Appl. Math. Inf. Sci. **8**, No. 1, 263-271 (2014)

263

Contour-Based Image Registration using Bipartite Graph Matching with Munkres Algorithm

Qingsong Zhu[1,2,3,4], *Tiexiang Wen*[1,2,3], *Yaoqin Xie*[1,2,3,4], *Jia Gu*[1,2,3,*] *and Lei Wang*[1,2,3,*]

[1] Key Lab for Health Informatics, Chinese Academy of Sciences, Shenzhen 518055, China
[2] Shenzhen Key Lab for Low-Cost Healthcare, Chinese Academy of Sciences, Shenzhen 518055, China
[3] Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China
[4] School of Medicine, Stanford University, Stanford, California, USA

Abstract: Image registration is a vital research topic in image processing. In this paper, we present a novel method to register color image and depth information captured by Kinect for Xbox 360. A contour-based approach is employed to find the correspondence points and estimate the transformation parameters model between color image and depth image. In our paper, the boundary contours are detected by Canny edge detector and located to sub-pixel level by a B-spline model. The shape descriptors of contours are computed by shape context. The correspondence points on both similar contours will have similar shape descriptors, which enables us to solve a bipartite graph matching with Munkres algorithm. Based on the correspondence points, the transform model that best aligns the two images can be estimated via RANSAC method. Experiments on a series of natural images are used for compare with some existing approaches and demonstrate the effectiveness of our method.

Keywords: Image registration, bipartite graph matching, Munkres algorithm

1 Introduction

Image registration is a vital research topic in image processing and it has been widely used in computer vision [1], remote sensing [2] and medical image analysis [3]. In some computer vision applications, there is a need to integration complementary data from different sensors, from different viewpoints, or from different times. A general survey of existing methods on image registration can be found in [4,5].

According to Zitova and Flusser [5], image registration algorithms consist of four main procedures.

● **Feature extraction.** It is one of the most important procedures in the feature-based image registration. For instance, points [6], edges [7] and closed boundary regions [8,9] can be manually or preferably automatically extracted. Those features can be described by their point representatives, such as distinctive points, line endings and differential descriptor for curves [10].

● **Feature correspondence.** In this procedure, the feature correspondence between input images is established based on the feature detection. Various feature descriptors

(chain-code [11], length code method [12] etc.) and similarity measures (Bessel method [13], mutual information [14] etc.) can be applied to determinate the correspondence points.

● **Transform model.** After the removal of the false-matching pairs, a set of correct correspondence pairs remains. Based on the coordinates of the correspondence points between two images, the type and parameters of transformation function is calculated using the established feature correspondence.

● **Image transformation.** Based on the mapping function, one image can be transformed into another image. This type of transformation is used to properly overlay two images.

Recently, Microsoft has launched the Kinect for Xbox 360 [15] which has received a growing amount of attention in computer vision community [16,17,18]. Kinect can simultaneously capture RGB image and range data at the maximum rate of 30 frames per second from different viewpoints. However, those images are obtained at the cost of decreased quality due to blur and noise. In

* Corresponding author e-mail: jia.gu@siat.ac.cn, wang.lei@siat.ac.cn

order to process the color information and the depth data further, we plan to match them to address the misalignment problem by using image registration technique.

The rest of the paper is organized as follows. Section 2 reviews the existing feature-based image registration algorithms. In Section 3, we introduce our contour-based image registration algorithm. Experimental results on a series of natural images and comparison with other existing methods are given in Section 4. Finally, Section 5 gives the conclusions and notes for future work.

2 Related work

The current registration techniques can be broadly sorted into two main approaches: area-based and feature-based. Area-based techniques do not extract features because they mainly focus on the feature matching procedure. Feature-based methods rely on the detection of salient structures-features in the images. The feature-based methods are priority used when the images contain enough distinctive and easily extractable objects. Feature-based techniques include two crucial steps: feature detection and feature matching.

• **Feature Detection(FD)** As for feature detection, feature-based methods can be divided into three main groups: point-based, contour-based and region-based.

a) Point-based FD. In [29], the point features consist of the methods working with high variance points, inflection points of curves etc.

b) Region-based FD. In [9], the region features are detected by means of segmentation methods. In addition, Zhu etc. [20] presented a novel segmentation method by recursive Kernel Density Learning framework and used a Bayes classifier to eliminate the misclassification points to improve the segmentation quality.

c) Contour-based FD. In [14], A. Alvarez et al. proposed a contour-based image registration using mutual information. In [21], Li et al. introduced an elastic contour matching scheme based on the active contour model to perform multi-sensor image registration. In [22], Li and Leung presented a novel image registration approach by extracting open contours and closed contours from the images.

