Core-Shell Detection in Images of Polymer Microbeads*

Yeonggul Jang, Byunghwan Jeon, and Yoojin Chung**

Department of Computer Science and Engineering
Hankuk University of Foreign Studies
Kyonggi, 449-791, Republic of Korea
{jyg1722,xpsxm85,chungyj}@hufs.ac.kr

Abstract. Microbeads of various size with complex core-shell structures are widely used in many applications such as drug delivery. During synthesis, it is important to characterize the beads' size such that uniform properties can be obtained from uniform size. The core-shell structures can be imaged with SEM (scanning electron microscope) or TEM (transmission electron microscope) but there are no available methods to quantitatively analyze the size and distribution automatically. In this paper, we propose two automated core-shell detection methods using Hough transform and generalized Hough transform. We show the capabilities of these methods using OpenCV and compare the relative advantages and limitations.

Keywords: Counting core-shells, Hough transform, OpenCV.

1 Introduction

Uniform sized polymer particles are widely used in many engineering and clinical applications such as drug delivery. In all applications, it is important to have tight control over the size and chemistry of the particles. For particles or beads with core-shell structure, this requirement is stricter such that the measurements and distribution should be analyzed for each set of experiment. As the particles are imaged with SEM (scanning electron microscope) and TEM (transmission electron microscope), an automated method that can measure and various parameters such as size of the bead, relative size of the core-shell structure, standard deviation and other parameters would be useful for the synthetic chemists.

The Hough transform is a feature extraction technique used in image analysis, computer vision, and digital image processing [1-5]. The Hough transform was initially developed to detect analytically defined shapes (e.g., lines, circles, ellipse etc.). The generalized Hough transform [2] is the modification of the Hough transform using the principle of template matching. This modification enables the Hough transform to be used for the detection of an arbitrary shape (i.e., shapes having no simple analytical form). OpenCV (Open Source Computer Vision Library) [6] is a library of programming functions mainly aimed at real-time computer vision, developed by Intel.

* This research was supported by Hankuk University of Foreign Studies Research Fund of 2012.
** Corresponding author.

T.-h. Kim et al. (Eds.): MulGraB/BSBT/IUrC 2012, CCIS 353, pp. 9–15, 2012.

In this paper, we propose two automated core-shell detection methods in image of polymer chemistry using Hough transform and generalized Hough transform. We show their experimental results using OpenCV and discuss their limitations.

2 Methods

2.1 Hough Transform

We first used Hough transform to perform the characterization. The steps are as follows.

Step 1) Image input: Any type of image can be used, B/W or color. If input image is a color with 3 channels (red, green, blue), it is transformed into 1channel mode.

Step 2) Smoothing: Noise in an image can be removed using smoothing to improve the signal to noise (S/N) ratio.

Step 3) Edge detection: Canny edge detection is performed using the matrix in Figure 1.

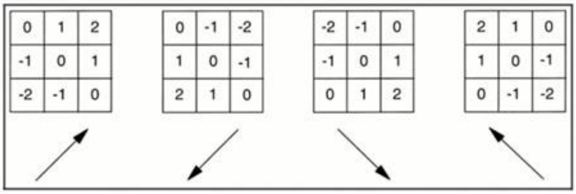

Fig. 1. The matrix for Canny edge detection

Step 4) Determining threshold value: Figure 2 shows the SEM micrograph of polymer core-shell structures. Each image has a characteristic value such as bead radius. When performing Hough transform, the number of core-shells in image depends on radius R. Thus this step can be used to automatically determine the threshold value for each image.

