Line Detection Using Radon Transform

Dr. H. H. Abbas* & Hussein Ali Hussein**

Received on: 3/8/2008
Accepted on: 5/3/2009

Abstract

To extract features from digital images, it is useful to be able to find simple shapes (straight lines, circles, ellipses, etc.).

In order to achieve this goal, one can be able to detect a group of pixels that are on a straight line or a smooth curve. That is what a Hough transform is supposed to do.

Since the Hough transform is a special case of Radon transform then line detection process is accomplished using Radon transform.

In this paper, software for line detection using Radon transform has been designed and implemented. Then the implemented software is tested under many conditions and circumstances, the results are discussed, and many points are concluded from the results.

Keywords: Curve detection, Image processing, Hough transform, Radon transform.

كشف الخطوط باستخدام خوارزمية (Radon)

الخلاصة

لغرض استخلاص خواص الصور الرقمية ، من المفيد أن نكون قادرين على كشف الأشكال البسيطة (الخطوط المستقيمة ، الدوائر ، الأشكال البيضاوية ، الخ.....).

ولإنجاز هذا الهدف، يجب أن نكون قادرين على كشف مجموعة من النقاط (Pixels) الواقعة فوق خط مستقيم أو منحني وهذا العمل من المفروض أن ينجزه (Hough Transform).

بما أن (Hough Transform) هو حالة خاصة من (Radon Transform) فأن عملية كشف الخطوط قد تمت في هذا البحث باستخدام (Radon Transform).

في هذا البحث تم تصميم خوارزمية و من ثم بناء برنامج حاسوبي لغرض كشف الخطوط باستخدام (Radon Transform). و بعد ذلك فأن البرنامج قد تم اختباره في حالات و ظروف مختلفة و من ثم فأن النتائج قد نوقشت و تم استنتاج عدة نقاط من النتائج.

Introduction

Hough transform is a feature extraction technique used in digital image processing. The classical transform identifies lines in the image, but it has been extended to identifying positions of arbitrary shapes. The transform universally used today was invented by Richard Duda and Peter Hart in 1972 [1], who called it a "generalized Hough transform" after the related 1962 patent of Paul Hough [2].

* Mansour University College / Baghdad
** College of Engineering , University of Al-Mustansirya / Baghdad

https://doi.org/10.30684/etj.28.6.18
2412-0758/University of Technology-Iraq, Baghdad, Iraq

Theory

At start, the case of straight lines in an image is considered. It is clear that for every point (x_i,y_i) in that image, all the straight lines passing through that point satisfy Eq. (1) for varying values of line slope (m) and intercept (c). (see Figure (1)).

$$y_i = mx_i + c \qquad (1)$$

Now if we reverse our variables and look instead at the values of (m,c) as a function of the image point coordinates (x_i,y_i), then Eq. (1) becomes Eq. (2) which describes a straight line [3, 4]. (see Figure (2)).

$$c = y_i - mx_i \qquad (2)$$

Let us consider two pixels P1 and P2, which lie on the same line in the (x,y) space. For each pixel, one can represent all the possible lines through it by a single line in the (m,c) space. Thus a line in the (x,y) space that passes through both pixels must lie on the intersection of the two lines in the (m,c) space representing the two pixels. This means that all pixels which lie on the same line in the (x,y) space are represented by lines which all pass through a single point in the (m,c) space (see Figure (3)).

One can now describe the original algorithm for detecting lines in images:

1. Find all edge points in the image using any suitable edge detection scheme.
2. Quantize the (m,c) space into a two-dimensional matrix H with appropriate quantization levels.
3. Initialize the matrix H to zero.
4. Increment by 1 each element of H(m_i,c_i) which corresponds to an edge point.
5. The result is a histogram or a 'vote matrix' showing the frequency of edge points corresponding to certain (m,c) values (i.e. points lying on a common line).
6. H is threshold such as only the large valued elements are taken. These elements correspond to "strong" lines in the original image [5, 6].

Practical Issues

To avoid the problem of infinite m values which occurs when vertical lines exist in the image, the alternative formulation shown in Eq. (3) can be used to describe a line.

$$x \cos\theta + y \sin\theta = r \qquad (3)$$

The parameter r represents the distance between the line and the origin, while θ is the angle of the vector from the origin to the closest point on the line (see Figure (4)).

It is well known that an infinite number of lines can go through a single point of the plane. If that point has coordinates (x_0, y_0) in the image plane, all the lines that go through it obey the following equation:

$$r(\theta) = x_0 \cdot \cos\theta + y_0 \cdot \sin\theta \; .....(4)$$

This corresponds to a sinusoidal curve in the (r,θ) plane, which is unique to that point. If the curves corresponding to two points are superimposed, the locations where they cross correspond to lines (in the original image space) that pass through both points. More generally, a set of points that form a straight line will produce sinusoids which cross at the parameters for that line [7, 8].

