Proceedings of the 11th IASTED International Conference
Intelligent Systems and Control (ISC 2008)
November 16 - 18, 2008 Orlando, Florida, USA

OBJECT REGISTRATION THROUGH STATISTICS AND HOUGH TRANSFORM

Andrew Nevin, A. Scottedward Hodel
Department of Electrical and Computer Engineering
Auburn University
Auburn, Al 36849, United States
email: nevinab@auburn.edu, hodelas@auburn.edu

David M. Bevly
Department of Mechanical Engineering
Auburn University
Auburn, Al 36849, United States
email: bevlydm@auburn.edu

ABSTRACT
We investigate different methods to detect objects in a sequence of images with digital image processing techniques, and propose a registration method that combines local statistics with gradients to identify and track objects from frame to frame. These methods are evaluated against both synthetic and captured video data. Common algorithms such as edge detection, Hough transform, and normalized cross correlation work well on synthetic data, but appear to be unreliable with data acquired from a digital camera. Our proposed registration method is shown to be effective with both synthetic data and acquired data.

KEY WORDS
object registration, Hough transform, signal and image processing, robotics

1. Introduction

In this paper, we study different methods of object registration in a sequence of images using digital image processing techniques, and propose a fusion of two methods to optimize the identification of objects for tracking. Object registration uses computer vision to locate key features in an image, and locate the same features in another image and calculate the transformation between the two images. Object registration is necessary for motion tracking, locating objects, or state estimation for unmanned ground vehicles. In this paper, we use object registration to identify specific colored blocks for a robot to navigate towards. The problem statement: use vision sensors on a robot to track multi-color blocks, which requires registering the blocks and extracting distance estimations.

There are two common types of methods that are used to identify objects: area-based and feature-based, as discussed in [1]. Area-based classifications look at the structure of the image with correlation metrics or Fourier analysis. Feature-based methods concentrate on the overall structure of the image, such as edges, line intersections, and curves. In [2], Kumar *et al* use an area-based algorithm to measure statistically a small window of points between two images for registering Synthetic Aperture Radar (SAR) images. A feature-based method, the Hough transform, can be used to detect the shapes of objects. In [3], Strzodka *et al* use the Hough transform to identify different shaped objects. In [4], Chia *et al* use the Hough transform to detect ellipses.

In this paper, we present results of our own investigation into the fusion of area-based and feature-based methods for image registration. We tested various techniques on both synthetic and digital video data. The task for these preliminary experiments is to locate a set of colored cubes (blocks) or rectangles in the image. The digital video images are problematic for edge-detection based methods due to clutter, yet our method works well.

The paper is organized as follows. In Section 2, we review two feature based methods used for object registration: edge detection and the Hough transform. In Section 3, we review an area-based method for object registration: normalized cross correlation. A proposed method based on a combination of feature-based and area-based registration is presented in Section 4. The proposed object registration method is applied to data in Section 5, and we make some conclusions in Section 6.

2. Feature Based Methods

2.1 Edge Detection

Edges in physical objects often result in large local gradients in an image. Edge detection is the process of extracting points where there is a sharp change or gradient of image brightness in an image. The edge detection result is the magnitude of the gradients in the x and y direction of an image. Detecting edges is useful for feature extraction or object recognition. There are many methods to detect edges: Sobel, Canny, Prewitt, and Roberts method as discussed in [5]. The main differences among these methods are the assigned weights of the kernels used to compute the gradients.

2.2 Hough Transform

The Hough Transform is used to detect straight lines in an image. Consider the slope-intercept form $y = mx + b$, where m is the slope of a line, and b is the y-intercept. Notice that m is infinite for vertical lines. The Hough transform instead parameterizes a line by the variables ρ and θ, where ρ is the distance between the line and the origin, and θ is the angle from the origin to the closest point on the line. The equation for a straight line can be written as:

$$y = -\frac{cos\theta}{sin\theta}x + \frac{\rho}{sin\theta} \quad (1)$$

which is simplified to:

$$\rho = xcos\theta + ysin\theta \quad (2)$$

A full derivation and description of the Hough transform is in [6].

