Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Technology 3 (2012) 196 – 204

Procedia Technology

The 2012 Iberoamerican Conference on Electronics Engineering and Computer Science

Detecting objects using color and depth segmentation with Kinect sensor

José-Juan Hernández-López[a], Ana-Linnet Quintanilla-Olvera[a], José-Luis López-Ramírez[a], Francisco-Javier Rangel-Butanda[a], Mario-Alberto Ibarra-Manzano[a], Dora-Luz Almanza-Ojeda[b]

[a] *Digital Signal Processing Laboratory; Electronics Department; DICIS; University of Guanajuato Carretera Salamanca-Valle de Santiago Km.3.5+1.8 Km, Comunidad Palo Blanco, 36885, Salamanca, Guanajuato, Mexico*
[b] *Departamento de Ingeniería Robótica; Universidad Politécnica de Guanajuato Av. Universidad Norte SN, Comunidad Juan Alonso, 38483, Cortazar, Guanajuato, Mexico*

Abstract

In order to optimize the movements of a robot, every object found in the work environment must not just be identified, but located in reference to the robot itself. Usually, object segmentation from an image is achieved using color segmentation. This segmentation can be achieved by processing the R, G and B chromatic components. However, this method has the disadvantage of been very sensitive to the changes on lighting. Converting the RGB image to the CIE-Lab color space avoids the lack of sensitivity by increasing the accuracy of the color segmentation. Unfortunately, if multiple objects of the same color are presented in the scene, is not possible to identify one of these objects using only this color space. Therefore, we need to consider an additional data source, in this case the depth, in order to discriminate objects that are not in the same plane as the object of interest. In this paper, we introduce an algorithm to detect objects, essentially on indoor environments, using CIE-Lab and depth segmentation techniques. We process the color and depth images provided by the Kinect sensor for proposing a visual strategy with real-time performance.

Keywords: Kinect, Object detection, Mobile robotics, Color segmentation, Depth segmentation

1. Introduction

The ability to detect and identify mobile and fixed obstacles plays an important role for achieving robots autonomy. Many formal methods and techniques from computer science have been proposed for solving this problem. Most common contributions are based on data from stereo images [1, 2] or a combination of visual imagery and 3D scanning [3]. Although these contributions give to systems a certain level of autonomy, object detection in real-time using low cost, compact and easy to use sensors remains a research

Email addresses: jj.hernandezlopez@ugto.mx (José-Juan Hernández-López), al.quintanillaolvera@ugto.mx (Ana-Linnet Quintanilla-Olvera), jl.lopezramirez@ugto.mx (José-Luis López-Ramírez), fj.rangelbutanda@ugto.mx (Francisco-Javier Rangel-Butanda), ibarram@ugto.mx (Mario-Alberto Ibarra-Manzano), luzdora@ieee.org (Dora-Luz Almanza-Ojeda)

2212-0173
doi:10.1016/j.protcy.2012.03.021

goal. Nowadays, cameras that capture RGB images with depth information are available. Such sensing systems allow to propose new attractive solutions for robot navigation problems like 3D mapping and localization [4], object recognition [5], 3D modeling [6], visual odometry [7] among others perception tasks.

This project is concerned with the problem of detecting objects based on a new sensing device RGB-D camera: the Microsoft Kinect sensor [8, 9]. The rapid evolution of this recent technology provides high quality synchronized-images of both color and depth at high-frequency performance. In particular, depth image is obtained by an infrared projected-matrix which bounces over the objects. The IR camera captures these beams, with an intensity variance, and computes the distance to every object in the scene. Here, we process the RGB color image for obtaining a CIE-Lab color space image. Then, we calculate the Euclidean distances among one pixel selected from the object of interest and the rest of the points in the image, this information is interpreted as a probability value for each image point on the color space [10]. Since color and depth information are provided by different sensors inside of the kinect, an homography operation is applied to the probability image in order to obtain a geometrical adequation with respect to the depth image. After that, we combine probability image and depth information for calculating final object segmentation on the scene.

This paper is organized as follows. The section 2 describes our proposed algorithm for object detection. The concept of homography and how to relate different image planes from the dataset is presented in section 3. The advantages and the RGB to CIE-Lab color image transformation is described in section 4. The section 5 describes how to make the color and depth segmentation followed by some experimental results and conclusions.

