Colored Point Cloud Registration with Improved Hue-Assisted Normal Distributions Transform

Hyunki Hong and Beomhee Lee

***Abstract*—This paper describes an improved Hue-Assisted Normal Distributions Transform (HANDT), which enhances the accuracy of the registration of point clouds. In our previous work, HANDT was developed to improve the speed and accuracy of registration by utilizing the hue from the hue-saturation-value model. To improve the accuracy, the score function of HANDT is changed from the normal distribution function to the Mahalanobis distance function. In addition, the functions of the hue mean and variance are modified for the circular property of the hue. The performance of normal distributions transform (NDT), HANDT, and improved HANDT are evaluated by benchmark data sets. As a result, the translation and rotation errors of improved HANDT are lower than those of NDT and HANDT.**

***Index Terms*—Hue, color, point cloud, registration, normal distributions transform (NDT).**

I. Introduction

A map built with a range sensor generally provides the structure of the environment. It is not easy to recognize what objects are located on the map. However, if the map is colored, it is possible to recognize objects and textures. Usually, a range sensor and a vision sensor are mounted together on robots. Due to the reliable calibration of the sensors, points scanned by the range sensor are able to be paired accurately with colors scanned by the vision sensor. In addition, low-priced depth cameras such as Microsoft Kinect and Asus Xtion were launched for the past years. It stimulated researchers to study the colored point cloud registration.

The registration of images or scans is an essential process of 3D graph Simultaneous Localization and Mapping(SLAM). Due to the good extensibility of Iterative Closest Point (ICP) algorithm, most of color-supported registration algorithms are based on ICP. The early algorithm is Colored ICP (cICP) [1]. Y, I, and Q from the YIQ color model were integrated to the distance function. Similarly, 4D ICP integrated the hue variable from the hue-saturation-lightness(HSL)model to the distance function [2]. Color-constrained ICP uses six classes corresponding to six intervals of the hue range. Points would be classified to classes according to hue. The points in the highest-scored class would be registered by ICP [3]. The modified Colored.

Manuscript received July 23, 2014; revised November 11, 2014.

The authors are with the Seoul National University, Department of Electrical and Computer Engineering, Seoul, Republic of Korea (e-mail: ckowl@snu.ac.kr, bhlee@snu.ac.kr).

ICP quantizes I and Q to classify reference points. The corresponding cell of a new point on the IQ plane is decided by I and Q of the new point, and it searches for the closest point in the cell [4]. Color-supported generalized ICP is a color-supported variant of the generalized ICP (gICP), which is a variant of ICP [5]. L*, a*, b* from the L*a*b* color model are integrated to the distance function of gICP.

NDT is a competitive algorithm with ICP. NDT is known for the fast registration of large point clouds. It is also known to be robust against outliers and noise. Moreover, since hundreds of figures in a voxel are reduced to twelve figures, it is much more efficient in saving data. Hence, NDT is the most suitable algorithm for the registration of a large number of points. There is a color-supported variant called Color NDT. Color NDT is improved by applying color weights to the score function [6]. In spite of the increasing popularity of NDT, no more color-supported variants of NDT have appeared so far.

In our previous work, Hue-assisted NDT (HANDT) was proposed [7]. We utilized the hue from the hue-saturation-value (HSV) model because of its light-invariant and viewpoint-invariant properties. After the reference point cloud is divided by the octree structure, the points in each unit voxel are classified according to their hue values. In addition, the score function of NDT was modified to be weighted by huecoefficients.

We found that errors occurred because of the simple mean and variance of the hue variable. In this paper, the circular mean and variance functions of the hue variable are described. In addition, the form of the score function is changed from the normal distribution function to the Mahalanobis distance function.

II. Background

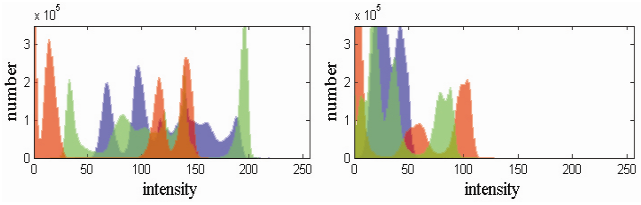
A. The Hue Variable from the HSV Model

We get color information by a vision sensor. The RGB values of an object are varying with the viewpoint of the sensor and the luminous intensity. On the contrary, the hue value from the HSV model is invariant to those effects. Two photographs are taken at different brightness levels as shown in Fig. 1. The distributions of RGB of photographs are as shown in Fig. 2. The distributions in Fig. 2(a) are widely distributed while those of Fig. 2(b) are concentrated at the left side. The hue distributions as shown in Fig. 3 are very close. Averages of four peaks in Fig. 3(a) are 32.58, 114.80, 155.34, and 246.88 while four averages in Fig. 3 (b) are 66.79, 127.43, 149.30, and 240.26. Due to the brightness-invariant property of the hue, the hue is possible to support the registration of two colored point clouds scanned at different brightness level.