• **Feature Matching** The aim of feature matching is to find the correspondence pairs between two images using their various descriptor features or spatial relations. Belongie [23,24] introduced a shape descriptor named shape context to solve the correspondence problem as an optimal assignment problem. Chui [25] developed an RPM algorithm which can estimate the correspondence and non-rigid transformation between two sets of data that may not be of the same size. Tu et al. [26] defined a data-driven technique to give a hybrid algorithm for solving the correspondence problem. Van Kaick et al. [27] presented the first Ant Colony Optimization (ACO)
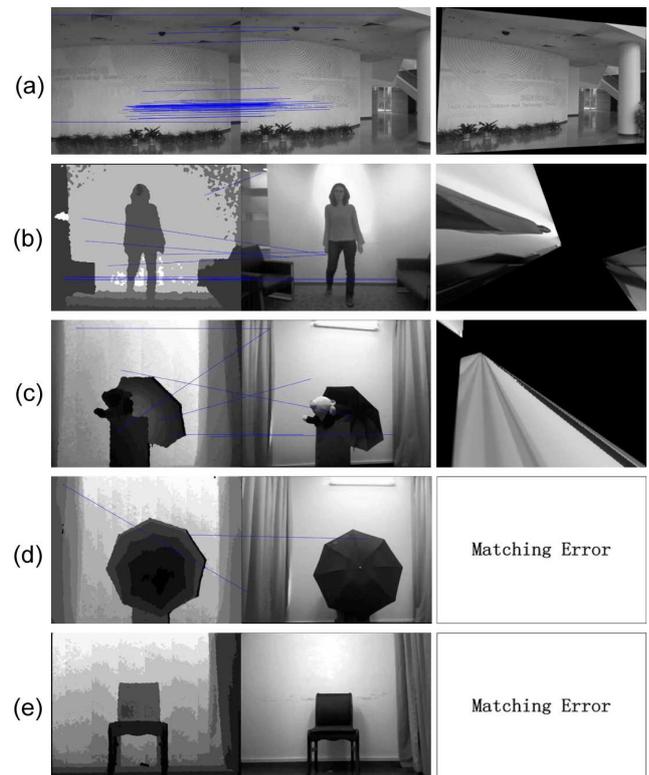

Fig. 1: Using SIFT feature for matching intensity image and depth image. The 3rd column shows the results of image transformation. (a) The matching of two intensity images works well. (b-e) The matching of depth image and intensity image is incorrect. (b) and (c) show the wrong matching points. (d) There are only two matching points and matching ends with error. (e) No matching points can be found)

algorithm specifically aimed at solving the QAP-based shape correspondence problem. Chitra et al. [28] used Euclidean distance and cost matrix with Hungarian algorithm to find correspondence.

All this work has been proved to be efficient in matching intensity images. One of the most successful methods was presented by D. Lowe [19,29]. The method can be used to perform reliable matching between different views of scene by extracting distinctive invariant features from images. Each local feature vectors is invariant to any scaling, rotation or translation of the image. SIFT features have been widely used in many applications, such as robot localization and mapping [30], panorama stitching [31] and 3D scene modeling recognition [32]. However, SIFT features do not perform very well for matching between intensity image and depth image. In our experiments the tests are conducted under the same conditions and the results are shown in Figure 1.

In this paper, we present a novel method to register color image and depth image captured by Kinect. We
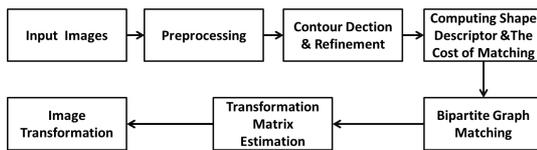
Fig. 2: Overview of our contour-based image registration method.

focus on the feature extraction and feature correspondence. Firstly, we coarsely segment the images to obtain the boundary contours. In order to increase the accuracy of matching, we select the similar contours from two images and refine them to sub-pixel level. Secondly, after obtaining the meaningful contours, we calculate the similarity matrix of contours. We introduce a bipartite graph matching method to detect the correspondence points between two contours. Finally, we can estimate the transform model based on correspondence pairs. In a series of experiments, the proposed contour-based image registration algorithm demonstrated good performance compared with other existing methods.

3 The Proposed Approach

This section describes the main steps of our method. The flow diagram of our method is shown in Figure 2. Implementation details are given as follows.

Given a color image and depth data, we first remove noise and smooth the input images for further processing. We explore the boundary information of the above images by using Canny edge detector. Then we manually select partial similar contours that may indicate the foreground object and refine the contours to sub-pixel by B-spline function. Next, we calculate the shape descriptor of contours by shape context and compute the similarity matrix which is called "cost of matching" based on Chi-square technique. We regard the matching of the above similar contours as bipartite graph matching, which can be solved by Munkres algorithm. Finally, we estimate the transformation matrix between two images by RANSAC method.