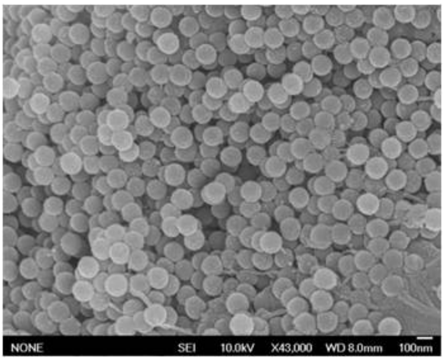

Fig. 2. SEM micrograph of polymer core-shell structures

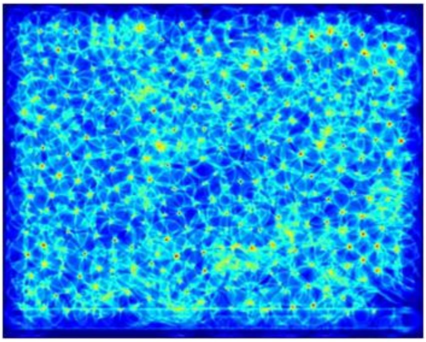

Fig. 3. Image after Hough transform

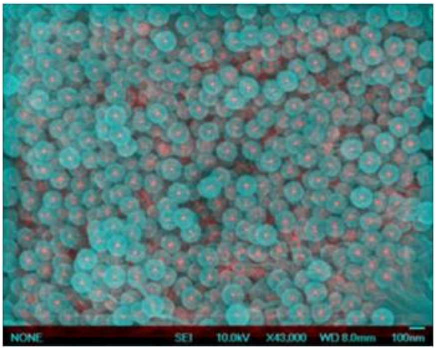

Fig. 4. Overlap image of Fig. 2 and Fig. 3

Step 5) Hough Transform: Each circle in SEM micrograph of polymer core-shell structures in Fig. 2 has slightly different radius value R. The center of each circle is determined after performing edge detection using threshold value. Thus the total number of the circles depends on threshold value. Figure 3 shows the image after performing Hough Transform. Figure 4 is the overlap image of Figure 2 and Figure 3. In Figure 4, red dots with higher intensity indicate point of intersection for higher number of circles.

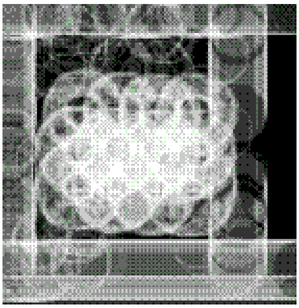

Fig. 5. Traces of circles after Hough transform

For circle detection in Hough transform, three dimensional parameter space (x0, y0, r) is used, where x0 and y0 are the coordinates of a circle center, r is the radius of a circle and its equation is in (1).

$$2(x - x_0) +2(y - y_0) = 2r \qquad (1)$$

$$\text{The parameter vector is } p= (x_0, y_0, r) \qquad (2)$$

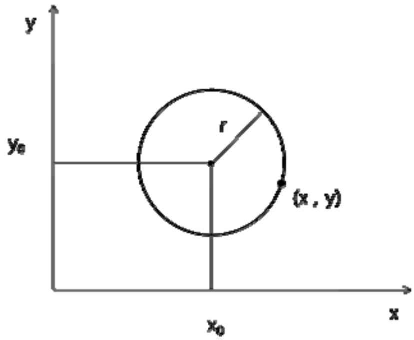

Fig. 6. Definition of circle in Hough Transform

The pseudo code for circle detection in Hough transform is as follows.

```
For each edge point (x0, y0)
    For (x0 = x0_min ; x0< x0_max x0++)
        For(y0= y0_min ; y0<= y0_max ; y0++)
            R=sqrt((x-x0)^2 + (y- y0)^2);
            Accumulator[r][x0][ y0]++;   //Voting
            Find local maxima in accumulator [r][x0][y0]
that higher than threshold
```

Step 6) Selecting the correct peak: A red peak or a point indicates high frequency of matches for a circle. These points are the centers of the core-shells and therefore the number of red points is equal to the number of core-shells. Using this approach, we can also reduce the total number and remove some spurious noise while keeping the bright spots. If the core-shell's size is increased, the edge's size will increase as well and there will be many points clustered around the real point. Therefore, it is important to obtain correct threshold values by removing the noise around the real center point. Next task is to determine the number of real center points by counting the number of overlapping edges. In some cases, only half of the edges might show up due to overlapping images. In addition, as a core-shell's size increases, the number of pixels that are at the edge of a core-shell will increase as well.

2.2 Generalized Hough Transform

As shown in Figure 2, because of the high density of core-shells in an image, there are many overlapping cores, showing only those that are in foreground. Those in the back appear as partial spheres as they are shadowed. In order to account for those that are partially blocked, we will need to implement a method to count them even if they do not show complete outline. In such cases, a template is needed to count incomplete crescent-shaped objects using generalized Hough transform, which consists of modeling and detection steps.

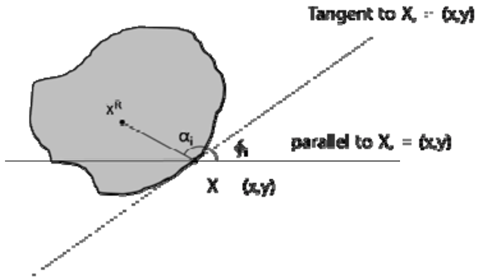

Fig. 7. Target shell modeling for non-symmetric shaped shells

Figure 7 shows an example of non-symmetric shaped object. In modeling step, after detecting the edge, following algorithm is performed to generate an R-Table. We need to build a separate R-table for each different object. The pseudo code for making R-Table (modeling step) is as follows. The R-table allows us to use the contour edge points and gradient angle to re-compute the location of the reference point.

```
Pick a reference point : X^R=(x_c,y_c)
For each edge point, X_i = (x_i, y_i)
    Calculate tangential angle ∮ i
    Calculate V_i=X^R-X_i and find norm r_i
    Direction angle α_i of v_i
    Store r_i, α_i in R-Table with an index of
    Ψ_i, quantized value of Φ i
End for
```