DOI: 10.7763/IJCTE.2016.V8.1021

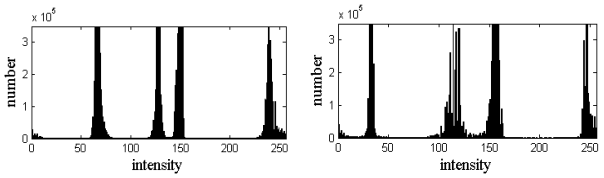
(a) (b)

Fig. 1. Colored papers taken at different brightness level.



(a) (b)

Fig. 2. RGB distributions of Fig. 1(a) and (b), respectively. red: R, green: G, blue: B.

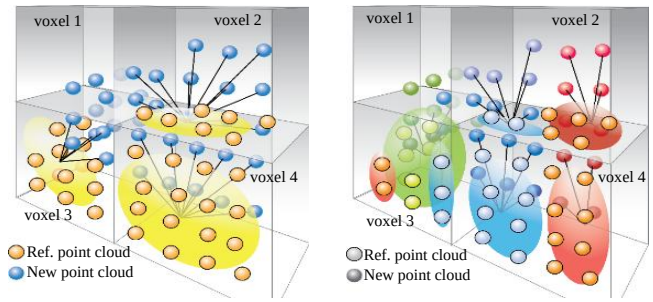


(a) (b)

Fig. 3. Hue distributions of Fig. 1(a) and (b), respectively.

B. NDT Algorithm

The NDT algorithm works at three steps which are dividing, computing, and registering [8]-[10]. At the first step, voxels with a pre-set size are set over reference point cloud C_{ref}. Orange balls in Fig. 4(a) are points in C_{ref}. Points in the ith voxel is a subdivided point cloud $C_{ref,i}$.



(a) (b)

Fig. 4. (a) NDT. Yellow ellipsoids are NDs. Black lines show the correspondences. (b) HANDT. Compare to (a), colors and brightness of balls are various. Two or more ellipsoids are in a voxel.

At the second step, it computes the mean $\mathbf{M}_i$ and covariance $\mathbf{\Sigma}_i$ of $C_{ref,i}$ by

$$\mathbf{M}_i = \frac{1}{N_i}\sum_{j=1}^{N_i}\mathbf{p}_{ref,ij} \tag{1}$$

$$\mathbf{\Sigma}_i = \frac{1}{N_i - 1}\sum_{j=1}^{N_i}\left(\mathbf{p}_{ref,ij} - \mathbf{M}_i\right)^2 \tag{2}$$

where $\mathbf{p}_{ref,ij} = (p_{x,j}\, p_{y,j}\, p_{z,j})$ is the position of the jth point in the cloud $C_{ref,i}$, and N_i is the number of points in $C_{ref,i}$. $\mathbf{M}_i$ and $\mathbf{\Sigma}_i$ are used to compose the ith normal distribution (ND) function in the form of (3).

$$s = \exp\left(-\frac{1}{2}(\mathbf{p}'_{new} - \mathbf{M}_i)^T \mathbf{\Sigma}_i^{-1}(\mathbf{p}'_{new} - \mathbf{M}_i)\right) \tag{3}$$

$$\mathbf{p}'_{new} = \mathbf{R}\mathbf{p}_{new} + \mathbf{T} \tag{4}$$

$$\mathbf{T} = \begin{pmatrix} T_x & T_y & T_z \end{pmatrix}^T \tag{5}$$

$$\mathbf{R} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & cos\,\alpha & -sin\,\alpha \\ 0 & sin\,\alpha & cos\,\alpha \end{pmatrix}\begin{pmatrix} cos\,\beta & 0 & sin\,\beta \\ 0 & 1 & 0 \\ -sin\beta & 0 & cos\,\beta \end{pmatrix}\begin{pmatrix} cos\,\gamma & -sin\,\gamma & 0 \\ sin\,\gamma & cos\,\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{6}$$

where s is the score of $\mathbf{p}'_{new}$, which is a vector transformed from $\mathbf{p}_{new}$ by rotational matrix $\mathbf{R}$ and translational vector $\mathbf{T}$. $\mathbf{p}'_{new,j}$ is a vector transformed from the jth newly scanned point $\mathbf{p}_{new,j}$ by (4). s_j is computed by substituting $\mathbf{p}'_{new,j}$ to(3).