3.1 Preprocessing

In general, the images to be matched have different scale and contain noise, motion and blur. To prepare the data for our processing, it may be necessary to smooth the input images. However, smoothing may lead to noise reduction and blurring. For noise reduction a quantity of methods can be employed such as Gaussian, mean, or median filter etc. Due to the difference between color image and depth data that are taken by Kinect, we use different methods to process these two types of images.

As for color image, we use median filter [33] for noise reduction. The median filter is a nonlinear digital filtering method and it is better than Gaussian blur at removing noise whilst preserving edges. It involves detecting the median value in a local neighborhood. Let's denote the image before smoothing by F and the image after smoothing by $\bar{F}$. The image F contains $m \times n$ pixels and the filter is of radius r pixels.

$$\bar{F}(i,j) = MEDIAN(F,i,j,r), \qquad (1)$$

where $MEDIAN(F,i,j,r)$ is a function that returns the median intensity in image F in a circular window of radius r centered at (i,j). Circular windows are used to make smoothing independent of image orientation.

As for depth information, for all points where the sensor is not stable to measure depth are set to 0. We regard all the zero pixels as empty pixels which need to be filled. Thus we take the median pixel value from several frames to achieve meaningful depth data in all pixels and then we use median filter with 5×5 window on the depth value to make the image smooth.

3.2 Feature Extraction

Image features are unique image properties that can be used to estimate the difference and similarity between two images. Boundary contour is a valid mean of finding edge of object in an image, and it typically occurs on the edge between two different regions in an image. For edge detection variety of methodologies can be applied such as Canny operator [34], LoG operator [35] and Sobel operator [36]. Only the Canny edge detector allows to obtain connected extracted edges but it leads to wrong edges and corners when applied in a blind manner. The Canny operator uses a filter based on a single of Gaussian to look for local maximum gradients in the gradient direction and its parameters can be adjusted to identify the edges of differing features depending on the specific requirement of a given implementation. Thus we roughly segment the input images using Canny edge detection to obtain the boundary contours and select the similar contours from two images for further processing.

In order to increase the accuracy of feature detection, we intend to locate the boundary contour to sub-pixel level. Different sub-pixel edge detectors can be found in the literature such as moments [37,38], interpolation [39, 40] and circular edges [41]. However, in all these methods, the estimation is local and doesn't include a noise model. In this paper, we use a global approach for the estimation of sub-pixel edges based on B-spline model [42]. A B-spline model is a spline curve parameterized by spline function that has minimal support with respect to a given degree, smoothness and domain partition and it has been widely used in contour detection application [43,44]. Given a set $\{t_0, t_1, \cdots, t_{m-1}\}$ of m real values named knots, spline functions are polynomial

inside each interval $[t_{j-1}, t_j]$. The set $\{B_{j,n}(t), j = 0, \cdots, m-n-1\}$ of B-splines composes a basis for the linear space of all the splines. Therefore, a spline curve $f(t)$ of degree n is given as follows:

$$f(t) = \sum_{j=0}^{m-n-1} p_i B_{j,n}(t), t \in [t_n, t_{m-n-1}] \qquad (2)$$

where p_i are the weights employed to the respective basis functions $B_{j,n}$. The $m-n-1$ basis B-splines of degree n are defined by the Cox-deBoor recursion formulas as follows:

$$B_{j,0}(t) = \begin{cases} 1, t_j \leq t \leq t_{j+1} \\ 0 \end{cases} \quad j = 0, \cdots, m-2 \qquad (3)$$

$$B_{j,n}(t) = \frac{t-t_j}{t_{j+n}-t_j} B_{j,n-1}(t) + \frac{t_{j+n+1}-t}{t_{j+n+1}+t_{j+1}} B_{j+1,n-1}(t) \quad (4)$$

where $B_{j,n}(t) \geq 0$ and the partition of unity property: $\sum_i B_{j,n}(t) = 1$ for all t.

3.3 Feature Correspondence

The accuracy of matching depends on feature correspondence. To find correspondence means to measure the similarity between two input images. Statistical approach is applied for the feature matching of the images.

3.3.1 Shape Descriptor

In the image registration, shape refers to the type of boundary information. In the context of shape similarity, some shape descriptors have been introduced, such as Fourier descriptors, Hausdorff distance [45] and medial axis transform [46]. Rich descriptors reduce the ambiguity in matching. In other words, the matching is easier if one employs a rich descriptor. In our framework we use shape context [23,24] as shape descriptor. Shape context can be treated as a very robust point set registration technique. It can greatly simplify the matching part and it's invariant to scale, translation, rotation and deformation to some extent.