2. Global segmentation algorithm

The block diagram of our segmentation strategy is presented in Fig. 1.

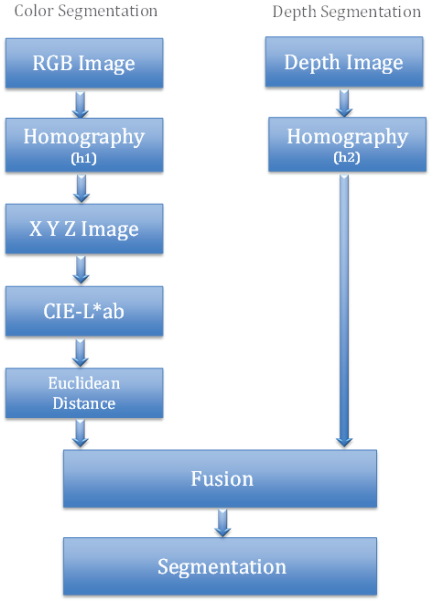

Fig. 1. Block diagram of the global segmentation algorithm

First of all, an image in RGB color space is acquired and an homography transformation is applied to it. Then, the pixels in RGB space color are transformed to the CIE-Lab color space. In order to minimize

the effects due to illumination changes, the euclidean distance in CIE-Lab space is calculated obtaining the level of similarity among the pixels [11, 12]. This procedure allow us to compute the delta values of the image yielding only one chromatic difference on an image referred as to delta image. The delta image can be interpreted as a probabilistic image. Since the chromatic difference between two pixels gives a low value when both pixels are similar and a high value otherwise, in according to this, such difference can be interpreted as a probability value of belonging to a particular color. A convenient value of threshold is used for selecting all the objects with similar color in the scene and from there an initial mask is generated. However, this mask does not isolate any object in particular. Therefore, as interesting object segmentation is not straightforward we use the depth data with the aim of complementing color information.

The right side of the Fig. 1 illustrate the processing carry out to the depth image. This image is also treated by an homography transformation for compensating differences in geometrical position between the sensors. We carried out a statistical analysis of the depth image using the information provided by the initial mask. That is, we combine the statistical values, for each isolate region highlighted in the probabilistic image. This procedure gives a final detection as a resulted image mask with only the desired object. It is important to point out that our statistical analysis employs fundamental operations like mean and variance, since does not require a high computations cost and the results provides the requirements for the segmentation task.

In the following sections, we provide a more detailed information about each module in our approach.

3. Homography

In this section, we describe essential concepts involved in the homography analysis. Homography is a concept in the mathematical science of geometry used when exist two images that were taken by the same camera or both are viewing the same plane but with different angle. The homography relationship is independent of the scene structure [13]. We can estimate the homography that relates them using:

$$X' = RX + T \tag{1}$$

where R is the rotation matrix of 3×3, T is the translation vector of 3×1, and the plane X and X' can be written of the next form.

$$X = \begin{bmatrix} x_a \\ y_a \\ 1 \end{bmatrix}, X' = \begin{bmatrix} x'_a \\ y'_a \\ 1 \end{bmatrix} \tag{2}$$

The translational motion of the robot is positive in the direction of the axis XYZ and the angles are considered positive according to the Fig. 2, and can be expressed using the angles of Euler [14].

A homography is defined mathematically as a 3×3 matrix and can be calculated with equation 3.

$$H = R + T(xyz) \tag{3}$$

Finally, equation 1 can be written as:

$$X' = HX \tag{4}$$

where the H matrix is defined as

$$H = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \tag{5}$$

This matrix allows to transform a point X in the plane P_a to a point X' in the plane P_b [15].

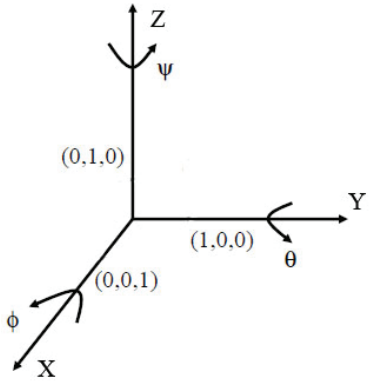

Fig. 2. Coordinate system of a robot with six degrees of freedom.

$$P_b = H_{ab} \times P_a \tag{6}$$

In order for this result to be useful in a practical context, we assume that P_a is an initial image that is transformed to image plane P_b. Uniquely after this transformation, we can use the RGB color image and apply the CIE-Lab transformation, that we will explain in the next section.