At the final step, a transformation vector $\boldsymbol{\xi} = [\alpha\ \beta\ \gamma\ T_x T_y T_z]^T$ is updatediteratively to get the best tor $\boldsymbol{\xi}^*$ which maximizes a score function as

$$S_{sp} = \sum_{j=1}^{N}\exp\left(-\frac{1}{2}(\mathbf{p}'_{new,j} - \mathbf{M}_j)^T \mathbf{\Sigma}_j^{-1}(\mathbf{p}'_{new,j} - \mathbf{M}_j)\right) \tag{7}$$

where S_{sp} is a spatial score, N is the number of points in C_{new}, $\mathbf{M}_j$ and $\mathbf{\Sigma}_j$ are the corresponding mean and covariance of $\mathbf{p}'_{new,j}$, respectively. The corresponding ND of $\mathbf{p}'_{\text{new},j}$ is the voxel where $\mathbf{p}'_{\text{new},j}$ is. Blue balls in Fig. 4(a) are C_{new}. Each ball is connected to an ellipsoid in the voxel where the ball is. To maximize S_{sp}, it is common to use Newton's method or Leven berg-Marquardt Algorithm.

C. The Review of the HANDT Algorithm

As well known, HANDT is a color-supported variant of NDT utilizing the hue from HSV model. HANDT is in the same frame work of original NDT which are dividing, computing, and registering.

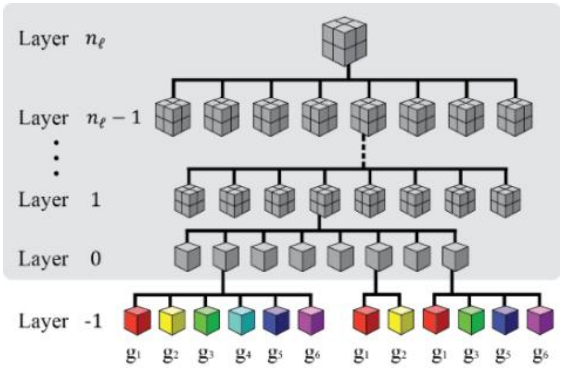


Fig. 5. The additional division of octree. The tree in the gray box is the original octree.

At the first step, the colored octree structure as showing Fig. 5 is applied. The tree in the gray box in Fig.5 is the original octree structure. The structure in the box is available to be replaced with other structures. The colored octree additionally divides the leaves of the octree into at most n_{hue} color leaveswhich are corresponding to hue intervals. For example as shown in Fig. 5, since n_{hue} is 6, six intervals of the hue range are$[0, 1/6)$, $[1/6, 2/6)$, $[2/6, 3/6)$, $[3/6, 4/6)$, $[4/6, 5/6)$, $[5/6, 1)$ .The kth reference point$\mathbf{p}_{ref,k}$ in the ith voxel is classified into the jth hue group $g_{hue,ij}$ if its hue is in the jth interval.

At the second step, we compute$\mathbf{M}_{ij}$ and $\mathbf{\Sigma}_{ij}$ for$g_{hue,ij}$. $\mathbf{M}_{ij}$ and $\mathbf{\Sigma}_{ij}$ are computed by (1) and (2).Next, we computes

extra mean μ_{ij} and variance σ_{ij} for hue as

$$\mu_{ij} = \frac{1}{N_{ij}} \sum_{k=1}^{N_{ij}} p_{ref,k,h} \tag{8}$$

$$\sigma_{ij} = \frac{1}{N_{ij} - 1} \sum_{k=1}^{N_{ij}} (p_{ref,k,h} - \mu_{ij})^2 \tag{9}$$

where N_{ij} is the number of points in $g_{hue,ij}$, and $p_{ref,k,h}$ is the hue of kth point $\mathbf{p}_{ref,k}$. μ_{ij} and σ_{ij} compose a normal distribution weight function as

$$w = \exp\left(\frac{-(p_{new,h} - \mu_{ij})^2}{2\sigma_{ij}} \right) \tag{10}$$

where $p_{new,h}$ is the hue of a new point $\mathbf{p}_{new}$. The weight w is computed by substituting $p_{new,h}$ to (10). There are at most n_{hue} normal distributions of the score and weight in a voxel. In contrary to the original NDT as shown in Fig. 4(a), there are multiple ellipsoids in a voxel in Fig. 4(b).

At the final step, HANDT registers C_{new} with C_{ref}. HANDT would search for the corresponding voxel of $\mathbf{p}_{new}$ in C_{new}. The corresponding voxel is the voxel where $\mathbf{p}_{new}$ is. Next, it would search for a corresponding hue group. The group would be decided by which interval its hue is in. The score s_k is computed by (3), and the weight w_k is computed by (10).