Consider that we extract the contour from each image, we need to find the best correspondence point between contours. Given a set $P = \{p_1, p_2, \cdots, p_m\}, p_i \in R^2$ of m points in a contour, for a point p_i of the contour, a coarse histogram g_i of the relative coordinates of the remaining $(n-1)$ points is computed [24]: $g_i(k) = \#\{q \neq p_i : (q - p_i) \in bin(k)\}$. This histogram $g_i(k)$ is referred to as the shape context of P. Also, we use a log-polar coordinate system to make the descriptor more sensitive to the positions of nearby sample points than to those of points farther away.

3.3.2 Cost of Matching

Consider a point p_i and another q_j from two contours on both images. Let us define $C_{ij} = C(p_i, q_j)$ as the cost of matching between the two points. The cost C_{ij} is computed by Chi-square (also χ^2-distribution) statistic [24].

$$C_{ij} = \frac{1}{2} \sum_{k=1}^{K} \frac{[g_i(k) - g_j(k)]^2}{g_i(k) + g_j(k)} \qquad (5)$$

where $g_i(k)$ and $g_j(k)$ indicate the K-bin (normalized) histogram at p_i and q_j, respectively.

3.3.3 Bipartite Graph Matching

Given the set of costs C_{ij} between all pairs of points p_i on the first contour and q_j on the second contour from two images, the total cost of matching should be minimized.

$$H(\pi) = \sum_i C(p_i, q_{\pi(i)}) \qquad (6)$$

where π is a permutation. Obviously, the correspondence between contours should be unique. This is a linear assignment problem instance. A variety of ways can be used to solve the bipartite graph matching problem, such as Ant Colony Optimization (ACO) algorithm [27], which is based on incorporating proximity information, COPAP method [47], Hungarian algorithm [48] and Munkres algorithm [49]. Mu-nkres algorithm is an efficient algorithm that can be used to solve the problem in polynomial-time. In practice, we use an improved Munkres algorithm [50]. The input to the assignment problem is a square cost matrix with entries C_{ij} and the minimized result of $H(\pi)$ can be achieved.

As the number of points in both contours may not be equal, we improved the algorithm so that it can identify a partial assignment, which indicates unassigned tasks. The procedure of the algorithm is given as follows:
1) Input the cost of matching matrix;
2) Subtract the smallest element from each row and find a zero in the resulting matrix;
3) Cover each column with a starred zero;
4) Save the smallest uncovered zero;
5) Construct a series of alternating primed and starred zeros;
6) Add the value found in step 4 to every element of each covered row;
7) Assignment pairs are manifested by the position of the starred zeros in the cost matrix.

3.4 Image Transformation

Given a set of correspondence points between two images, we can estimate a plane transformation to map points from one form to another. Considering a set of

observed data P which contains outliers and a set of model data Q, we want to establish the correspondence transformation matrix between P and Q. RANSAC [51] is an efficient method that finds the inliers and estimates parameters of a mathematical model between these two sets of data. RANSAC instantiates this model by using small, random subsets of the data until the model becomes consistent with a large subset of the data. The steps of the basic method are described as follows:

1) Randomly select the minimum number of data required to estimate the model parameters.

2) Solve the parameter values of the model.

3) Determine Determine the subsets of data that fit the predefined tolerance ε.

4) Re-estimate the model parameters until the number of inliers becomes less than the predefined threshold τ.

5) Otherwise, repeat steps 1 to 4 until reach the maximum of iterative times N.

4 Experimental Results

In order to validate the performance of the proposed technique, four cases are tested on several color images and depth information taken from Kinect device. We compare the matching results, RMSE and run time with existing methods like COPAP, ACO and Hungarian all these methods replace Munkres algorithm in the step of feature correspondence). We assume that $P = \{p_1 \cdots p_N\}$ and $Q = \{q_1 \cdots q_N\}$ are two sets of correspondence points. The RMSE is defined as:

$$RMSE = \sqrt{\sum_{i=1}^{N} \left[p_i(x,y) \cdot H - q_i(x,y) \right]^2 \Big/ N} \qquad (7)$$

where H is the estimated transformation matrix, N is the number of correspondence pairs. The four tests are performed in MATLAB R2012a and are performed on a PC with i32120 processor and 2 GB of RAM.

