Automatic detection and counting of circular shaped overlapped objects using circular hough transform and contour detection

Conference Paper · June 2016

DOI: 10.1109/WCICA.2016.7578268

Jianjun Ni

Hohai University

Zubair Khan

Shanghai Jiao Tong University

Syed Kamran Haider

Hohai University

2016 12th World Congress on Intelligent Control and Automation (WCICA)
June 12-15, 2016, Guilin, China

Automatic Detection and Counting of Circular Shaped Overlapped Objects Using Circular Hough Transform and Contour Detection

Jianjun Ni, Zubair Khan, Shihao Wang, Kang Wang and Syed Kamran Haider

Abstract— **It is an important task to count the number of the objects in the image automatically. Because the objects in the image are sometimes overlapped with each other and even covered with different shaped objects, it is very difficult to detach these objects before counting them. Circular Hough transform has been used extensively to detect and count overlapped objects with circular shapes. However it can not detach the overlapping circular objects efficiently, and the counting task will provide inaccurate results when the circular objects are covered with random shaped objects. To deal with this problem, an integrated method is proposed based on the Hough Transform and Contour Detection methods. Furthermore, a software is implemented using the open source computer vision library Opencv 3.0 with visual studio 2012. The experimental results show that the proposed method can detach and count the objects accurately.**

I. Introduction

COMPUTER vision is a field that includes methods for capturing, pre-processing, analyzing, and understanding images. A theme in the development of this field has been to duplicate the abilities of human vision by electronically perceiving and understanding an image [1].

Computer vision is a vast field, with a lot of different sub-fields, and all the sub fields are different from each other. In these sub-fields of computer vision, it is an important task to count the objects in the image [2], [3]. Normally counting objects with a naked human eye looks easy, but the main thing is it will take some time until the person completes counting all the objects in a specified area. The human errors in counting objects will decrease the efficiency of the work and it is also very tiring for the person performing the task. There can also be some overlapping objects in the scene which make the task challenging to accomplish [4].

For the detection of any type of objects in image processing, it is necessary to distinguish the object you want to detect from other objects in the scene or in the background. This can be achieved with several image processing techniques like applying threshold technique on the images and obtaining a binary image to detect the blob of white pixels. Morphological approaches like erosion and dilation can also be applied. However, depending upon the scenerio there are some deficiencies existing in the methods introduced above, for example, the morphological based approach is not so much efficient to deal with clumps of circular shaped objects e.g. blood cells [5].

After detection of the objects, the next step is to count objects. For more accuracy, it is necessary to segment all the objects very well before counting them, which means that the objects should be separated from each other. This detached object scenario provides us the advantage of counting random shaped objects precisely.

Various approaches have been proposed to deal with the counting problem by images. For example, Rizon, et al., [6] used the Circular Hough Transform (CHT) technology to detect single circular object, where the histogram equalization technique is applied as a first step to increase the contrast in the image. After that canny edge detection is applied, followed by separability filter and finally CHT is applied to detect circles. Orazio, et al., [7] proposed a method based on Circular Hough transform in combination with neural networks to differentiate ball from other circular shaped objects.

On the basis of these works above, an integrated approach is proposed to deal with the object counting task in this paper, which can be applied to an image having random colors of circular objects and non-circular objects. Some experiments are carried out, where the test image has some circular shaped overlapping objects while one circular and non-circular shaped objects are overlapped with each other. Our proposed approach will successfully segment all the overlapping objects and will also count them efficiently.

The rest of the paper is organized as follows. In Section II, the proposed approach is discussed. Section III focused on implementation results. Some discussions are given out in Section V. At last, the conclusion is given in Section IV.

II. Methodology

In this study, an integrated approach is proposed to realize the separation and counting task by images, and the basic contents of this task are summarized in Fig. 1.

In the proposed integrated approach, there are seven main steps: (1) Firstly, the image is retrieved and then it is converted into a grayscale image, for the sake of fast computation and data reduction. (2) In the next step, the canny edge detection is applied to detect the edges [8]. Edges are detected at the locations where pixels have sharp changes in their intensities, as some foreground pixels are same as the background so those edges are not detected properly. After

Jianjun Ni, Zubair Khan, Shihao Wang, Kang Wang and Syed Kamran Haider are with College of IOT Engineering, Hohai University, Changzhou, 213022 China. Jianjun Ni, Zubair Khan, Shihao Wang are also with Jiangsu Provincial Collaborative Innovation Center of World Water Valley and Water ecological civilization, Nanjing, 211100, China (e-mail: jianjun_ni@hhu.edu.cn).