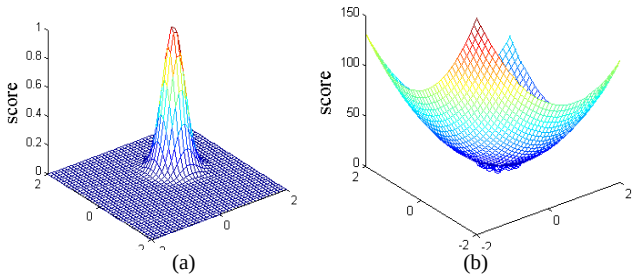
HAND Titeratively updates $\boldsymbol{\xi}$ by Newton's method to obtain the best vector $\boldsymbol{\xi}^*$ which maximizes a weighted score function as

$$S_{wsp} = \sum_{k=1}^{N} w_k s_k \tag{11}$$

where w_j is the weight, and s_j is the score of $\mathbf{p}_{new,j}$. $\boldsymbol{\xi}$ is iteratively updated as

$$\boldsymbol{\xi}_{k+1} = \boldsymbol{\xi}_k + \alpha \Delta \boldsymbol{\xi}_k \tag{12}$$

where $\Delta\boldsymbol{\xi}_k$ is a step vector, and α is a step length decided by Armijo's rule in kth iteration.



(a) (b)

Fig. 6. The score distributions of one hundred random points. The mean is (0.52, 0.45). (a) The normal distribution score function. (b) The Mahalanobis distance score function.

III. The Improved HANDT

The HANDT algorithm is improved by two modifications. First, the score function is changed from the normal distribution function to the Mahalanobis distance function. Second, the mean and variance of the hue are changed from simple functions to circular functions. After those modifications are described, the details of improved HANDT algorithm would be described.

A. Mahalanobis Distance

The example of the two dimensional normal distribution in a grid is as shown in Fig. 6(a). Since the scores outside the certain range are approximately zero, NDT or HANDT is not able to attract the exact correspondence which are out of that range. How to attract those corresponding points as well as avoid the effect of noise and outliers to obtain the better step vector is the problem. To score all points in the voxel, it is appropriate to replace the normal distribution score function by Mahalanobis distance function.

The Mahalanobis distance function is

$$D_m(\mathbf{p}) = \sqrt{(\mathbf{p} - \mathbf{M})^T \boldsymbol{\Sigma}^{-1} (\mathbf{p} - \mathbf{M})} \tag{13}$$

where $\mathbf{p}$ is the coordinate vector. The score function is changed to be in the form of the square of the Mahalanobis distance as

$$s_{md} = D_m{}^2(\mathbf{p}'_{ref}) = \left(\mathbf{p}'_{ref} - \mathbf{M}\right) \boldsymbol{\Sigma}^{-1} \left(\mathbf{p}'_{ref} - \mathbf{M}\right) \tag{14}$$

B. The Circular Mean and Variance of the Hue

Since the hue value is circular, the simple mean and variance would lead to a wrong distribution. Therefore, in the HANDT algorithm, the simple mean and variance of the hue are used. The circular mean μ_{cir} is computed by

$$\mu_{cir} = \frac{1}{2\pi} \left(\tan^{-1} \frac{\sum_{j=1}^{N} \sin(2\pi p_{ref,j,h})}{\sum_{j=1}^{N} \cos(2\pi p_{ref,j,h})} + k\pi \right) \tag{15}$$

where $k = 1$ if the sum of cosine is smaller than 0, $k = 2$ if the sum of sine is smaller than 0 and the sum of cosine is bigger than 0, and $k = 0$ for others.

For the circular property of the hue, the difference between two hue values should be in the range as $[-0.5, 0.5]$. The circular difference is formulated as

$$d_h(h_1, h_2) = \min\left(|h_1 - h_2|, 1 - |h_1 - h_2| \right) \tag{16}$$

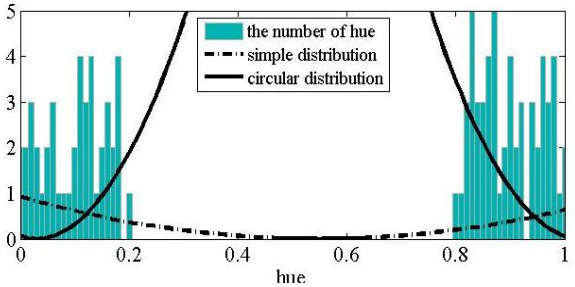


Fig. 7. Comparison between the distribution of the simple mean and variance and the distribution of the circular mean and variance.

