Hough Transform for line & circle detection

Amir M. Mousavi

Department of Computer Engineering
Shahid Rajee University

Tehran, Iran

AmirMahmood.Mousavi@yahoo.com

Abstract **- Hough transform is a feature extraction algorithm widely used in digital image analysis. It identifies objects with specified shapes by letting all edge pixels in the image "vote" and extract candidates with the highest votes. The classical Hough transform is associated with the identification of straight lines, but with some modification, the algorithm can be used to detect objects of arbitrary shapes (usually circles, ellipses) in an image. Therefore, Hough transform has various applications in the field of computer vision. in this paper we implemented Hough transform to detect straight lines and circles in real world images**

This process is used in many industrial computer vision applications such as recognition of objects, segmentation of images, and finding defects. This article presents an introduction to Hough transform and we will, discuss its advantages and disadvantages and general comparison , then we will show the MATLAB implementation and discuss the results .

Keywords— Hough transform , Computer Vision, , Image Processing, HT , line detection , circle detection ,Grayscale , vote .

.

I. Introduction

The Hough transform (HT), one of the oldest algorithms of computer vision [1, 2], has been attracting a continuous interest for more than 50 years, as testified by a number of surveys [3–5], the most recent one [6], mentioning more than 2500 research papers, and citing around 100post-2000 references. This is due not only to the constant development of applications based on graphics recognition but also to the elegance and generality of the Hough framework, allowing to detect analytical [7–10] as well as non-analytical [11–13] shapes in a wide variety of relevant manners..

The basic principle of the HT is to project the image data within a parameter space representing possible positions of a shape in the image space, then to search accumulation points in the parameter space, corresponding to the most probable positions of the shape in the image. Although many variations have been proposed until recently, it is remarkable that in most cases, the projection (voting process) is sparsely performed on contour or salient points of the image space. In addition, it is almost always performed using one of the two dual methods: one-to-many (one point in the image space votes for an multidimensional surface in the parameter space) or many-to-one (a set of points in the image space votes for one single point in the parameter space). Although dense HTs have been proposed already, and one-to-one voting has also been used by some authors, to our knowledge, the two concepts have not been used together..

In this paper we have proposed to provide a comprehensive tutorial about review of the Hough transform (HT) which is discussed in section II . Then a brief explanation of line and circle detection are in sections III and IV . Finally. The results of the implementation are presented V and the conclusions are discussed in section VI .

II. Hough transform

The Hough transform is a technique which can be used to isolate features of a particular shape within an image. Because it requires that the desired features be specified in some parametric form, the classical Hough transform is most commonly used for the detection of regular curves such as lines, circles, ellipses, etc. A generalized Hough transform can be employed in applications where a simple analytic description of a feature(s) is not possible. Due to the computational complexity of the generalized Hough algorithm, we restrict the main focus of this discussion to the classical Hough transform. Despite its domain restrictions, the classical Hough transform (hereafter referred to without the classical prefix) retains many applications, as most manufactured parts (and many anatomical parts investigated in medical imagery) contain feature boundaries which can be described by regular curves. The main advantage of the Hough transform technique is that it is tolerant of gaps in feature boundary descriptions and is relatively unaffected by image noise.

As mentioned before The Hough transform is a feature extraction technique used in image analysis, computer vision, and digital image processing.[1] The purpose of the technique is to find imperfect instances of objects within a certain class of shapes by a voting procedure. This voting procedure is carried out in a parameter space, from which object candidates are obtained as local maxima in a so-called accumulator space that is explicitly constructed by the algorithm for computing the Hough transform.

The classical Hough transform was concerned with the identification of lines in the image, but later the Hough transform has been extended to identifying positions of arbitrary shapes, most commonly circles or ellipses. The Hough transform as it is universally used today was invented by Richard Duda and Peter Hart in 1972, who called it a "generalized Hough transform"[2] after the related 1962 patent of Paul Hough.[3][4] The transform was popularized in the computer vision community by Dana H. Ballard through a 1981 journal article titled "Generalizing the Hough transform to detect arbitrary shapes".

III. HOUGH TRANSFORM FOR LINE AND CIRCLE DECTETION

The Hough technique is particularly useful for computing a global description of a feature(s) (where the number of solution classes need not be known a priori), given (possibly noisy) local measurements. The motivating idea behind the Hough technique for line detection is that each input measurement (e.g. coordinate point) indicates its contribution to a globally consistent solution (e.g. the physical line which gave rise to that image point).