Hough Transform Example

Given three data points, shown in Figure (5) as black dots.

Then the following steps are applied:

1. For each data point, a number of

lines are plotted going through it, all at different angles. These are shown in Figure (5) as solid lines.

2. For each solid line a line is plotted which is perpendicular to it and which intersects the origin. These are shown as dashed lines.

3. The length and angle of each dashed line is measured. In the figure above, the results are shown in tables.

4. This is repeated for each data point

5. A graph of length against angle is then created.

The point where the lines intersect gives a distance and angle. This distance and angle indicate the line which bisects the points being tested [5-8].

Advantages and Disadvantages

The advantage of Hough transform is that pixels lying on one line need not all be contiguous. This can be very useful when trying to detect lines with short breaks in them due to noise.

The disadvantages of Hough transform [3-8]:

1. It can give misleading results when object happen to be aligned by chance.

2. Detected lines are infinite lines described by their (m,c) values, rather than finite lines with defined end points.

The relation between Hough and Radon Transforms

The basic idea behind using Hough transform to identify lines in digital images is quite straightforward and follows directly from the definition as given below:

$$H_g(p,\theta) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x,y)\delta(p - x\cos\theta - y\sin\theta)dxdy \quad \text{…..} (5)$$

Thus Hough transform of a function concentrated at a point $\delta(x-x_0)\delta(y-y_0)$ yields a sinusoidal curve given by:

$$p = x_0 \cos\theta + y_0 \sin\theta \quad \text{….}(6)$$

in the θρ-plane.

The line integral of the function f along the line L is given by:

$$R_f(\rho,\theta)=\int_{L(\theta,\rho)} f(x,y)\, ds \quad \text{……}(7)$$

where ds is an increment of length along L.

If $R_f(\rho,\theta)$ is known for all ρ and θ, then $R_f(\rho,\theta)$ is the two-dimensional Radon transform of f(x,y). Using a delta function, Eq. (7) can be rewritten as:

$$R_f(\rho,\theta) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(\rho - x\cos\theta - y\sin\theta)dxdy \quad \text{…..}(8)$$

Substituting $f(x,y) = \delta(x-a)\times\delta(y-b)$ into Eq. (8) we have

$$R_f(p,\theta)=\delta(p-p_0) \quad \text{…..} (9)$$

Where

$$p_0 = a\cos\theta + b\sin\theta \quad \text{....}(10)$$

Equation (9) actually expression what Hough meant [9, 10].

Software design and implementation

The software for line detection using Radon transform was implemented according to the following steps:

1. The user must input the threshold value used for detecting lines.

2. The user must input the input image which must be black and white or gray mode. If the input image is a RGB then it must be converted to gray level image.

3. Then the edges in the image will be detected and the result from this step is a binary file.

4. The next step is to compute the accumulator matrix using Radon transform (this matrix is similar to the tables shown in Figure (5)). The accumulator matrix is a two dimensional matrix and from this matrix the parameters r and θ for the

detected lines will be computed.

5. Then the maximum value in the accumulator matrix is computed. This value is a candidate for parameters of a line (r and θ). The threshold value (the threshold = a× the maximum where a is from 0 to 1) is used in this step to increase sensitivity (ability to detect lines) of line detection process.

6. Then for each candidate (r and θ) the coordinates of the detected line will be computed according to the following steps:

a. At start, one can use the values of r and θ to calculate a point in the space (origin point) according to the following equations: image

$$xorigin = r \cos\theta \quad \ldots (11)$$

$$yorigin = r \sin\theta \quad \ldots .(12)$$

b. From step (a), the end points for the detected line is computed according to the following equations

$$ystart = yorigin + (xstart-xorigin) \tan (\theta\text{-}90) \quad \ldots . .(13)$$

$$yend = yorigin + (xend-xorigin) \tan (\theta\text{-}90) \quad \ldots\ldots(14)$$

The values of xstart and xend are selected according to the display coordinates.

c. The final step is to plot the detected line on the same graph with the input image.

Results and discussion

After implementing the software according to the steps given in the previous paragraph (the Matlab package was used in the implementation process), it has been tested under various conditions (lines with different slopes) and for different images. The test results are then discussed to clarify various factors that affect line detection process.

Case studies

a. The first Case study (lines with different slopes but not horizontal or vertical).