4. CIE-Lab transformation

In 1976, the Commission Internationale de l'Eclairage (CIE) introduced the CIE-Lab color space. It was intended to provide a standard, approximately uniform color scale used by everyone in order that color values could be easily compared [16].

The CIE-Lab color space is organized in a cube form as the one shown in the Fig. 3. The L axis runs from top to bottom. The maximum for L is 100, which represents a perfect reflecting diffuser, the minimum for L is zero, which represents black. The $a$and b axes have no specific numerical limits, but positive a is red, negative a is green, positive b is yellow and negative b is blue.

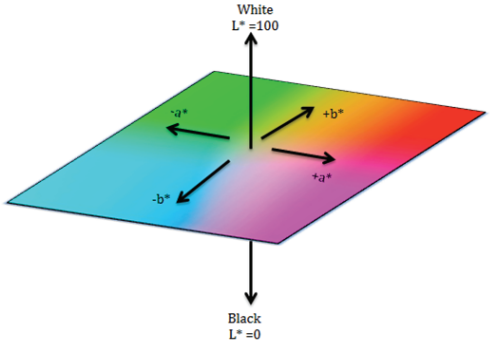

Fig. 3. RGB representation of the CIE-Lab color space

There are delta values associated with this color scale. ΔL, Δa and Δb indicate how much a standard and simple difference from one another in L, a, and b, respectively. These delta values are often used for quality control or formula adjustment. The sign of the delta value indicates if the sample is redder or greener than

the standard. The total color difference, ΔE is a single value that is calculated using ΔL, Δa and Δb.

This color space provides a greater immunity to the illumination changes, since it has a component that represents the light intensity [17]. The coordinates are calculated from a linear transformation of the RGB space components, obtaining therefore equation 7.

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \tag{7}$$

The CIE-XYZ model is based on the color description, the Y component describes the illumination, the X and Z components are normalized spectral ponderation curves based on statistical experiments with human observers.

The color coordinates of every pixel in the CIE-Lab are obtained from non-linear transformation applied on the XYZ coordinates using equations 8, 9, 10 and 11. [16].

$$L^* = 116\left[f\left(\frac{Y}{Y_w}\right)\right] - 16 \tag{8}$$

$$a = 500\left[f\left(\frac{X}{X_w}\right) - f\left(\frac{Y}{Y_w}\right)\right] \tag{9}$$

$$b = 200\left[f\left(\frac{Y}{Y_w}\right) - f\left(\frac{Z}{Z_w}\right)\right] \tag{10}$$

$$f(t) = \begin{cases} t^{\frac{1}{3}} & if\, t > \left(\frac{6}{29}\right)^3 \\ \frac{1}{3}\left(\frac{29}{6}\right)^2 t + \frac{4}{29} & if\, t \leq \left(\frac{6}{29}\right)^3 \end{cases} \tag{11}$$

Where X_w, Y_w and Z_w are tristimulus of CIE-XYZ values with reference to the "white spot".

$$\begin{pmatrix} X_w \\ Y_w \\ Z_w \end{pmatrix} = \begin{bmatrix} 0.9504 \\ 1.0000 \\ 1.0887 \end{bmatrix} \tag{12}$$

The chromatic components a, b and L are used to obtain the delta values and thus calculating the Euclidean distance using equation 13 for obtaining the initial mask of the color segmentation [18] .

$$\Delta E_{ab} = \sqrt{\Delta L^{*2} + \Delta a^2 + \Delta b^2} \tag{13}$$

In the next section, we will describe the procedure for fusioning the depth and color information that will provide us the final detected-object.

5. Color-Depth Segmentation

First, we explain the results using only the color segmentation algorithm. The test consist in several kind of chairs with different color as is shown in Fig. 4.a. The corresponding color segmented image when we select the orange chair is depicted in Fig. 4.b. By examining this figure, we note that the color of the selected object is highlighted in black, being the rest of the objects represented in gray scale. In the Fig. 4.c shows the case when we select the blue chair.