As a simple example, consider the common problem of fitting a set of line segments to a set of discrete image points (e.g. pixel locations output from an edge detector). Figure 1 shows some possible solutions to this problem. Here the lack of a priori knowledge about the number of desired line segments (and the ambiguity about what constitutes a line segment) render this problem under-constrained.

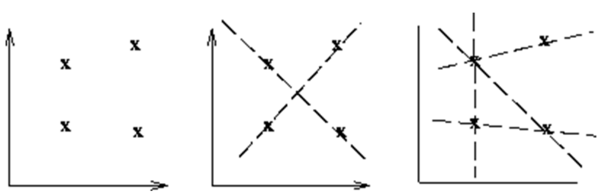

Figure 1 a) Coordinate points. b) and c) Possible straight line fittings.

We can analytically describe a line segment in a number of forms. However, a convenient equation for describing a set of lines uses parametric or normal notion:

$$x \cos\theta + y \sin\theta = r$$

where r is the length of a normal from the origin to this line and θ is the orientation of r with respect to the X-axis. (See Figure 2.) For any point (x,y) on this line, r and θ are constant.

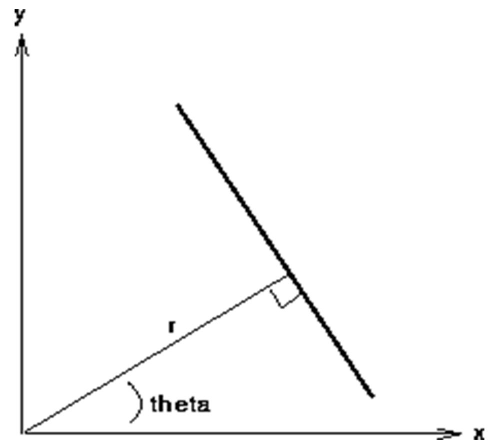

Figure 2 Parametric description of a straight line.

In an image analysis context, the coordinates of the point(s) of edge segments (i.e. (x,y)) in the image are known and therefore serve as constants in the parametric line equation, while r and θ are the unknown variables we seek. If we plot the possible θ, r values defined by each (x,y), points in cartesian image space map to curves (i.e. sinusoids) in the polar Hough parameter space. This point-to-curve transformation is the Hough transformation for straight lines. When viewed in Hough parameter space, points which are collinear in the cartesian image space become readily apparent as they yield curves which intersect at a common (r,θ) point.

The transform is implemented by quantizing the Hough parameter space into finite intervals or accumulator cells. As the algorithm runs, each (x,y) is transformed into a discretized θ, r curve and the accumulator cells which lie along this curve are incremented. Resulting peaks in the accumulator array represent strong evidence that a corresponding straight line exists in the image.

We can use this same procedure to detect other features with analytical descriptions. For instance, in the case of circles, the parametric equation is :

$$(x-a)^2 + (y-b)^2 = r^2$$

where a and b are the coordinates of the center of the circle and r is the radius. In this case, the computational complexity of the algorithm begins to increase as we now have three coordinates in the parameter space and a 3-D accumulator. (In general, the computation and the size of the accumulator array increase polynomially with the number of parameters. Thus, the basic Hough technique described here is only practical for simple curves.)

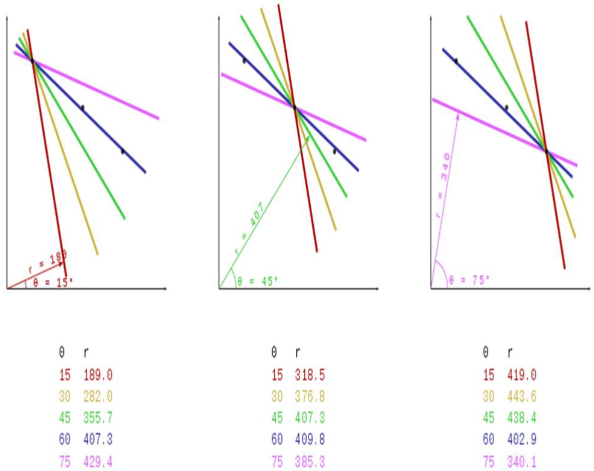

Figure.3. an example of line detection

IV. EXPERIMENTAL RESULTS

In this section we will represnt the result of implementation of (HT) for line and circle detection in MATLAB R2014a, and tested with the following images (Fig.4) . The objective is to produce an image with the strongest lines and circles.


