 2D projection (e.g., as seen with data *candy*). In contrast, our method succeeded in registering range images with an accuracy of around the resolution of the range images for all data. Through all these experiments, our proposed method was the most accurate and stable in terms of the accuracy of estimated transformation.



(a) First image. (b) Second image. (c) Superimposed.

(d) First image. (e) Second image. (f) Superimposed.

(g) First image. (h) Second image. (i) Superimposed.

Fig. 25. Initial state for data *hand* (top), *box* (middle), and *candy* (bottom).

(a) Albedo first image. (b) Albedo second image. (c) Albedo first image. (d) Albedo second image.

(e) Albedo first image. (f) Albedo second image.

Fig. 26. Estimated albedo for data hand, box, and candy.

(a) Ground truth. (b) Our method. (c) ICPA. (d) ICP-CG. (e) SURF.

Fig. 27. Registration results obtained with data *hand*.

(a) Ground truth. (b) Our method. (c) ICPA. (d) ICP-CG. (e) SURF.

Fig. 28. Registration results obtained with data *box*.

(a) Ground truth. (b) Our method. (c) ICPA. (d) ICP-CG. (e) SURF.

Fig. 29. Registration results obtained with data *candy*.

Table 7
Quantitative evaluation of registrations, using data *hand*, *box*, and *candy*.

		NbMPts	Final_rot (angle; axis)	Final_trans
Hand				
Proposed method		2500	(17, 84; −0.01, −0.93, −0.36)	(−7.05, 6.66, −0.66)
ICPA		36,000	(17.68; −0.01, −0.93, −0.36)	(−6.90, 6.61, 0.60)
ICP-CG		37,000	(17.46; −0.01, −0.94, −0.34)	(−6.80, 6.64, −0.55)
SURF		189	(7.21; 0.26, −0.68, −0.68)	(−4.53, 6.55, 2.63)
	Proposed method	ICPA	ICP-CG	SURF
Error	0.82	1.14	1.20	16.02
Time	4.5 mn	9.8 mn	9.7 mn	0.0 mn
Box				
Proposed method		12,000	(17.75; −0.02, −0.94, −0.33)	(−4.56, −1.11, −2.21)
ICPA		57,000	(17.64; −0.03, −0.92, −0.35)	(−0.14, −1.62, −2.15)
ICP-CG		57,000	(17.67; −0.02, −0.94, −0.35)	(−1.51, −1.17, −1.98)
SURF		210	(17.39; −0.00, −0.95, −0.30)	(−4.25, −1.63, −2.70)
	Proposed method	ICPA	ICP-CG	SURF
Error	1.27	9.13	6.56	2.96
Time	12.0 mn	44.5 mn	41.1 mn	0.0 mn
Candy				
Proposed method		2000	(13.72; −0.01, −0.94, −0.35)	(−11.46, 1.43, 2.62)
ICPA		29,000	(9.94; −0.03, −0.92, −0.40)	(−10.52, 1.30, 2.54)
ICP-CG		30,000	(11.23; −0.01, −0.94, −0.34)	(−10.72, 1.18, 2.33)
SURF		210	(6.83; −0.28, 0.59, 0.75)	(−5.84, 2.45, 1.78)
	Proposed method	ICPA	ICP-CG	SURF
Error	0.81	5.14	4.03	23.25
Time	4.5 mn	7.7 mn	7.4 mn	0.0 mn

Table 8
Average of time consumption of each step of our proposed method, ICPA, and ICP-CG, with the percentage of time used for each step.

	Attribute definition	Matching	Outlier elimination
Proposed method	0.03 mn/0.2%	7.47 mn/94.8%	0.40 mn/5.0%
ICPA	0.00 mn/0.0%	14.12 mn/72.0%	5.49 mn/28.0%
ICP-CG	0.00 mn/0.0%	12.46 mn/72.0%	4.85 mn/28.0%

6. Conclusion

We introduced region-based registration of range images using reflectance attributes obtained under rough estimates of illumination conditions. Our method stably extracts reliable attributes that capture the local distribution of albedo on the object surface. These attributes are defined by adaptively growing regions that are generated using a level-set method. Such attributes are used to evaluate the similarity of points to robustly obtain correspondences in points even under rough estimates of illumination conditions. Our method also efficiently removes mismatches by using the rigidity constraint of surfaces, which enhances the robustness of the registration process. Our experiments using synthetic and real data demonstrated improvements in the robustness and the accuracy of registration results under rough estimates of illumination.