In the first case, we used a regular image (a woman). We manually extracted the similar contours by setting thresh = 0.8 and sigma = 1 of Canny operator for the depth image and thresh = 0.4, sigma = 1 for the color image. In the second case, we take an image with a bear and an umbrella. The parameters of Canny operator were set to thresh = 0.55, sigma = 0.9 for the color image and depth information. In the third case, we took an image with an umbrella. The parameters of Canny operator were set to thresh = 0.6, sigma = 0.9 for the depth information and thresh = 0.5, sigma = 0.5 for the color image. In the fourth case (the image with a chair), we set thresh = 0.44, sigma = 0.8 for the depth image and thresh = 0.53, sigma = 0.6 for the color image. The images and contours are shown in Figure 3. As for COPAP method, we reduced the contour points to avoid memory overflow. In addition, we took the different parameters of RANSAC to make a best align between the two images for the four methods.
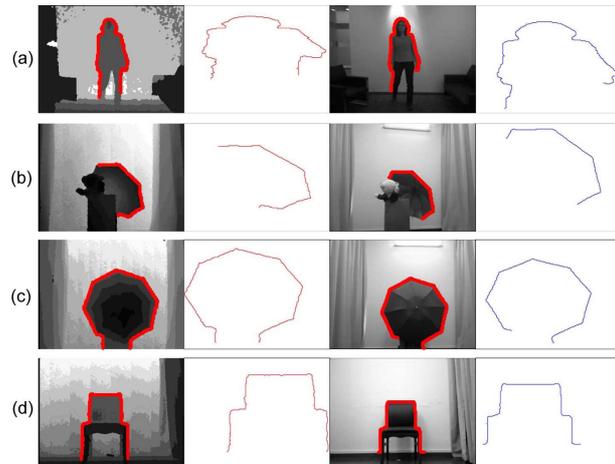

Fig. 3: Images and Contours. The first column shows the depth image, and the 3rd column shows the intensity image. The 2nd and 4th columns show the contours of the 1st and 3rd columns, respectively. (a) First case (670 points for the depth image, 600 points for the intensity image). (b) Second case (400 points for the depth image, 337 points for the intensity image). (c) Third case (875 points for the depth image, 730 points for the intensity image). (d) Fourth case (735 points for the depth image, 600 points for the intensity image).

All compared results of the four methods in the four cases are shown in Figure 4-7.

Table 1: Comparison of avg. RMSE and computational cost.

Comparison	COPAP	ACO	Hungarian	Our Method
	Contour Points			
Woman	337	600	600	600
Bear	337	337	337	337
Umbrella	337	730	730	730
Chair	310	600	600	600
Comparison	*Avg.RMSE*			
Woman	Nan	0.897	0.4405	**0.4135**
Bear	Nan	0.9042	0.4581	**0.4488**
Umbrella	Nan	1.4219	0.4091	**0.4597**
Chair	Nan	1.3931	0.4283	**0.4453**
Comparison	*Computational Cost(s)*			
Woman	63.1784	39.5858	166.7152	**29.4487**
Bear	56.7317	17.4582	30.0764	**6.242**
Umbrella	57.1358	55.3965	412.507	**28.3143**
Chair	43.0963	39.2268	173.3719	**34.5688**

Observing Table 1, for COPAP method, none of the tests provided correct matching. We can find that the shape context technique combined with Munkres algorithm has a significant advantage in detecting correspondence pairs. We can obtain less RMSE and computational cost. In addition, as the number of contour
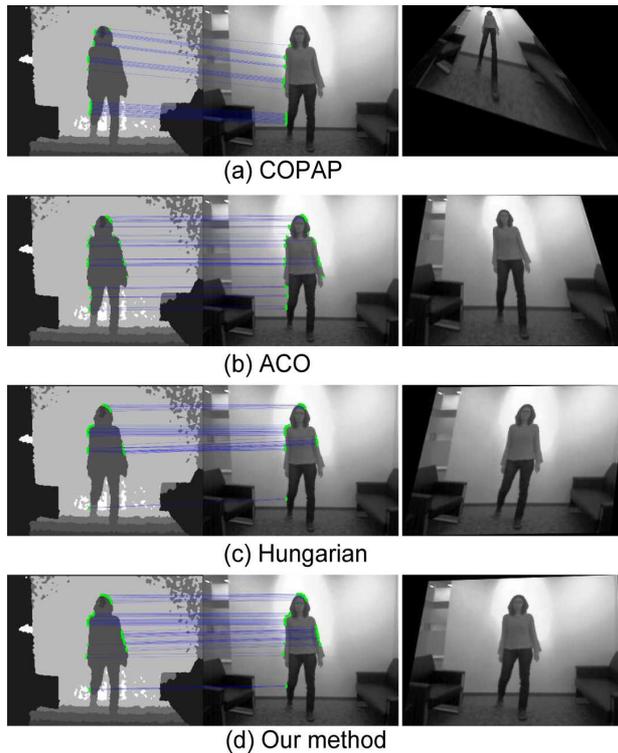
Fig. 4: Correspondence points between two images. (a) COPAP (40 pairs). (b) ACO (100 pairs). (c) Hungarian (66 pairs). (d) Our method (100 pairs).
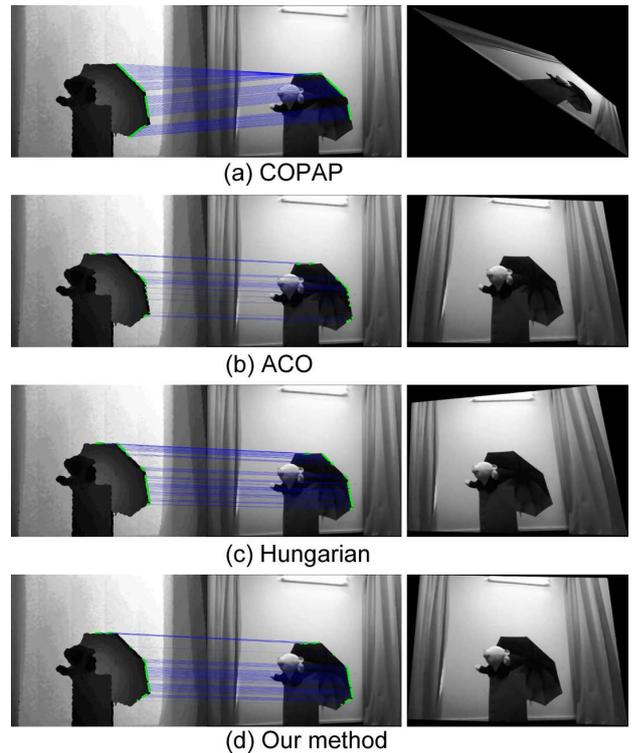