Fig.4

❖ Line :

We have implemented the hough for line detection in 2 ways :

١. Using MATLAB's function (houghlines)
٢. Writing the algorithm of hough (first we applied a smooth filter then used the canny to get the edges then we used a peak to select the top lines with the most votes then draw the lines)

The results are in fig 5- 13:

- MATLAB's function:

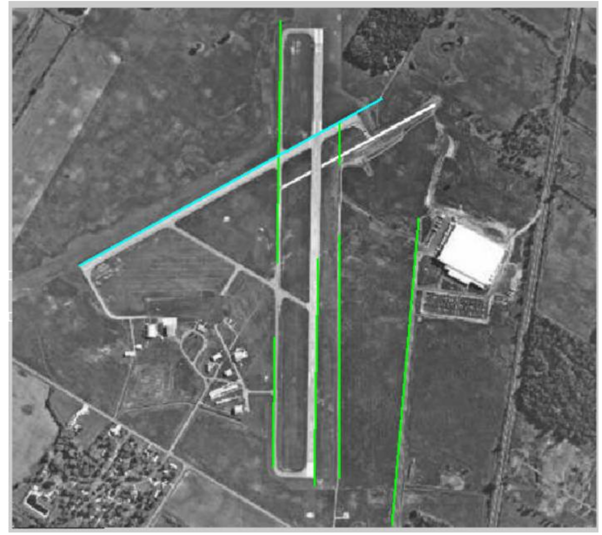
Fig.5. Using MATLAB's function and finding top 5, and the top of the votes is shown in cyan color

✓ **Conclusion :** we can observe that some of the lines are not fully connected , we guess that the source is the canny , because of the edges these lines are not verified as a full line and it's been apart . but on the other hand the lines are in a fit size and corrected guessed .

- Writing the algorithm of hough :

Fig.6. smooth filter (guassian)

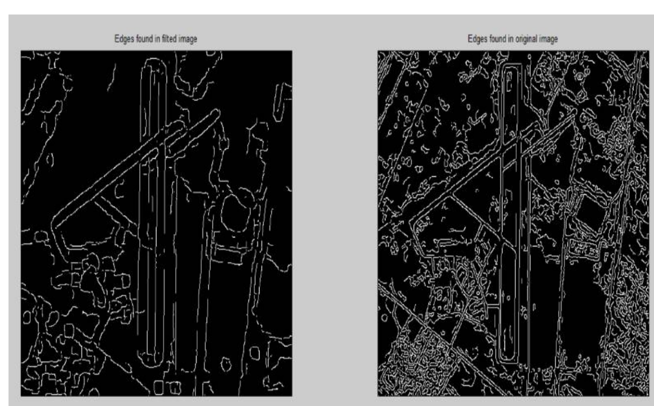

Fig.7. edge detection (canny)

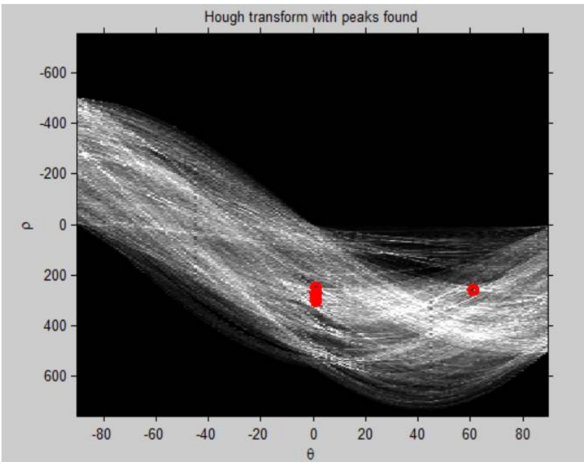

Fig.8. hough transform with 5 peaks

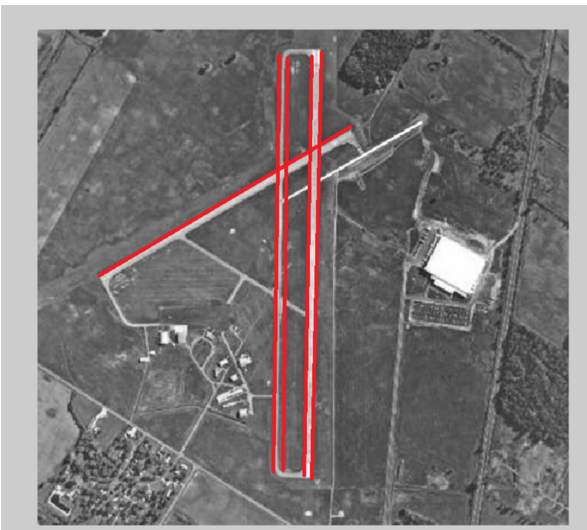

Fig.9. final image with 5 lines

- ✓ **Conclusion :** we can observe that edge detection is an important part that can give us the better lines , we applied Gaussian filter to smooth the image then we used canny for edge detection .