3. Area-Based Method

3.1 Normalized Cross Correlation

Image cross correlation is defined as:

$$\gamma(x,y) = \frac{\sum_{s,t}[(f(s,t) - \bar{f}(s,t)][h(x+s,y+t) - \bar{h})]}{\sqrt{\{\sum_{s,t}[f(s,t) - \bar{f}(s,t)]^2 \sum_{s,t}[h(x+s,y+t) - \bar{h}]^2\}}} \quad (3)$$

where h (template) is the image of the object to be located within f. The primary reason to use image correlation is for matching. The cross correlation coefficient γ will be at a maximum where the template and image are co-located or displaced on top of each other. See [5] for details. In [7], Duda *et al* use a fast normalized cross correlation algorithm for self-localization of a robot. There is a disadvantage by using image correlation: there is high probability that a window containing a smooth area without any prominent details will be matched incorrectly with other smooth areas in the reference image due to its non-saliency [1]. It is clear that normalized cross-correlation is not the ideal approach to feature tracking since it is not invariant with respect to imaging scale, rotation, and perspective distortions [8].

4. Proposed method

A proposed method to detect objects is to combine a series of two identification methods. The first method will be based on local averages and variance of surrounding pixels, and the second method will be based on edge detection and the Hough transform.

4.1 Area-Based

Our method uses six statistical parameters to classify on object: the local averages and variance for the red, green, and blue channels in a given pixel window that compose a digital image. A template $h(x,y)$ contains the object that is desired to be located in an image $f(x,y)$. The object in $h(x,y)$ is divided in the computer as a three-dimensional array (M x N x 3). The three dimensions are composed of pixel values for the red, green, and blue space. Statistical analysis is used to characterize each color matrix. First, the average values for each color matrix is calculated. Then the variance for each pixel in the template $h(x,y)$ is calculated. After the template $h(x,y)$ has been classified with this method, the algorithm searches the image $f(x,y)$ for areas that match the parameters for the template $h(x,y)$. If a pixel in $f(x,y)$ matches the parameters given by $h(x,y)$, then the location of the pixel is mapped into a separate output two-dimensional matrix with the same row-column size of the image $f(x,y)$. There is a tolerance value that will allow values that have small deviation from the calculated parameters to be accepted as a match. This method fails to incorporate any information regarding the shape of the object. The result from this method, is then tested with the Hough transform to match the shape of the object.

4.2 Feature-Based

The output matrix from the proposed area-based method is used as the input to the Hough transform. After the Hough transform is performed, the output lines are sent to an algorithm that checks the angle of intersection between the line segments. For the synthetic and acquired data discussed in Section 5, we know that the objects are cubes. We can expect the intersection of the gradients to be perpendiular. To calculate the angle of intersection, the slope of each line is calculated. After all of the slopes are calculated, the algorithm compares the negative inverse slope to the all of the other slopes. If they are equal within a controllable window of tolerance, then the lines are perpendicular. If there is no match, the line segment is thrown out. If the lines intersections are perpendicular, then the probability of the detected region being the template $h(x,y)$ is high. If the angles are not perpendicular, then there is some error. Sets of randomly distributed line segments are used as inputs to test the method for determining if line segments are perpendicular within 3 degrees of tolerance. The plot in Fig 1 contains the randomly created line segments and the output, which is a plot of the lines that are perpendicular. It appears that other sets of lines are perpendicular to each other, which is the reason that the tolerance of error (in degrees) can be adjusted. This procedure works on these randomly created line segments, so the next step is to use this method on the real line segments extracted from the Hough transform.

4.3 Code Outline

Inputs- the template $h(x,y)$ and the image $f(x,y)$
Output- an image of the detected object with lines drawn around it

1. *calculate N_h x M_h of the template $h(x,y)$ and N_f x M_f input image $f(x,y)$*

2. *calculate the average value and variance for the pixel values in the template $h(x,y)$ for each RGB channel.*

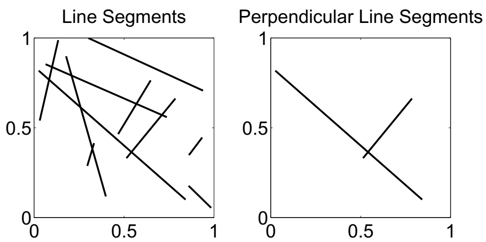


Figure 1. The plot on the left contains 10 randomly generated line segments. The plot on the right are the line segments that are perpendicular to each other.