Fig. 5: Correspondence points between two images and the result from image transformation (a) COPAP (139 pairs). (b) ACO (58 pairs). (c) Hungarian (85 pairs). (d) Our method (59 pairs).

points that need to be processed decreases, the execution time of our algorithm also decreases.

5 Conclusion and future work

In this paper, we present a novel method to match color image and depth image captured from Kinect. A contour-based approach is applied to find correspondence points and estimate the transformation matrix between color image and depth image. In the method, the boundary contours are extracted based on Canny edge detector and located to sub-pixel level by a B-spline model. The shape descriptor of contour is computed using shape context and the bipartite graph matching is solved by Munkres method. Finally, the transformation matrix is estimated by RNASAC method. The experiments verify that our algorithm has the highest performance compared with other exist approaches. Our future work will focus on improving the algorithm execution time and implementing more complex scenarios.

Acknowledgement

References

[1] J. Salvi, C. Matabosch, D. Fofi and J. Forest, Image and Vision Computing, **25**, 578 (2007).

[2] Y. Bentoutou, N. Taleb, K. Kpalma, and J. Ronsin, IEEE Transactions on Geoscience and Remote Sensing, **43**, 2127 (2005).

[3] R. W. Cox, A. Jesmanowicz and others, Magnetic resonance in medicine, **42**, 1014 (1999).

[4] L. G. Brown, ACM computing surveys (CSUR), **28**, 325 (1992).

[5] B. Zitova and J. Flusser, Image and vision computing, **21**, 977 (2003).

[6] M. Holm, Proceedings of the 11th Annual International Geoscience and Remote Sensing Symposium, 2439 (1991).

[7] S. Li, J. Kittler and M. Petrou, Proc. European Conference on Computer Vision (ECCV), 857 (1992).

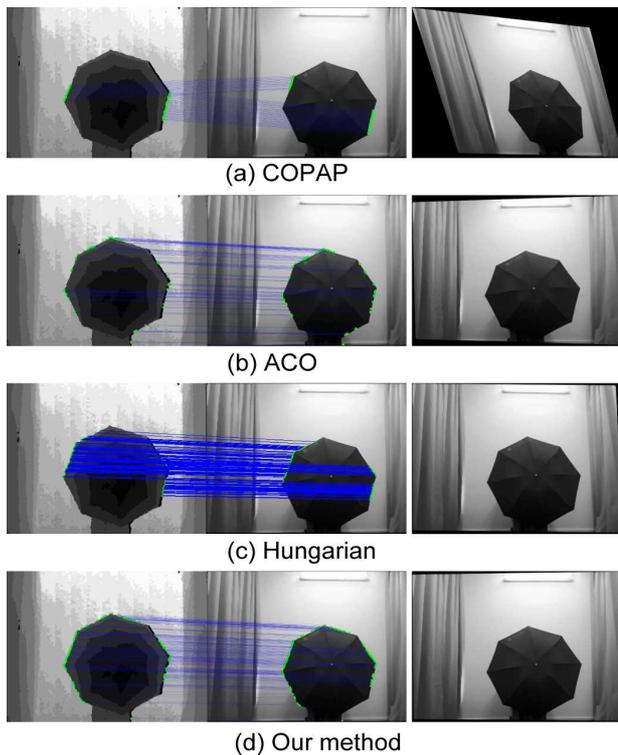[8] F. Eugenio F. Marques and J. Marcello, **6**, 3390 (2002).