This work was supported by the National Natural Science Foundation of China (61203365, 61573128), the Jiangsu Province Natural Science Foundation (BK2012149), the Fundamental Research Funds for the Central Universities (2015B20114, 2015B14614), and the Jiangsu Province Innovation Project for Postgraduate (KYLX15_0496).

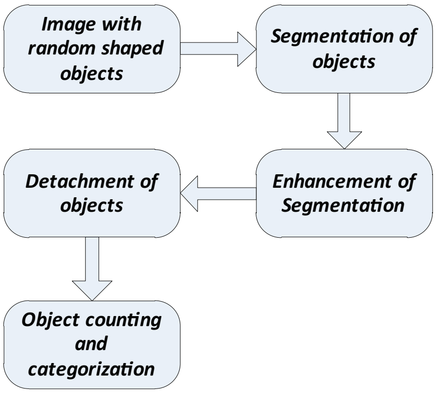


Fig. 1. overall flow diagram of the procedure adopted.

the edge detection step there is still some noise caused by the background, because image is not so smooth and also has illumination changes. Setting the parameters of the canny edge detector will remove the noise pixels to the greatest extent and the edges of the desired objects become clearer. (3) The next step is to find the contours of the detected edges in the image [9]. To create easiness in the next steps, all the contours are drawn with white color on a black background, for having a better visibility and a greater pixel intensity difference. (4) After that circular Hough transform is applied, which is a variation of Hough transform, to detect the circular object contours. (5) Then the contours of circular shape objects are drawn and filled internally. (6) The circular Hough transform is again applied but this time with a different purpose, to get the absolutely detached objects. (7) In the last step any technique can be applied to detect the white blobs in the image. In the proposed approach, the contour detection technique is used again to count objects irrespective of their shapes. Then those results are drawn on the original image to display the final output. The work flow of the proposed approach is shown in Fig. 2 and the main processes of the proposed approach are introduced as follows in detail.

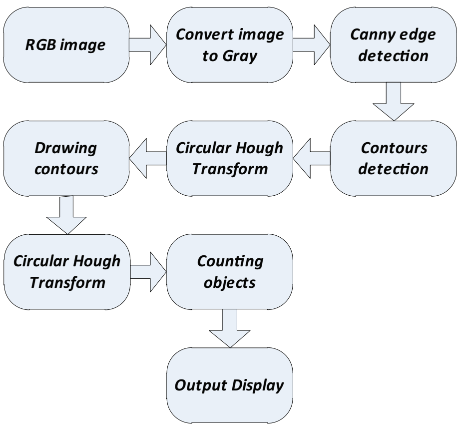


Fig. 2. Flow diagram of implementation.

A. Edge Detection

The aim of performing edge detection in general is to significantly reduce the amount of data in an image, which will be used in the further processing. This step increases the computation speed of the approach while preserving the structural properties of the image. Edge detection in image processing can be applied to the RGB images and can also be applied to the grayscale image. Applying on the grayscale images decrease the computational requirements. Several algorithms exist for edge detection, such as Prewitt, Sobel, Roberts and canny are available [10], [11], [8], out of which the performance of canny is far better. Even though it is quite old, it has become one of the standard edge detection methods and is used in our work.

The Canny Edge Detection algorithm consists of five steps:

(1) Smoothing: In this step, the image is blurred to remove noise, so that it can be used for further processing.

(2) Finding gradients: The edges should be marked where the gradients of the image has large magnitudes. It means that the edges where the intensity of the image has an abrupt change will be marked as an edge.

(3) Non-maximum suppression: Non-maximum is actually a technique to perform edge thinning. Only local maxima should be marked as edges, and edges with the values less than the local maximum are discarded to obtain more accurate and sharp edge detections.

(4) Double thresholding: Potential edges are determined by thresholding. Even after the application of non-maximum suppression, the detected edge pixels are quite good enough to present the real edge. However, there are still some edge pixels at this step caused by noise and color variation. In order to remove the noise caused by these two factors, it is required to filter out the edge pixels having the weak gradient value and only preserve the edges with the high gradient value.

(5) Edge tracking by hysteresis: After performing all the above steps we obtain almost accurate edges but still there can be edges which are separate, raising the possibility of the edges detected caused due to noise, so the final edges will be determined by suppressing all edges that are not connected to a very certain or strong edge.

B. Contour detection

Contour detection is a technique to find out the boundaries of pixels with sharp intensity changes. Contours consist of a set of points connected to each other, most likely to be located on the outlines of objects. Contour detections take a binary image as an input which is the output of the canny edge detector or a binary image obtained by applying the global thresholding technique on a grayscale image. It calculates the boundaries of objects, makes a hierarchy of the object contours to keep the record of the holes inside the parent objects. This information can be used to extract and draw any contour depending upon the user requirement.