3. *create an averaging mask with the row-column dimensions N_h x M_h*
4. *increase the resolution of $f(x, y)$ to $2N_f$ x $2M_f$ with nearest-neighbor interpolation*
5. *perform two-dimensional convolution with the averaging mask and the input image $f(x, y)$ for each RGB channel separately.*
6. *calculate the variance for each pixel in the input image $f(x, y)$ within the two-dimensional space spanned by the size of the template $h(x, y)$ and the variance of the output from the two-dimensional convolution with the input image $f(x, y)$ and the averaging mask for each RGB channel separately.*
7. *set the window of tolerance for accepting values as a match with the template*
8. *search the average values and variance from the input image $f(x, y)$ in each RGB channel and compare with the six parameters calculated from the template $h(x, y)$.*
9. *if the values satisfy all six parameters store the pixel locations in a separate output matrix $out(x, y)$ with row-column dimensions $2N_f$ x $2N_f$*
10. *use the output matrix $out(x, y)$ as the input to the Hough transform*
11. *calculate the slope in degrees relative to the horizontal for all line segments*
12. *verify the line segments are perpendicular*
13. *calculate the center and area of the object*
14. *plot all perpendicular line segments onto the input image $f(x, y)$ with row-column dimensions $2N_f$ x $2M_f$.*

5. Experiment and Results

For this experiment, all of the object registration methods discussed are applied to synthetic data (Fig 2) and acquired data (Fig 6). Synthetic data is useful to test digital image

Table 1. Synthetic Red Block Characteristics

	Expected Value	Variance
Red	219	0
Green	60	0
Blue	54	0

Figure 2. Synthetic image which contains the red block (lower left-hand corner) with other colored blocks.

processing algorithms on, because there is no noise, no artifacts from the video coding, and a less busy background. After the object registration methods work on the synthetic data, they are used on the acquired data from the digital camera.

5.1 Synthetic Data

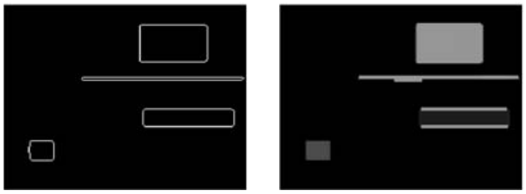
A red-block is created and placed in the bottom left-hand corner in (Fig 2). There is also a line with the same RGB characteristics along with blue and green blocks added to the image. The RGB values for the red-block were set to the values in Table 1. A 6 x 6 pixel block was created to resemble the acquired red block data from the digital camera in the next set of experimental data. These dimensions of the block were chosen because, we wanted a block that was smaller than other blocks in the image. Having a smaller block introduces a higher level of difficulty for digital image processing algorithms. Table 1 describes the synthetic block according to its expected value and variance of each red, green, and blue parameters.

5.1.1 Sobel Edge Detection

The synthetic image in Fig 2 is used as the input to the Sobel edge detection algorithm. The magnitude of the gradients in the x and y direction is displayed in Fig 3(a).

5.1.2 Hough Transform

The result from the Sobel edge detection is used as the input to the Hough transform. The lines are extracted from the Hough transform and plotted on top of the original input image in grayscale in Fig 3(b). The Hough transform detects the strongest edges or gradients in the image. Not all edges were detected by the Hough transform in this experiment. Even if the Hough transform detects all of the edges,



(a) (b)

Figure 3. (a) The magnitude of the x and y gradients using the Sobel kernels. (b) Lines extracted from the Hough transform are plotted on top of the grayscale input image.

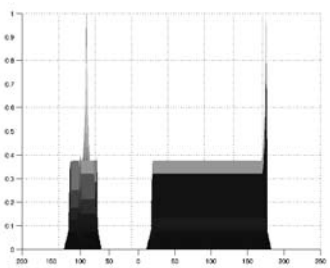


Figure 4. This is a surface plot of the magnitude of the normalized correlation coefficients. The red block is on the left, and the red straight line is on the right. Notice that the maximum coefficient is in the middle of the red block.

there is no method to determine which block is which from this information alone.

5.1.3 Normalized Cross Correlation

The red block is used as the template $h(x, y)$ in the normalized cross correlation in (3). The image in Fig 2 is $f(x, y)$. The output is shown as a surface plot of all of the normalized cross correlation coefficients in Fig 4. The maximum coefficient is where there is a match of $h(x, y)$ and $f(x, y)$. The red block is the group of coefficients to the left. The group to the right are from the red straight line. The highest correlation value is 1, which is near the middle of the red block. This method successfully registered the template $h(x, y)$ within the image $f(x, y)$. However, there are high correlation values for the red line, which could cause problems in other cases.