Its detection step is described in the following pseudo code. At first, it calculates Φ i for each pixel as in modeling step. Acc[][] in the pseudo code is the accumulated array (vector) of the candidate reference points X^R for a given model. Then it adds the vector to coordinate (x_i, y_i) of current edge pixel and finds out candidate standard point. Finally, possible locations of the object contour are given by local maximum in the array Acc[][].

```
Clear accumulator array : Acc[][] =0
For each edge point : E_i =(x_i, y_i)
    Calculate tangential angle Φ_i
    For every item in R-Table indexed by Ψ_m quantized value of Φ i
        Find X^R_m,j =E_i+V^j_m, where V^j_m=r^j_m ∠α^j_m
        X^R_m,j=E_i+V^j_m=(x^R_m,j, y^R_m,j)
        Acc[y^R_m,j][x^R_m,j]<- Acc[y^R_m,j][x^R_m,j]+1
    End For
End For
```

3 Experiments

When the data contain non-overlapping images of low density particles, it is relatively simple to determine the number of core-shells using straight forward methods. This is shown in Figure 8, where each particle can be accurately detected and the number of core-shells can be counted with high accuracy in straight forward manner. However, when the images contain high density of core-shells (shown in Figure 9), there are many overlapping particles making the determination of total number and size of particles difficult. For these types of images, we have successfully implemented Hough transform for irregularly shaped particles and successfully taken into account of non-symmetric crescent shaped particles.

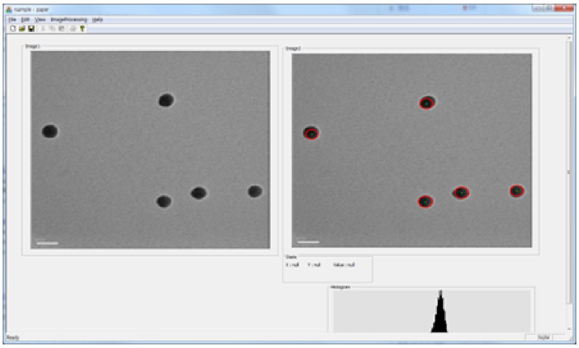

Fig. 8. Result of simple shaped core-shell structures without overlapping

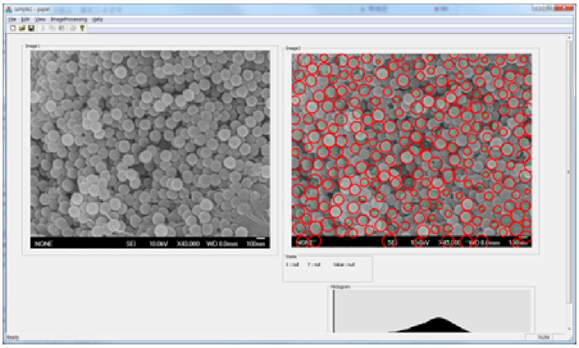

Fig. 9. Result of high density number of core-shells

4 Conclusion

We have developed a robust method for accurately analyzing the total number and size of particles from images. Using appropriate edge detection protocol together with generalized Hough Transform, we have successfully analyzed the images of uniform polymer particles synthesized by polymer chemists.

Improvements to accuracy may result from better edge detection methods using GHT based on pixel domain. Using generalized Hough transform references R-table, it is possible to generate templates automatically by using another algorithm and reference R-tables of template for each case.

References

1. Shapiro, L., Stockman, G.: Computer Vision. Prentice-Hall, Inc. (2001)
2. Ballard, D.H.: Generalizing the Hough Transform to Detect Arbitrary Shapes. Pattern Recognition 13(2), 111–122 (1981)
3. Basça, C.A., Taloş, M., Brad, R.: Rondomized Hough transform for ellipse detection with result clustering. In: Proceedings of International Conference on Computer as a Tool, vol. 2, pp. 1397–1400 (2005)
4. Ogawa, K., Ito, Y., Nakano, K.: Efficient Canny edge detection using a GPU. In: Proceedings of International Workshop on Advances in Networking and Computing, pp. 279–280 (2010)
5. Duda, R.O., Hart, P.E.: Use of the Hough Transformation to Detect Lines and Curves in Pictures. Comm. ACM 15, 11–15 (1972)
6. `http://opencv.org/`