Fig. 6: Correspondence points between two images and the result from image transformation. (a) COPAP (55 pairs). (b) ACO (111 pairs). (c) Hungarian (133 pairs). (d) Our method (151 pairs).
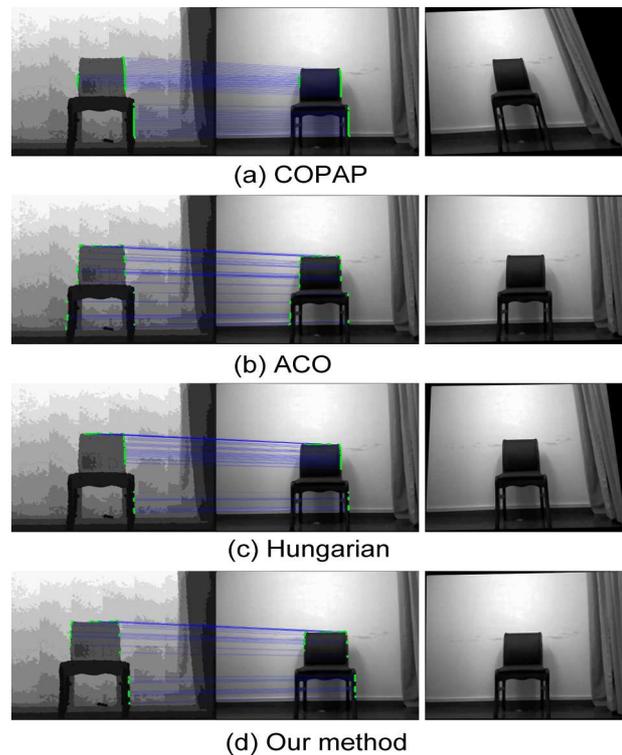

Fig. 7: Correspondence points between two images and the result from image transformation. (a) COPAP (108 pairs). (b) ACO (129 pairs). (c) Hungarian (113 pairs). (d) Our method (94 pairs).

[9] A. Goshtasby, G. C. Stockman, and C. V. Page, IEEE Transactions on Geoscience and Remote Sensing, **GE-24**, 390 (1986).

[10] M. Sester, H. Hild and D. Fritsch, International Archives of Photogrammetry and Remote Sensing, **32**, 538 (1998).

[11] X. Dai and S. Khorram, Geoscience and Remote Sensing, IEEE Transactions on, **37**, 2351 (1999).

[12] R. A. Baggs, DE. Tamir and T. Lam, Proc. Bringing Together Education, Science and Technology, 257 (1996).

[13] D. Ionescu, S. Abdelsayed and D. Goodenough, Canadian Conference on Electrical and Computer Engineering, 710 (1993).

[14] N. Alvarez, J. Sanchiz, J. Badenas, F. Pla and G. Casa, Pattern Recognition and Image Analysis, 405 (2005).

[15] Microsoft kinect. http://www.xbox.com/kinect,2010.

[16] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, etc., Proceedings of the 24th annual ACM symposium on User interface software and technology, 559 (2011).

[17] I. Oikonomidis, N. Kyriazis and A. Argyros, BMVC, **2**, (2011).

[18] A.D. Wilson, ACM international conference on interactive tabletops and surfaces, 69 (2010).

[19] D. G. Lowe, The Proceedings of the Seventh IEEE International Conference on Computer Vision(ICCV), **2**, 1150 (1999)

[20] Q. Zhu, Z. Zhang and Y. Xie, Applied Mathematics & Information Sciences, **6**, 363 (2012).

[21] H. Li, B. S. Manjunath and S.K. Mitra, IEEE Transactions on Image Processing, **4**, 320 (1995).

[22] Z. Li, H. Leung, 10th International Conference on Information Fusion, 1 (2007).

[23] S. Belongie, J. Malik, and J. Puzicha, Advances in neural information processing systems, 831 (2001).

[24] S. Belongie, J. Malik, and J. Puzicha, IEEE Transactions on Pattern Analysis and Machine Intelligence, **24**, 509 (2002).

[25] H. Chui, A. Rangarajan, IEEE Conference on Computer Vision and Pattern Recognition, **2**, 44 (2000).

[26] Z. Tu, S. Zheng and A. Yuille, Computer Vision and Image Understanding, **109**, 290 (2008).

[27] O. Kaick, G. Hamarneh, H. Zhang, P. Wighton, 15th Pacific Conference on Computer Graphics and Applications, 271 (2007).