Acknowledgment

This work was in part supported by JST, CREST.

References

[1] T. Masuda, Log-polar height maps for multiple range image registration, Computer Vision and Image Understanding 113 (11) (2009) 1158–1169.
[2] J. Salvi, C. Matabosch, D. Fofi, J. Forest, A review of recent range image registration methods with accuracy evaluation, Image and Vision Computing 25 (5) (2007) 578–596.

[3] L. Cerman, A. Sugimoto, I. Shimizu, 3D shape registration with estimating illumination and photometric properties of a convex object, in: Proc. of CVWW'07, 2007, pp. 76–81.
[4] P.J. Besl, N.D. McKay, A method for registration of 3-D shapes, IEEE Transactions on Pattern Analysis and Machine Intelligence 14 (2) (1992) 239–256.
[5] Z. Zhang, Iterative point matching for registration of free form curves and surfaces, International Journal of Computer Vision 13 (1994) 119–152.
[6] V.-D. Nguyen, V. Nzomigni, C.V. Stewart, Fast and robust registration of 3D surfaces using low curvature patches, in: Proc. of 3DIM'99, 1999, pp. 201–208.
[7] B. Jian, B.C. Vemuri, A robust algorithm for point set registration using mixture of Gaussian, in: Proc. of ICCV'05, vol. 2, 2005, pp. 1246–1251.
[8] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (4) (2002) 509–522.
[9] A.E. Johnson, M. Hebert, Surface registration by matching oriented points, in: Proc. of 3DIM'97, 1997, pp. 121–128.
[10] Z. Xie, S. Xu, X. Li, A high-accuracy method for fine registration of overlapping point of clouds, Image and Vision Computing 28 (4) (2010) 563–570.
[11] Y. Liu, Automatic range image registration in the markov chain, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (1) (2010) 12–29.
[12] G. Godin, D. Laurendeau, R. Bergevin, A method for the registration of attributed range images, in: Proc. of 3DIM'01, 2001, pp. 179–186.
[13] S. Weik, Registration of 3-D partial surface models using luminance and depth information, in: Proc. of 3DIM'97, 1997, pp. 93–101.
[14] A.E. Johnson, S.B. Kang, Registration and integration of textured 3D data, Image and vision computing 17 (2) (1999) 135–147.
[15] I.S. Okatani, A. Sugimoto, Registration of range images that preserves local surface structures and color, in: Proc. of 3DPVT'04, 2004, pp. 786–796.
[16] N. Brusco, M. Andreetto, A. Giorgi, G.M. Cortelazzo., 3D registration by textured spin-images, in: Proc. of 3DIM'05, 2005, pp. 262–269.
[17] K. Pulli, S. Piiroinen, T. Duchamp, W. Stuetzle, Projective surface matching of colored 3D scans, in: Proc. of 3DIM'05, 2005, pp. 531–538.
[18] D.G. Lowe, Object recognition from local scale-invariant features, in: Proc. of ICCV'99, vol. 2, 1999, pp. 1150–1157.
[19] H. Bay, T. Tuytelaars, L.V. Gool, Surf: speeded up robust features, in: Proc. of ECCV'06, 2006, pp. 404–417.
[20] E. Tola, V. Lepetit, P. Fua, Daisy: an efficient dense descriptor applied to wide baseline stereo, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (5) (2010) 815–830.
[21] L. Ibanez, W. Schroeder, L. Ng, J. Cates, The Insight Software Consortium, The ITK Software Guide, second ed., Kitware, Inc, 2005.
[22] Y. Liu, H. Zhou, X. Su, M. Ni, R.J. Lloyd., Transforming least squares to weighted least squares for accurate range image registration, in: Proc. of 3DPVT'06, 2006, pp. 232–239.
[23] J. Barron, D. Fleet, S. Beauchemin, Performance of optical flow techniques, International Journal of Visual Computing 12 (1) (1992) 43–77.