The results related to the first case study in this paper are presented in Figure (7). The image used in this case consists of lines with different slopes (but not horizontal or vertical). Different threshold values have been used and from these values many points can be concluded.

b. The second case study (horizontal and vertical lines).

The results related to the second case study in this paper are presented in Figure (8). The image used in this case consists of vertical and horizontal lines. Different threshold values have been used and from these values many points can be concluded.

Discussion of the Results

From previous subsection, one can conclude many points from the two cases considered before:

1. In general, increasing the threshold increases the sensitivity of detecting lines process (only strong lines will be detected (one can see the number of blue lines in Figure (7) and Figure (8))).

2. In general, decreasing the threshold decreases the sensitivity of detecting lines process (weak and strong lines will be detected (one can see the number of blue lines in Figure (7) and Figure (8))).

3. For low threshold values each line is detected multiple times.

4. The optimum threshold values is (0.6 - 0.7).

References

[1]Duda, R. O. and Hart, P. E., Jan 1972,"Use Hough transformation to detect lines and curves in pictures", Commun. Ass. Comput. Mach., vol. 15.

[2]Hough, P. V. C., Dec. 18, 1962,"Method and means for recognizing complex patterns", U. S. 2 069 654.

[3] Image Analysis Group, Chalmers University of Technology, 2000, "The Hough Transform".

[4]Mores, Bryan S., 2000, "Segmentation (Edge Based, Hough Transform)", Brigham Young University.

[5]Toft, Peter. The Radon Transform: Theory and Implementation. Denmark: Technical University; 1996. Ph.D. Thesis.

[6] Machine Vision–Ramesh Jain, Rangachar Kasturi, Brian G Schunck, McGraw-Hill, 1995.

[7] Machine Vision-Wesley E. Snyder, Hairong Qi, Cambridge University Press, 2004.

[8] Adel Faridani, Tomography and Sampling Theory, The Radon Transform and Applications to Inverse Problems (Providence, RI, USA), AMS Proceedings of Symposia in Applied Mathematics, American Mathematical Society, 2006.

[9] Nguyen Dinh Thus and Duong Anh Duc, 2004, "The Hough Transform-A Radon-Like Transform ", Dept of Information technology, University of Natural Sciences.

[10] Van Ginkel, M. et al, 2004, "A short introduction to the Radon and Hough transforms and how the relate to each other", Quantitative Imaging Group, Imaging Science & Technology Department, Faculty of Applied Science, Delft University.

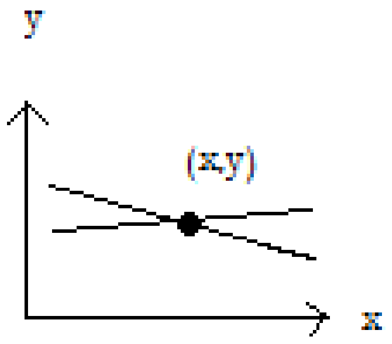


Figure (1) Lines through a point in the Cartesian domain.

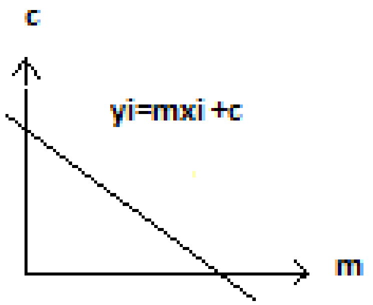
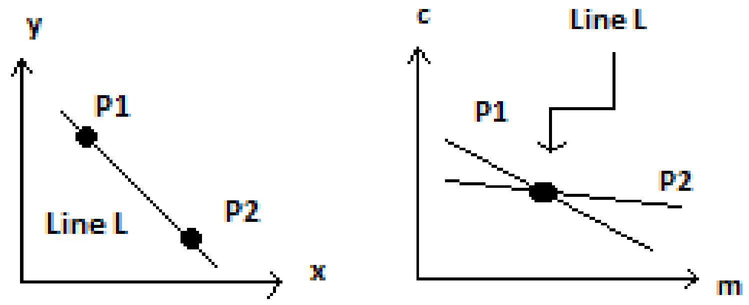


Figure (2) The (m, c) domain.



(a)

(b)

Figure (3) (a) Points on the same line, and
(b) The mapping of P1 and P2 from Cartesian space to the (m,c) space.