[28] D. Chitra, T.Manigandan, and N. Devarajan, Universiti Malaysia Perlis, (2009).

[29] D. G. Lowe, International journal of computer vision, **60**, 91 (2004).

[30] S. Se, D. G. Lowe and J. Little, Proceedings of IEEE International Conference on Robotics and Automation(ICRA), **2**, 2051 (2001).

[31] M. Brown, D. G. Lowe, Proceedings of the Ninth IEEE International Conference on Computer Vision(ICCV), **2**, 1218 (2003).

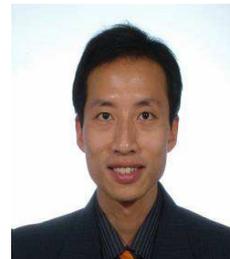[32] I. Gordon, D. G. Lowe, Toward category-level object recognition, Springer, **67** (2006).

[33] J. S. Lim, Englewood Cliffs, NJ, Prentice Hall, **1**, 710 (1990).

[34] C. John, IEEE Transactions on Pattern Analysis and Machine Intelligence, **PAMI-8**, 679 (1986).

[35] F. Ulupinar, G. Medioni, Computer Vision, Graphics, and Image Processing, **51**, 275 (1990).

[36] N. Kanopoulos, N. Vasanthavada and R.L. Baker, IEEE Journal of Solid-State Circuits, **23**, 358 (1988).

[37] E. P. Lyvers, O. R. Mitchell, IEEE Transactions on Pattern Analysis and Machine Intelligence, **10**, 927 (1988).

[38] S. Ghosal, R. Mehrotra, Pattern Recognition, **26**, 295 (1993).

[39] S. Hussmann, H. Thian, Real-Time Imaging, **9**, 361 (2003).

[40] K. Jensen, D.Anastassiou, IEEE Transactions on Image Processing, **4**, 285 (1995).

[41] F. Chen, S. Lin, Computer Vision and Image Understanding, **78**, 206 (2000).

[42] C. De Boor, Applied Mathematical Sciences, (Springer, New York), (1978).

[43] P. Brigger, J. Hoeg and M. Unser, IEEE Transactions on Image Processing, **9**, 1484 (2000).

[44] Y. Wang, E. K. Teoh, IEEE Transactions on Pattern Analysis and Machine Intelligence, **29**, 1853 (2007).

[45] M. Dubuisson, A. K. Jain, Proceedings of the 12th IAPR International Conference on Computer Vision Image Processing, **1**, 566 (1994).

[46] D. T. Lee, IEEE Transactions on Pattern Analysis and Machine Intelligence, **PAMI-4**, 363 (1982).

[47] C. Scott,R. Nowak, IEEE Transactions on Image Processing, **15**, 1831 (2006).

[48] R. Jonker, T. Volgenant, Operations Research Letters, **5**, 171 (1986).

[49] F. Bourgeois, J. C. Lassalle, Communications of the ACM, **14**, 802 (1971).

[50] H. Zhu and M. Zhou and R. Alkins, IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, **42**, 739 (2012).

[51] M. A. Fischler, R. C. Bolles, Communications of the ACM, **24**, 381 (1981).

Qingsong Zhu received the BS and MS degrees in computer science from University of Science and Technology of China (USTC), Hefei, China. He is currently an assistant professor in Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences. His current research interests focus on computer vision, statistical pattern recognition, machine learning and robotics. He is a member of the IEEE.

Tiexiang Wen He is currently an Engineer in Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences. His research interests are in the areas of Computer Vision and Image Processing.

Yaoqin Xie Professor, supervisor of the doctoral students and Overseas High-Caliber Personnel in Shenzhen, China. He received his B.Eng, M.Eng, and Ph.D. from Tsinghua University of China in 1995, 1998, and 2002, respectively. He joined Stanford University as a postdoctoral fellow from 2006 to 2008. Dr. Xies research areas have been focused on image-guided surgery.

Appl. Math. Inf. Sci. **8**, No. 1, 263-271 (2014) / www.naturalspublishing.com/Journals.asp

271

Jia Gu received his master and Ph.D. degree in 2001 and 2005 from Southeast University and University of Rennes, respectively. Since 2008, Dr. Gu was recruited by Shenzhen Institutes of Advanced Technology of Chinese Academy of Sciences, as a full professor.

Lei Wang received his B.Eng in Information and Control Engineering and Ph.D in Biomedical Engineering, in 1995 and 2000, respectively, from Xian Jiaotong University, China. He was with University of Glasgow and Imperial College London during 2000-2008. He is now with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, as a full professor. Dr. Wangs research interests focus on Body Sensor Network. He has published more than 200 scientific papers, authored four book chapters and filed 60 patents.