Thus, the circular variance σ_{cir} is computed by

$$\sigma_{cir} = \frac{1}{N - 1} \sum_{j=1}^{N} d_h(p_{ref,j,h}, \mu)^2 \tag{17}$$

The circular weight w_{cir} is computed by

$$w_{cir} = \exp\left(\frac{-d_h{}^2\left(p_{ref,h}, \mu_{cir}\right)}{2\sigma_{cir}}\right) \tag{18}$$

The circular distribution and simple distribution of the hue are compared as Fig. 7. The circular mean is almost at the center of hue values while the simple mean is at the center of the hue range which is the wrong mean.

C. The Improved HANDT Algorithm

The process of the improved HANDT is as shown in Table I. The main difference between HANDT and improved HANDT is the function of the weight and the score.

TABLE I: HUE-ASSISTED NORMAL DISTRIBUTIONS TRANSFORM

Algorithm $\boldsymbol{\xi} = \textbf{HANDT}\left(C_{ref}, C_{new}, \ell, \mathrm{n}_\ell, n_{hue}\right)$

Inputs:
C_{ref} : reference point cloud, C_{new} : new point cloud, ℓ: the length of voxel, n_{hue}: the number of hue intervals, n_ℓ: the number of layers of Octree

Outputs:
$\boldsymbol{\xi}$:transformation vector

Dividing process:
1: $layer = n_\ell$
2: **repeat** recursively
3: **if** ($layer > 0$)
4: Divide the point clouds by sub-voxels
5: **else if** ($layer = 0$)
6: Divide point clouds into n_{hue} hue groups
7: **end if**
8: $layer = layer - 1$
9: **while** ($layer > -1$)

Calculating process:
1:**for** all voxels v_i
2: **for** all hue groups u_{ij}
2: **if** (number of points in u_{ij} >5)
3: Calculate $\mathbf{m}_{ij}, \boldsymbol{\Sigma}_{ij}, \mu_{ij}, \sigma_{ij}$
4: **end if**
5: **end for**
6: **end for**

Registering process:
1: while (step length > threshold)
2: Compute score s_w, $\mathbf{g}$, $\mathbf{H}$ of $C_{new,k}$
3: Compute the step vector $\Delta\boldsymbol{\xi}_k$ and length α
4: Update $\boldsymbol{\xi}_k$
4: Transform C_{new} by $\boldsymbol{\xi}_k$ and get $C_{new,k+1}$
5: end while
6: return $\boldsymbol{\xi}_k$

The weighted score function of the improved HANDT is modified as

$$S_{imp} = \sum_{i=1}^{n_{new}} w_{cir,i} s_{md,i} \tag{19}$$

where $w_{cir,i}$ is the weight of $\mathbf{p}_i$, and $s_{md,i}$ is the spatial score of $\mathbf{p}_i$. Since the score function is the square of the Mahalanobis distance, the optimization problem is changed to finding the transformation vector $\boldsymbol{\xi}^*$ which minimizes (19), and $\boldsymbol{\xi}^*$ is computed iteratively by (12). $\Delta\boldsymbol{\xi}$ in (12) is computed by

$$\Delta\boldsymbol{\xi}_k = -\frac{\mathbf{H}_k^{-1}\mathbf{g}_k}{\left\|\mathbf{H}_k^{-1}\mathbf{g}_k\right\|} \tag{20}$$

where $\mathbf{g}_k$ and $\mathbf{H}_k$ are the first and second order partial derivatives of (21) which are substituted with $\boldsymbol{\xi}_k$. Those $\mathbf{g}_k$ and $\mathbf{H}_k$ are computed as

$$\mathbf{g}_k = \sum_{i=1}^{n_{new}} w_{cir,i}\left(\mathbf{p}'_{new,i} - \mathbf{m}_i\right)^T \boldsymbol{\Sigma}_i^{-1} \frac{\partial \mathbf{p}'_{new,i}}{\partial \boldsymbol{\xi}}\bigg|_{\boldsymbol{\xi}=\boldsymbol{\xi}_k} \tag{21}$$

$$\mathbf{H}_{k,uv} = \sum_{i=1}^{n_{new}} w_{cir,i}\left(\mathbf{p}'_{new,i} - \mathbf{m}_i\right)^T \boldsymbol{\Sigma}_i^{-1} \frac{\partial^2 \mathbf{p}'_{new,i}}{\partial \boldsymbol{\xi}_u \partial \boldsymbol{\xi}_v} + \sum_{i=1}^{n_{new}} w_{cir,i} \frac{\partial \mathbf{p}'_{new,i}}{\partial \boldsymbol{\xi}_v}^T \boldsymbol{\Sigma}_i^{-1} \frac{\partial \mathbf{p}'_{new,i}}{\partial \boldsymbol{\xi}_u} \tag{22}$$

where $\mathbf{H}_{k,uv}$ is the (u, v) entry of $\mathbf{H}_k$.