C. Hough Transform

The Hough Transform (HT) has been recognized as a very powerful tool for the detection of parametric curves in images[12], [13]. It is implemented by a voting process that maps image edge points into manifolds in a properly defined parameter space. The Circular Hough Transform (CHT) is one of the modified versions of the HT. The purpose of using CHT is to find the circular patterns within an image scene [14]. The CHT is used to transform a set of feature points in the image space into a set of accumulated votes in a parameter space. Then, for each feature point, votes are accumulated in an accumulator array for all parameter combinations. The array elements that contain the highest number of votes indicate the presence of the shape. A circle pattern is described by

$$r = \sqrt{(x - x_0)^2 + (y - y_0)^2} \tag{1}$$

where x_0 and y_0 are the coordinates of the center and r is the radius of the circle.

The workflow diagram of CHT to detach circular objects in this paper is shown in Fig. 3.

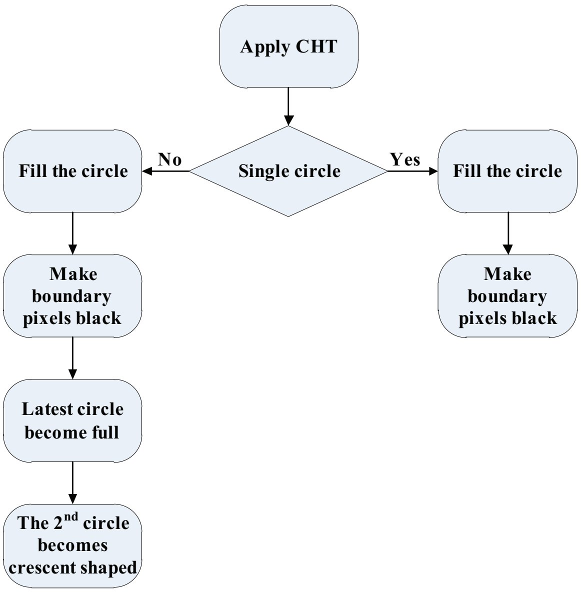
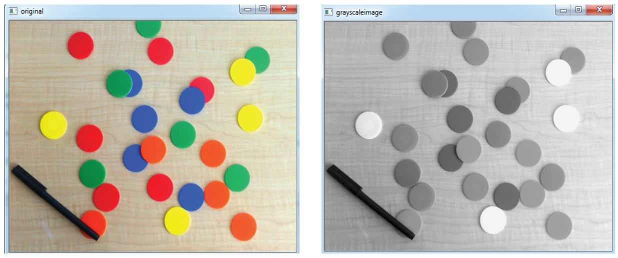


Fig. 3. The workflow diagram of detachment process based on CHT.

III. Implementation Results

To test the performance of the proposed approach, some experiments are carried out. In this study, the experimental work is implemented using opencv3.0 libraries linked with visual studio 2012. Opencv with C++ is really efficient and has fast processing and less time consumption for performing computer vision tasks. Opencv is adopted as an implementation tool in our method [15].

In the experiments, an RGB 8-bits 3-channels image is imputed from the memory firstly (see Fig. 4(a)), which has resolution of 512*386 pixels [14]. Then the image is converted into the grayscale image which will be used in the next steps to save the computation time, as time is considered to be the most important factor in image processing (see Fig. 4(b)).

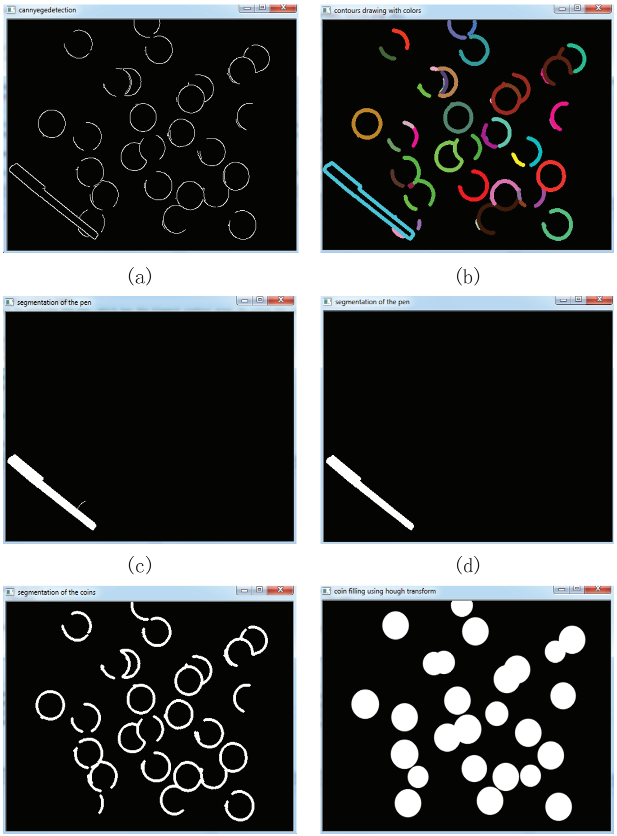


(a) (b)

Fig. 4. The image used in the experiment: (a) the original RGB color image; (b) the grayscale image.