❖ Circle :

We have implemented the hough for circle detection by Writing the algorithm of hough (first we applied a smooth filter then used the canny to get the edges then we used a peak to select the top circles with the most votes then draw the circles)

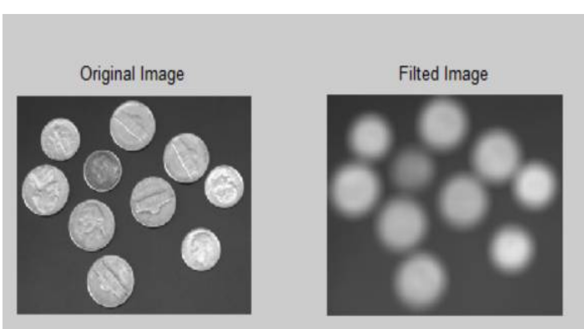

Fig.10. smooth filter (guassian)

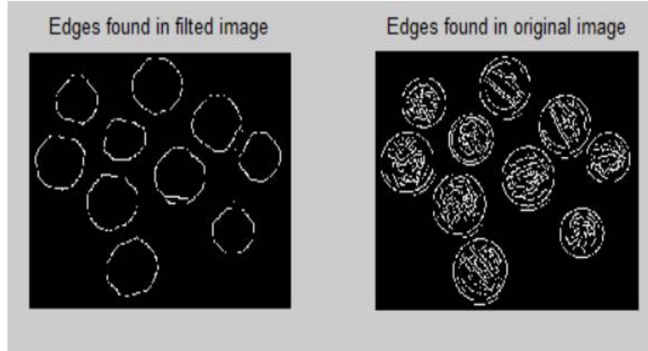

Fig.11. edge detection (canny)

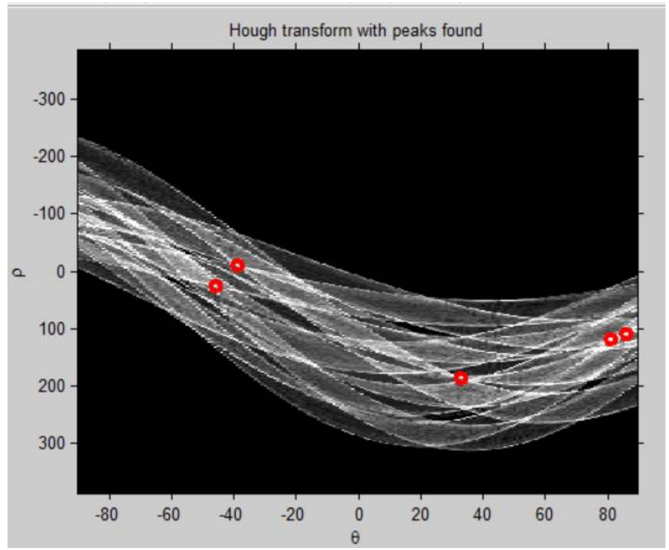

Fig.12. hough transform with 5 peaks

Fig.13. final image with 5 circles.

- ✓ **Conclusion :** we can observe that edge detection is an important part that can give us the better circles , we applied Gaussian filter to smooth the image then we used canny for edge detection .

V. CONCLUSION

In the discipline of computer vision, image processing is a quickly moving field. Its growth has been fueled by technological advances in digital imaging, computer processors and mass storage devices. In this paper an attempt is made to review the hough transform (HT) which are based on voting . The relative performance of HT technique is carried out with two images for circle and line detection by using MATLAB software. It is observed from the that edge detection and tresholding techniques are an important part that can give us the better lines and circles , we applied Gaussian filter to smooth the image then we used canny for edge detection. Even though, so many edge detection techniques are available in the literature, since it is a challenging task to the research communities to detect the exact image without noise from the original image.

REFERENCES

[1] Canny, J. F (1983) Finding edges and lines in images, Master's thesis, MIT. AI Lab. TR-720..

[2] Gonzalez, R., Woods, R., & Eddins, L. (2009). Digital image processing using MATLAB. New Delhi: TATA McGraw-Hill Education.

[3] Canny, J. F (1986) “A computational approach to edge detection”, IEEE Transaction on Pattern Analysis and Machine Intelligence, 8, 679-714..

[4] Hanzi Wang (2004) Robust Statistics for Computer Vision: Model Fitting, Image Segmentation and Visual Motion Analysis, Ph.D thesis, Monash University, Australia.

[5] Wikipedia.com

" EDGE DETECTION TECHNIQUES FOR IMAGE SEGMENTATION “ Muthukrishnan.R1 and M.Radha