IV. EXPERIMENTS

NDT, HANDT, and the improved HANDT are implemented by Point-Cloud-Library (PCL) [11]. We ran the implementation on Intel Core i7-3770 with PC1333 DDR3 dual-channel memory. Data sets we use are from [12]. Data sets were recorded by Microsoft Kinect, and the ground-truth trajectories were captured with 100Hz tracking cameras. 'freiburg1_room', 'freiburg2_desk', 'freiburg3_nostructure_texture_far', and 'freiburg3_structure_texture_far' are chosen.

First, the performance of the different n_{hue} of the improved HANDT are evaluated. Second, the performance of NDT, HANDT, and the improved HANDT are evaluated. The length of voxels is set to 5cm for the data set of the room and 10cm for other data sets. The time step is 0.27 second. To avoid singular covariance, the spatial score function of a hue group is available only if the number of points in the group is bigger than 5. The performance are compared in terms of the iteration, runtime, translation error, and rotation error. The convergence criterion is the step length. If the length is smaller than a threshold which is 10^{-6}, then the algorithms will stop.

A. Evaluation of Improved HANDT of Different n_{hue}

$n_{hue} = 1$ is chosen to evaluate to check how the average of hue supports NDT. Due to three primary colors, n_{hue} starts with 3. Next, n_{hue} is doubled up to 24. The results are as shown in Table II.

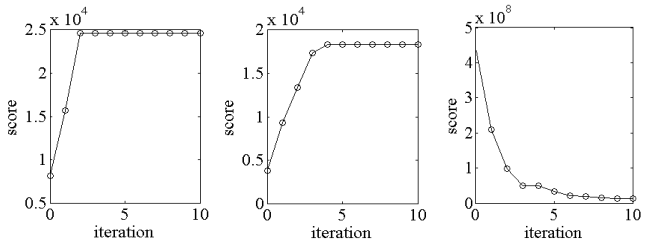
TABLE II: THE COMPARISON BETWEEN DIFFERENT N_{HUE}

Dataset	$\mathrm{n}_{\mathrm{hue}}$	Iterations	Runtime(s)	Error(m / °)
Freiburg1 room	1	4.965	19.695	0.119 / 5.130
	3	5.646	19.219	0.119 / 4.629
	6	5.439	14.918	0.119 / 5.192
	12	5.706	16.087	0.109 / 4.079
	24	6.127	16.344	0.116 / 4.492
Freiburg2 desk	1	5.181	35.196	0.077 / 1.637
	3	5.349	36.868	0.082 / 1.555
	6	5.222	29.842	0.079 / 1.496
	12	5.441	25.479	0.080 / 1.595
	24	5.846	32.859	0.081 / 1.501
Freiburg 3 Nostructure Texture(far)	1	4.965	39.515	0.077 / 0.721
	3	4.929	39.426	0.077 / 0.695
	6	4.393	35.900	0.076 / 0.621
	12	4.729	34.821	0.079 / 0.678
	24	4.346	33.595	0.082 / 0.619
Freiburg 3 Structure Texture(far)	1	4.548	36.431	0.058 / 0.901
	3	5.417	39.344	0.060 / 0.803
	6	4.833	36.092	0.060 / 0.824
	12	5.050	28.459	0.060 / 0.759
	24	5.717	40.636	0.059 / 0.711

As shown in Table II, the performance of the improved HANDT become better as n_{hue} increases until $n_{hue} = 12$. Since the hue intervals become narrower as n_{hue} increases, the number of points in each hue group becomes smaller. Due to the constraint against singular covariance, the number of available groups decreases. Hence, more newly scanned points are possible to be dropped because the corresponding groups are not available. Moreover, as n_{hue} increases, the hue intervals become narrower, and the hue of a newly scanned point and the hue mean of the corresponding group become closer. It means that the point clouds are possible to find more appropriate correspondences. Due to this reason, the errors become lower as n_{hue} increases. However, as n_{hue} is 24, the errors increase. The reason is that the algorithm drops too many newly scanned points because the hue intervals are too narrow to be available. As the result shown in Table II, the result of $n_{hue} = 12$ is generally the best.

B. Evaluation of NDT, HANDT, and Improved HANDT

In this experiment, n_{hue} of HANDT and the improved HANDT are set to 12. The results are as shown in Table III. Graphs in Fig. 8 show the convergence of algorithms. The point cloud sat 9116.45 and 9116.68 second in 'freiburg3_structure_texture' data set are used. The data is also used to show the score function against the step length as shown in Fig. 9.



(a) (b) (c)

Fig. 8. The convergence of algorithms. (a) NDT (b) HANDT (c) Improved HANDT.