After that canny edge detector is applied on the grayscale image to compute the edges of sharp intensity (see Fig. 5(a)). It can be easily seen that with the edges of circles there is some noise coming from background as the background is not plain. The output of canny edge detection gives us the edges of the objects but because of some objects are overlapped with each other. It is difficult to discriminate which edge belong to which particular object and which edge is overlapping. To make it easy, contour detection is applied on the output of canny edge detector and each edge is drawn with a separate random color making it easy to see which edges are separate and which edges are merged together into a single edge (see Fig. 5(b)). In Fig. 5(b)), it is evident that the edges are merged with each other. At this step, even if circular Hough transform is applied only coins will be detected and counted because of their circular shape but pen will not be detected because it is not circular in shape. The purpose of the proposed approach is to use circular Hough transform in such a way to get the results in which all the desired objects are detached from each other and after that using a generic image processing technique e.g. contour detection to count objects irrespective of the of their shapes. So for the purpose of simplification the pen contour is separated and filled internally to be prominent, even though the results can be achieved otherwise (see Fig. 5(c)). From Fig. 5(c), we can see that the pen still has some noise as a part of the coin is attached to it, which can be removed by drawing the contour with a black boundary around it (see Fig. 5(d)). In the proposed method, no morphological approach is applied as used in[5]. Contours of the circles are drawn on a new image which is an 8-bits 3-channels image (see Fig. 5(e)). In this paper, Circular Hough transform is used not just only to detect the circular shaped objects, but also to draw the closed contour circles. After the contour detection step, a lot of circles have open and unconnected boundaries. So by applying circular hough transform, retrieved the circles having a closed contour and after that circles are filled to to be visually prominent, but still the objects are not totally detached (see Fig. 5(f)).

Circular Hough transform is again applied on the already



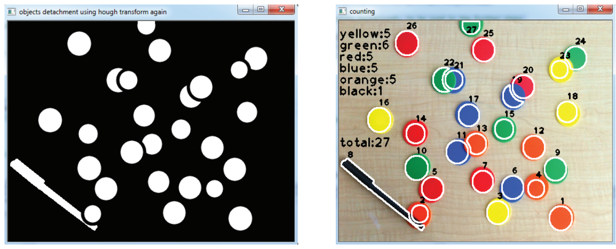
(a) (b) (c) (d) (e) (f)

Fig. 5. The image processing results in the experiment: (a) the Canny edge detection; (b) the contours drawn with random colors; (c) the contour of pen drawn and internally filled; (d) the noise removed by again drawing the contour; (e) drawn contours of the coins; (f) the circular Hough transform with filled circles.

filled circles while making the circle boundary pixels black for object separation. Output of the following step transforms overlapped objects into completely detach objects and it will treat one object as a full circle and other object will be the remaining part of the total overlapped area. After that, pen and coins are combined on the same image and it is quite evident that all the objects are detached (see Fig. 6(a)). According to the initial statement made, the purpose of our proposed approach is to detach all the objects. Objects can be circular or any random shaped objects attached or overlapped with the circular shaped objects. Now because all the objects are detached, they can be counted using a lot of techniques such as blob detection or contour detection. In this study, contour detection is applied to complete the task of object counting. The final results of the experiment is shown in Fig. 6(b).

IV. Discussion

To further prove the performance of the proposed approach, it is compared with the implementation in Matlab and a comparison experiment is carried out [16]. In this experiment, the original image is the same as that in Section III. The final results of the proposed approach is shown in



(a) (b)

Fig. 6. The final experiment results: (a) the completely detached objects; (b) the counting result of objects in the experiment.

Fig. 6(b) and the results obtained by Matlab is shown in Fig. 7. The average computational times of the proposed approach based on Opencv with C++ and of the method implemented in Matlab are 0.32s and 1.38s respectively. This comparison of the experimental results show that the proposed method can find total objects in the image accurately, but the method in Matlab just can find the circle objects (the pen in the image isn't found). Furthermore, the proposed method based on Opencv with C++ has outperformed Matlab in processing speed, being 5-6 times faster than Matlab. This performance of the proposed approach is very important for the computer vision applications in real time operations.