5.1.4 Proposed Method

The new method is tested on the synthetic image in Fig 2, and the output image is shown in Fig 5. The red block was detected. The next step is to test this method on the acquired data, which has more background noise.

Figure 5. The output from the algorithm detects the red block in the synthetic image in Fig 2.

5.2 Acquired Data

Digital video was obtained in an arena designed for a robotic competition. Four different colored blocks were placed in front of a robot. A digital camera was placed on top of a robot in the rink, and was pushed forward for a couple of seconds. A total of 299 images were used as data to process. The thirty-seventh frame is shown in Fig 6. A red block is in the top right corner. Edge detection, the Hough transform, and the proposed method are applied to the image sequence. We illustrate an approach to detect and track it over the image sequence.

5.2.1 Sobel Edge Detection

The Sobel method for detecting edges is used on the input image in Fig 6. The magnitude of the gradients is displayed in Fig 7(a). All edges are detected, however it is difficult to segment any of the blocks because of background noise from the input image. There is a dark blue surface in the bottom right corner, there is a white sandy surface in the middle, there is a gravel surface in the back, and there is a window in the background. For these reasons, it is difficult to successfully register the blocks.

5.2.2 Hough Transform

The Hough transform of the image was performed, and the lines were plotted on top of the input image. The output image is displayed in Fig 7(b). The strongest edges in the image comprise the border of the rink and the window in the background. Even if the blocks edges were detected, the color of the blocks would be difficult to identify from the results, because the image is converted to a binary image.

5.2.3 Normalized Cross Correlation

The red block is used as the template h(x,y). The input image is the image from Fig 6. An issue in using normalized cross correlation is that $h(x, y)$ and $f(x, y)$ need to be in grayscale. Intensity information in the RGB space is lost in this process. As a result, there are many areas away from the block with high correlation coefficients.

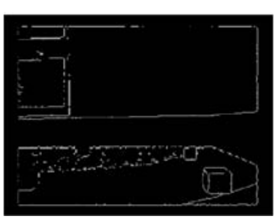
Figure 6. The output from the digital camera. Included in the image are the four blocks to be detected. There is a black background on the rink, there is a blue and white surface, and there are some pebbles.

Table 2. The Acquired Red Block Characteristics

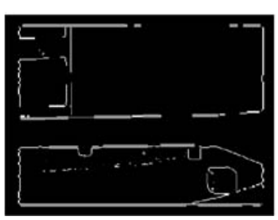
	Expected Value	Variance
Red	202	85
Green	71	40
Blue	33	58

5.2.4 Proposed Method

The red-block was copied from the input image in Fig 6 and saved as the template $h(x, y)$. It is characterized by the values in Table 2. The proposed identification method is performed on the entire image sequence (299 frames) taken from the digital camera. The area-based search for local averages and variance for each pixel is performed, then the output is sent to the Hough transform. The lines that are extracted from the Hough transform are sent into an algorithm to check that the lines are perpendicular. A vector from the center pixel in the image $f(x, y)$ is connected to the center of the registered object. The final results are displayed in Fig 8. The same template is used for the entire image sequence. The registration method located $h(x, y)$ in all of the images. If $h(x, y)$ is smaller than the object in $f(x, y)$, it appears to have minimal effect on the outputs.



(a)



(b)

Figure 7. (a) Magnitude of the x and y gradients using the Sobel kernels. (b) Lines extracted from the Hough transform plotted on top of the Sobel edge detection in (a).

(a) Frame 37

(b) Frame 64

(c) Frame 91

(d) Frame 129

Figure 8. Four outputs from the proposed object registration method.

Frame	X position	Y position	Area ($pixels^2$)
5	326	441	113.12
10	321	442	305.25
28	331	447	242.25
37	327	446	414.62
50	343	445	477.62
64	326	452	520.00
75	344	468	469.00
91	358	496	691.50
110	378	512	1387.38
129	392	548	2071.25

Figure 9. The center of the registered object red block in ten selected frames from the image sequence is shown above.