As the results shown in Table III, although the runtime per iteration of the improved HANDT is shorter, the runtime is longer than NDT and HANDT. However, the translation and rotation errors of the improved HANDT are lower than those of NDT and HANDT.

TABLE III: THE COMPARISON BETWEEN ALGORITHMS 'IMHANDT' IS THE IMPROVED HANDT

Dataset	Algorithm	Iterations	Runtime (s)	Runtime per iteration	Error (m / °)
Freiburg1 room	NDT	3.915	16.421	4.194	0.178 / 4.394
	HANDT	3.669	12.367	3.371	0.175 / 4.024
	imHANDT	5.706	16.087	2.819	0.109 / 4.079
Freiburg2 desk	NDT	2.876	22.774	7.919	0.075 / 1.231
	HANDT	3.306	22.519	6.812	0.067 / 1.174
	imHANDT	5.050	28.460	5.636	0.060 / 0.759
Freiburg 3 Nostructure Texture(far)	NDT	3.610	31.795	8.807	0.271 / 1.191
	HANDT	5.288	37.714	7.132	0.205 / 1.111
	imHANDT	4.729	34.821	7.363	0.079 / 0.678
Freiburg 3 Structure Texture(far)	NDT	3.620	24.392	6.738	0.135 / 1.912
	HANDT	4.686	25.802	5.506	0.123 / 1.886
	imHANDT	5.441	25.479	4.683	0.080 / 1.595

Fig. 8(a) shows the convergence of NDT. It shows that NDT would stop after the second iteration. Fig. 9(a) and Fig. 9(b) show the score functions of NDT against the step length at the first and second iterations. The step length obtained at second iteration is approximately zero. If the step length is shorter than threshold, we regard that the registration is converged.

As shown in Table III, the translation and rotation errors of HANDT and the improved HANDT are lower than NDT. The reason is that the reference point cloud is more geometrically well-represented by distributions of hue groups. In addition, the algorithms find better correspondences than NDT. However, the runtime of HANDT and the improved HANDT are longer than NDT. Fig. 8(b) and Fig. 8(c) show HANDT and the improved HANDT need more iterations until stop than NDT.

The improved HANDT performs the more accurate registration than HANDT due to the Mahalanobis distance score function. For a point far away from the mean of the voxel, whose eigenvalues of the covariance matrix are small, the score of the point computed by improved.

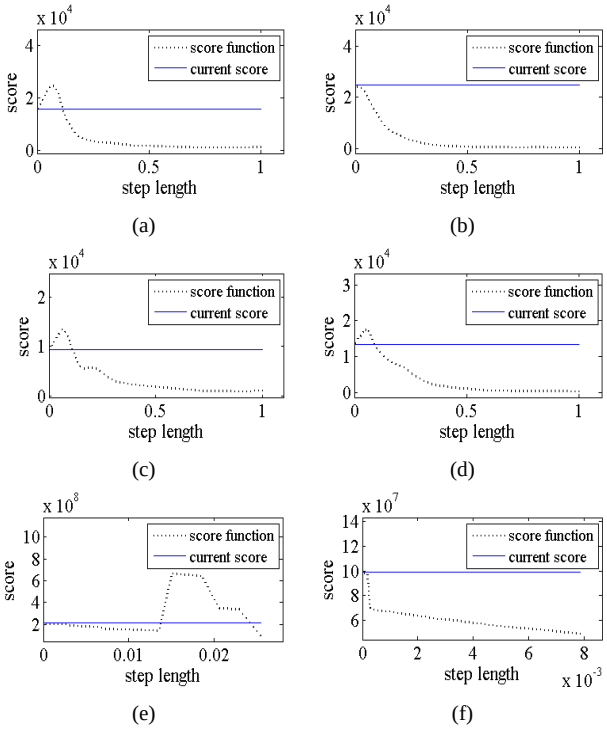


Fig. 9. Score functions. (a) and(b) are the score functions of NDTplotted against the step length. (c) and (d) are those of HANDT. (e) and (f) are those of the improved HANDT. (a), (c), (e) are the score functions of the 1st iterations, and (b), (d), (f) are those of the 2nd iterations.

HANDT is significant while the score computed by HANDT is approximately zero. It means that all of the newly scanned points in the voxel participate in scores, first and second order partial derivatives. Due to this fact, the improved HANDT registers all of the newly scanned points to reference point cloud.

The initial guess of the step length of NDT or HANDT is usually one. As the graphs shown in Fig. 9(a) and (b), which are the score function against the step length at the first and second iterations of NDT, the score increases as the step length decreases, so does HANDT as shown in Fig. 9(c) and (d). On the other hand, the initial guess of the improved HANDT is better not to be one because the step length obtained by Armijo's rule is usually very big as shown in Fig. 9(e), and it would lead to the failure of registration. In

this experiment, the initial guess of the step length is chosen as $\|\mathbf{H}_k^{-1}\mathbf{g}_k\|$, which is much smaller than one. Since the initial guess usually satisfies Armijo's rule, the runtime per iteration of the improved HANDT is shorter than HANDT and NDT.