The RGB color image shown in the Fig. 5.a is transformed to the CIE-Lab color space as described in the previous section. This transformation simplifies the color segmentation and it is only a matter of

(a) (b) (c)

Fig. 4. (a) RGB color image (b) Euclidean distance when the orange chair is selected (c) Euclidean distance when the blue chair is selected

(a) (b) (c)

Fig. 5. (a) RGB color image (b) Euclidean distance from each pixel to the point of interest (the probabilistic image) (c) Initial mask

computing the Euclidean distance for each component to the point of interest. An example of this procedure is presented in Fig. 5.b, where the point of interest is chosen at the middle chair.

The probabilistic image allows to identify all the objects with the same color. The initial mask is obtained by eliminating all the objects for which the Euclidean distances are greater than a threshold defined experimentally. This mask is presented in the Fig. 5.c. So far, all the objects of the same color can be extracted from the RGB image by applying the initial mask. However, in order to obtain another mask that contains only the object of interest, the depth information is used. Therefore, the initial mask and the depth information are combined in order to eliminate the objects with different color and geometrical information to the interesting object.

The last step consists on discarding the objects located at different depths taking into account the selected objects position. This procedure is carried out by a statistical analysis of the regions highlighted on the initial mask. That is, we take as our value of reference the depth information of the initial selected point to build a "volume" integrated by all the statistical and adjacent points without discontinuities in position from the initial mask. We employ the term discontinuities in position for referring to the separation among several cloud of points in the depth information. In fact, this statistical analysis provides that our algorithm be more robust even in cases when objects of similar colors are at the same depth in the scene. The Fig. 5.a illustrates the depth image, the Fig. 5.b shows the final segmentation of the interesting object applying the procedure explained in this section.

6. Experimental results

In this section, we describe the experiments carried out to evaluate our method using the technique of color-depth segmentation. First of all, our algorithm was developed in the MATLAB© software, in a PC

(a) (b)

Fig. 6. (a)Depth image (b) Resulting image segmentation

with an i7 2.2 GHz Intel microprocessor and 4 GB of RAM. We use the Microsoft Kinect sensor to provide the RGB-D information.

We determine the efficiency of our algorithm by performing several tests in a typical indoor environment, using two or more objects of similar color but positioned at different depths. In particular, we will describe the experiments carried out using the color images shown in Fig. 5.a and Fig. 7.a. From such images, we select a point belonging to a particular object that we want our algorithm to detect. For the case of Fig. 5.a, this point corresponds to the middle chair at the image. For that of the Fig. 7.a, the chair and the speaker are the most prominent objects in the scene, therefore, the interesting point was selected on the speaker. Both corresponding initial masks are shown in Fig. 5.c and Fig. 7.b, respectively. As we can see in both images, the highlighted objects have the same color. So that, the depth images will provide us the necessary information for discarding undesired objects in the scene (Fig. 5.a and Fig. 8.a). Using the procedure previously explained, we isolate the right interesting object in a scene, in our experiments an orange chair Fig. 5.b and a speaker Fig. 8.b. The computation time for these results was 11 Hz and 15 Hz for Fig. 7.a and 5.a, respectively.

(a) (b)

Fig. 7. (a) RGB color image (b) Initial mask.

(a) (b)

Fig. 8. (a) Depth image (b) Resulting image segmentation.

7. Conclusions

In this paper, we propose a segmentation algorithm that uses the information provided by color and depth images obtained from the Kinect sensor. The experimental results show that our algorithm can effectively detect one object of interest by color and depth, even if more objects of the same color are in the field of view of the sensor. By now, our algorithm achieves a satisfactory frequency of performance (around 15 Hz) allowing us to propose this strategy as a feasible solution in the challenging issue of robot visual navigation. Furthermore, it is relatively cheap in comparison to other technologies like *LADAR* or *SICK* sensors and more robust than stereo vision systems. Also, our strategy can be applied in multiple tasks such as object segmentation, human detection, tracking, and activity analysis of a robot.

In the future, we plan to carry out reconstruction of 3D objects and environments for robot navigation. We intend to increase the speed of our strategy by re-programming the algorithm for embedded systems like an FPGA or a GPU. We also intend to explore other techniques of obtaining objects identification using geometrical and color features.