5.3 Tracking

After the object template $h(x, y)$ is registered in the input image $f(x, y)$, the center pixel $center(x, y)$ and area of the object are calculated. The area of the object is measured, because the area of the object is proportional to the distance of the object to the camera. In [9], Atkins *et al* use the area of an object in a color blob tracking algorithm to extract distance from the camera to the object. This measurement will provide information about the z distance from the camera. Typically if the area increases, stays the same, or decreases, the distance from the camera will decrease, stay the same, or increase respectfully.

6. Conclusion and Futrue Work

6.1 Conclusion

In this paper, we reviewed some old methods for object registration in digital images, and propose a new one, that is composed of a statistical area-based method and a feature-based method. Edge detection and the Hough transform by themselves are not very useful in detecting objects in a busy environment. Normalized-cross correlation proved to be very useful for the synthetic data, but for the acquired data, there were many other areas away from the object with high correlation coefficients. This method successfully registered the object in every frame. Issues with this method are computation time. For example in step 5 of the algorithm in Section 4.3, there are approximately 184 million floating point operations. This amount of operations is due to the increased resolution of f(x,y) from step 4, and performing two-dimensional convolution in the spatial domain in step 5. Implementing step 5 in the frequency domain will reduce the number of operations. Another issue is the algorithm searches the entire frame for the template $h(x, y)$. The next section discusses a method to reduce the search space.

6.2 Future Work

If we can predict the motion of the object from frame to frame, we can limit the search area for the algorithm. Instead of searching the entire frame each iteration, search the area where the predicted object is going to be in the next frame. The following equation provides an open-loop estimation of the location of the object and the size of the object:

$$Obj_{k+1}(x, y, z) = \begin{bmatrix} x_k + (x_{k-2} - x_{k-1}) \\ y_k + (y_{k-2} - y_{k-1}) \\ z_k + (z_{k-2} - z_{k-1}) \end{bmatrix} \quad (4)$$

where the variables x and y refer to the x and y coordinate of the center of the registered object, and z refers to the area of the registered object. The largest prediction error for the location of the center of the object for the x-cooridnate is about 20 pixels. This error is the same for the prediction of the center of the object for the y-coordinate. The prediction of the area of the block error is significantly higher, due to the linear prediction model, and the nonlinear growth of the area of the block as it gets closer to the camera. To further reduce these error predictions, a Kalman filter will be implemented.

References

[1] Jan Flusser and Barbara Zitová, Image registration methods: a survey, *Image and Vision Computing*, vol. 21, 2003, pp. 977-1000

[2] Suvesh Kumar, Vinay Rishiwal, and P.N. Joglekar, Area Based Approach to Registration of SAR Images with Sub-pixel Accuracy, *TENCON 2006. 2006 IEEE Region 10 Conference*, pp.1-4, Nov. 2006

[3] Robert Strzodka, Ivo Ihrke, and Marcus Magnor, A graphics hardware implementation of the generalized Hough transform for fast object recognition, scale, and 3D pose detection, *Image Analysis and Processing Proceedings 12th International Conference*, 2003, pp. 188-193

[4] A.Y. Chia, M.K.H. Leung, How-Lung Eng, and S. Rahardja, Ellipse Detection with Hough Transform in One Dimensional Parametric Space, *Image Processing, 2007. ICIP 2007. IEEE International Conference*, vol.5, no., pp.V -333-V -336, Sept. 16 2007-Oct. 19 2007

[5] R. C. Gonzalez and R. E. Woods, *Digital Image Processing, 2nd ed.* (New Jersey: Prentice Hall, 2002.)

[6] Richard O. Duda, Peter E. Hart, Use Of The Hough Transformation To Detect Lines and Curves In Pictures, *Comm.ACM*, vol 15, no. 1, pp. 11-15 (January 1972).

[7] K. Briechle, and U.D. Hanebeck, Self-localization of a mobile robot using fast normalized cross correlation, *Systems, Man, and Cybernetics, 1999. IEEE SMC '99 Conference Proceedings. 1999 IEEE International Conference*, vol.4, no., pp.720-725 vol.4, 1999

[8] J.P. Lewis, Fast Normalized Cross-Correlation, *Vision Interface*, *Canadian Image Processing and Pattern Recognition Society*, 1995

[9] E. M. Atkins, J. A. Lennon, and R. S. Peasco, Vision-based following for cooperative astronaut-robot operations, *Aerospace Conference Proceedings, 2002. IEEE* , vol.1, no., pp. 1-215-1-224 vol.1, 2002