V. Conclution

The performance of improved HANDT is evaluated by benchmark data sets. As results, the accuracy of the registration is much better than the cases of NDT and HANDT. The improvement is approached by changing the weighted score function to Mahalanobis distance function and the modification of the hue mean and hue variance. Improved HANDT is expected to overcome the difficulty of the registration at flat or repeated structure by the hue.

Acknowledgment

This work was supported in part by the National Research Foundation of Korea(NRF) grant funded by the Korea government (MSIP) (No. 2013R1A2A1A05005547), in part by the Brain Korea 21 Plus Project, in part by ASRI, in part by the Industrial Foundation Technology Development Program of MOTIE/KEIT [Development of CIRT (Collective Intelligence Robot Technologies)], and in part by Bio-Mimetic Robot Research Center funded by Defense Acquisition Program Administration [UD130070ID].

References

[1] A. E. Johnson, E. Andrew, and S. B. Kang, "Registration and integration of textured 3D data," *Image and vision computing*, vol. 17, no. 2, pp. 135-147, 1999.

[2] H. Men, B. Gabre, and K. Pochiraju, "Color point cloud registration with 4D ICP algorithm," in *Proc. IEEE International Conf. Robotics and Automation*, Shanghai, 2011, pp. 1511-1516.

[3] S. Druon, M. Aldon, and A. Crosnier, "Color constrained ICP for registration of large unstructured 3D color data sets," in *Proc. IEEE International Conf. on Information Acquisition*, Weihai, 2006, pp. 249-255.

[4] J. H. Joung, K. H. An, J. W. Kang, M. J. Chung, and W. Yu, "3D environment reconstruction using modified color ICP algorithm by fusion of a camera and a 3D laser range finder," in *Proc. IEEE/RSJ International Conf. Intelligent Robots and Systems*, St. Louis, 2009, pp. 3082-3088.

[5] M. Korn, M. Holzkothen, and J. Pauli, "Color supported generalized-ICP," in *Proc. International Conf. Computer Vision Theory and Applications*, Lisbon, 2014, pp. 592-599.

[6] B. Huhle, M. Magnusson, W. Straßer, and A. J. Lilienthal, "Registration of colored 3D point clouds with a kernel-based extension to the normal distributions transform," in *Proc. IEEE International Conf. Robotics and Automation*, Pasadena, 2008, pp. 4025-4030.

[7] H. K. Hong and B. H. Lee, "The improved 3D normal distributions transform algorithm using hue," in *Proc. the 8th Korea Robotics Society Annu. Conf.*, 2014.

[8] P. Biber and W. Straßer, "The normal distributions transform: a new approach to laser scan matching," in *Proc. IEEE International Conf. Robotics and Automation*, Las Vegas, 2003, pp. 2743-2748.

[9] E. Takeuchi and T. Tsubouchi, "A 3-D scan matching using improved 3-D normal distributions transform for mobile robotic mapping," in *Proc. IEEE International Conf. Robotics and Automation*, Beijing, 2006, pp. 3068-3073.

[10] U. Cihan and H. Temeltaş, "3D multi-layered normal distribution transform for fast and long range scan matching," *Journal of Intelligent and Robotic* Systems, pp. 85-108, 2013.

[11] R. Rusu and S. Cousins, "3D is here: point cloud library (pcl)," in *Proc. IEEE International Conf. Robotics and Automation*, Shanghai, 2011, pp. 1-4.

[12] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *Proc. IEEE/RSJ International Conf. Intelligent Robots and Systems*, Vilamoura, 2012, pp. 573-580.

Hyun K. Hong received his B.S. degree in electrical and computer engineering from Seoul National University in 2012. He is currently a M.S candidate in the Department of Electrical and Computer Engineering at Seoul National University. His current research interests include graph SLAM and multi-agent control.

Boem H. Lee received the B.S. and M.S. degrees in electronics engineering from Seoul National University, Seoul, the Republic of Korea in 1978 and 1980, respectively, and the Ph.D. degree in computer, information, and control engineering from the University of Michigan, Ann Arbor, MI in 1985.

He was an assistant professor with the School of Electrical Engineering at Purdue University, West Lafayette, IN from 1985 to 1987. He joined Seoul National University in 1987, and is currently a professor with the Department of Electrical and Computer Engineering. His research interests include multi-agent system coordination, control, and application.

Prof. Lee has been a fellow of the Robotics and Automation Society since 2004.