Acknowledgments

This work was partially funded by the CONACyT with the project entitled “Diseño y optimización de una arquitectura para la clasificación de objetos en tiempo real por color y textura basada en FPGA”, too we would like to express our gratitude for the Universidad de Guanajuato with the project entitled “Sistema de Identificación de objetos mediante los atributos del color y la textura utilizando arquitectura reconfigurable con FPGA” with number 000175/11 and finally to CONACyT for scholarship support for master program with the number 422102.

References

[1] A. Ess, B. Leibe, K. Schindler, L. van Gool, Moving obstacle detection in highly dynamic scenes, in: IEEE International Conference on Robotics and Automation, 2009. ICRA ’09, IEEE, 2009, pp. 56–63. doi:10.1109/ROBOT.2009.5152884.

[2] M.-A. Ibarra-Manzano, A.-O. D.-L., High-Speed Architecure Based on FPGA for a Stereo-vision Algorithm, InTech, 2011, Ch. 5, pp. 71–88.

[3] M. Quigley, S. Batra, S. Gould, E. Klingbeil, Q. Le, A. Wellman, A. Y. Ng, High-accuracy 3D sensing for mobile manipulation: Improving object detection and door opening, in: IEEE International Conference on Robotics and Automation, 2009. ICRA ’09, IEEE, 2009, pp. 2816–2822. doi:10.1109/ROBOT.2009.5152750.

[4] P. Henry, M. Krainin, E. Herbst, X. Ren, D. Fox, Rgbd mapping: Using depth cameras for dense 3d modeling of indoor environments, in: In RGB-D: Advanced Reasoning with Depth Cameras Workshop in conjunction with RSS, 2010.

[5] K. Lai, L. Bo, X. Ren, D. Fox, Sparse distance learning for object recognition combining RGB and depth information, in: 2011 IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2011, pp. 4007–4013. doi:10.1109/ICRA.2011.5980377.
[6] P. Henry, M. Krainin, E. Herbst, X. Ren, D. Fox, Rgbd mapping: Using depth cameras for dense 3d modeling of indoor environments, in: In RGB-D: Advanced Reasoning with Depth Cameras Workshop in conjunction with RSS, 2010.
[7] F. Steinbruecker, J. Sturm, D. Cremers, Real-time visual odometry from dense rgb-d images, in: Workshop on Live Dense Reconstruction with Moving Cameras at the Intl. Conf. on Computer Vision (ICCV), 2011.
[8] Microsoft, `http://www.xbox.com/en-us/kinect` (Oct 2011).
[9] PrimeSense, `http://www.primesense.com` (Sep 2011).
[10] L. M. Blumenthal, Theory and Applications of Distance Geometry, 2nd Edition, Chelsea House Pub, 1970.
[11] T. Gevers, A. W. M. Smeulders, Color-based object recognition, Pattern Recognition 32 (3) (1999) 453–464. doi:10.1016/S0031-3203(98)00036-3.
[12] Y. Ohta, T. Kanade, T. Sakai, Color information for region segmentation, Computer Graphics and Image Processing 13 (1) (1980) 222 – 241.
[13] G. Kayumbi, A. Cavallaro, Robust homography-based trajectory transformation for multi-camera scene analysis, ICDSC 59 (66) (2007) 25–28.
[14] R. I. Hartley, A. Zisserman, Multiple View Geometry in Computer Vision, 2nd Edition, Cambridge University Press, ISBN: 0521540518, 2004.
[15] `http://www.people.scs.carleton.ca/c_shu/Courses/comp4900d/notes/homography.pdf` (Nov 2011).
[16] C. I. de l'Eclairage, `http://www.cie.co.at/` (July 2010).
[17] M. Ibarra-Manzano, M. Devy, J.-L. Boizard, Real-time classification based on color and texture attributes on an fpga-based architecture, in: A. Morawiec, J. Hinderscheit (Eds.), Proceedings of the 2010 Conference on Design and Architecture for Signal and Image Processing, Electronic Chips and Systems design Initiative, 2010, pp. 53–60.
[18] M. Ibarra-Manzano, Vision multi-caméra pour la détection d'obstacles sur un robot de service: des algorithmes à un système intégré, Ph.D. thesis, Université de Toulouse (2011).