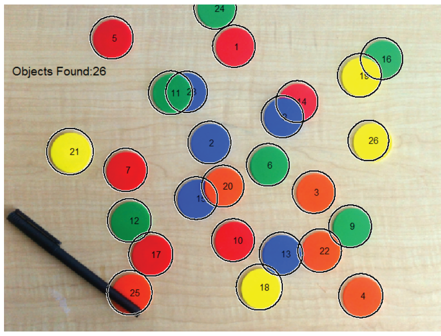


Fig. 7. The final experiment results based on Matlab.

V. Conclusion

In this paper, the counting task in the image is studied, where the image has circular shaped objects that are overlapping or very close to each other, including some random shaped overlapping objects. In the proposed approach, all the objects are detected, segmented, counted and sorted efficiently. Our proposed approach successfully segmented all the circular shaped objects e.g. coins and random shaped object e.g. pen attached to the coin. In our proposed approach, automatic segmentation of different shaped objects is done. The experiment results of the proposed approach show the efficiency of the proposed approach.

The proposed method can be used to solve the problem of detachment of random shaped cells attached with the circular shaped cells. As we know, in industries, for the estimation of goods infrared sensors based systems are used. They only detect the objects but have no color information, thus can't keep the record of each object separately. Using the proposed system objects can be quantized and categorized into different colors. The techniques applied in the system can also be applied to the cell segmentation and counting in the human blood, using the captured image of blood smears.

REFERENCES

[1] M. Sonka, V. Hlavac, and R. Boyle, *Image processing, analysis, and machine vision*. Cengage Learning, 2014.

[2] J. G. A. Barbedo, "A review on methods for automatic counting of objects in digital images," *IEEE Latin America Transactions*, vol. 10, no. 5, pp. 2112 – 2124, 2012.

[3] T. Liebig, "Privacy preserving centralized counting of moving objects," in *Lecture Notes in Geoinformation and Cartography*, vol. 217, Lisbon, Portugal, 2015, pp. 91 – 103.

[4] C. Arteta, V. Lempitsky, J. A. Noble, and A. Zisserman, "Learning to detect partially overlapping instances," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, 2013, pp. 3230–3237.

[5] N. Abbas, D. Mohamad, and A. H. Abdullah, "Semi-automatic red blood cells counting in microscopic digital images," *Jurnal Teknologi*, vol. 73, no. 2, 2015.

[6] M. Rizon, H. Yazid, and P. Saad, "A comparison of circular object detection using hough transform and chord intersection," in *Geometric Modelling and Imaging, GMAI 2007, Proceedings*, Zurich, Switzerland, 2007, pp. 115 – 120.

[7] T. D'Orazio, C. Guaragnella, M. Leo, and A. Distante, "A new algorithm for ball recognition using circle hough transform and neural classifier," *Pattern Recognition*, vol. 37, no. 3, pp. 393 – 408, 2004.

[8] J. Canny, "A computational approach to edge detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, no. 6, pp. 679–698, 1986.

[9] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 5, pp. 898–916, 2011.

[10] R. Maini and D. H. Aggarwal, "Study and comparison of various image edge detection techniques," *International Journal of Image Processing*, vol. 3, no. 1, pp. 1–11, 2009.

[11] O. Vincent and O. Folorunso, "A descriptive algorithm for sobel image edge detection," in *Proceedings of Informing Science & IT Education Conference (InSITE)*, vol. 40, 2009, pp. 97–107.

[12] R. O. Duda and P. E. Hart, "Use of the hough transformation to detect lines and curves in pictures," *Communications of the ACM*, vol. 15, no. 1, pp. 11–15, 1972.

[13] P. Mukhopadhyay and B. B. Chaudhuri, "A survey of hough transform," *Pattern Recognition*, vol. 48, no. 3, pp. 993 – 1010, 2015.

[14] S. Bindu, S. Prudhvi, G. Hemalatha, M. N. R. Sekhar, and M. V. Nanchariahl, "Object detection from complex background image using circular hough transform," *Int. Journal of Engineering Research and Applications*, vol. 4, pp. 23–28, 2014.

[15] S. Brahmbhatt, *Practical OpenCV*. Berkely, USA: Apress, 2013.

[16] S. Matuska, R. Hudec, and M. Benco, "The comparison of CPU time consumption for image processing algorithm in matlab and opencv," in *Proceedings of 9th International Conference, ELEKTRO 2012*, Zilina - Rajecke Teplice, Slovakia, 2012, pp. 75 – 78.