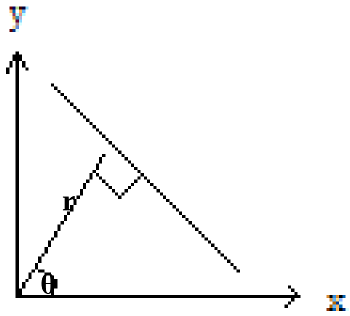
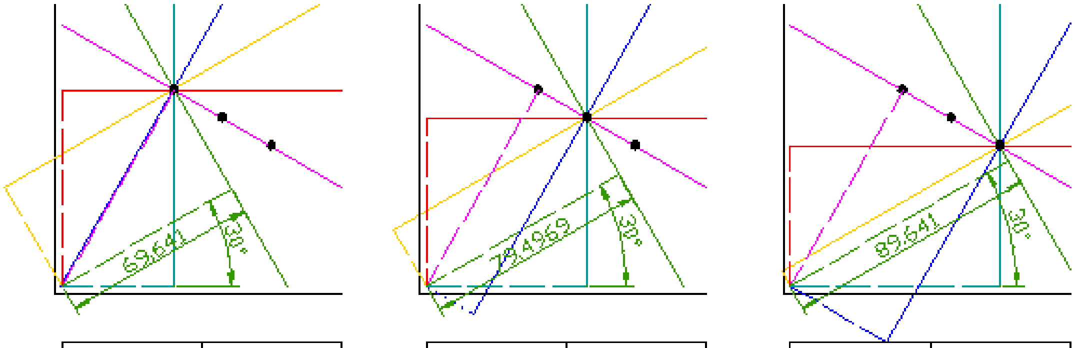


Figure (4) The representation of a line in the (x,y) space using (r_i,q_i).



Angle	Dist.
0	40
30	69.6
60	81.2
90	70
120	40.6
150	0.4

Angle	Dist.
0	57.1
30	79.5
60	80.5
90	60
120	23.4
150	-19.5

Angle	Dist.
0	74.6
30	89.6
60	80.6
90	50
120	6.0
150	-39.6

Figure (5) Example of using Hough Transform.

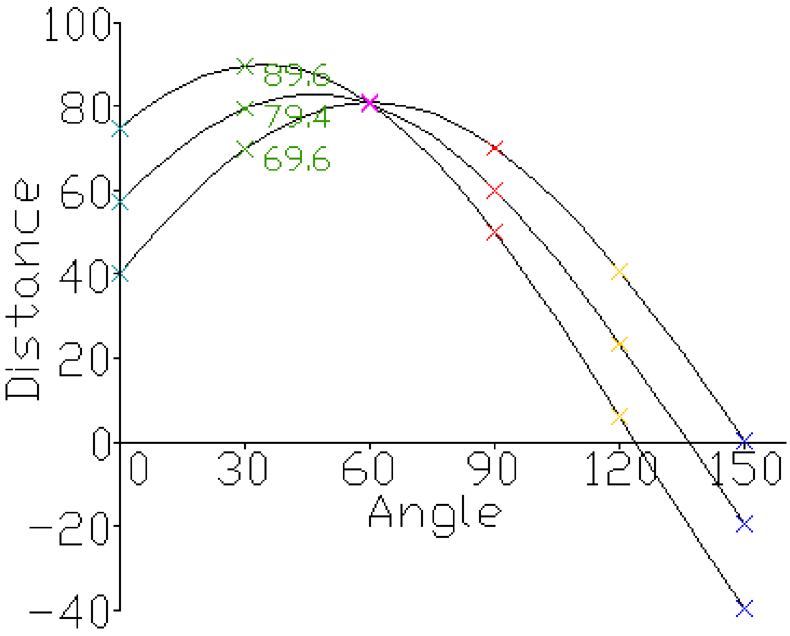
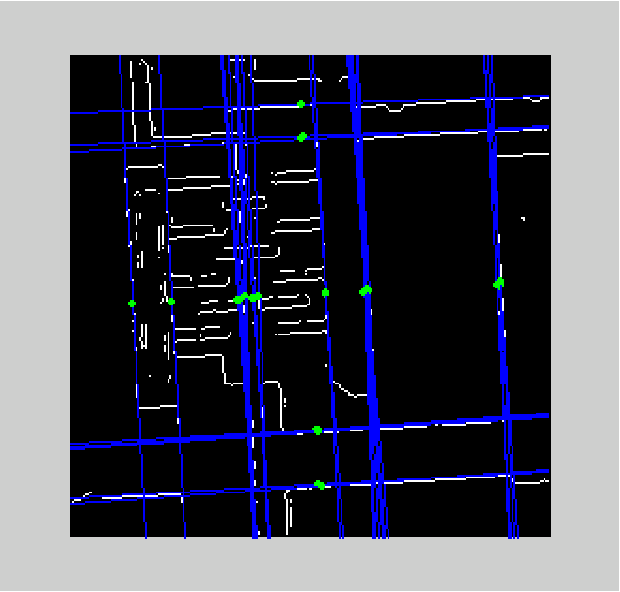
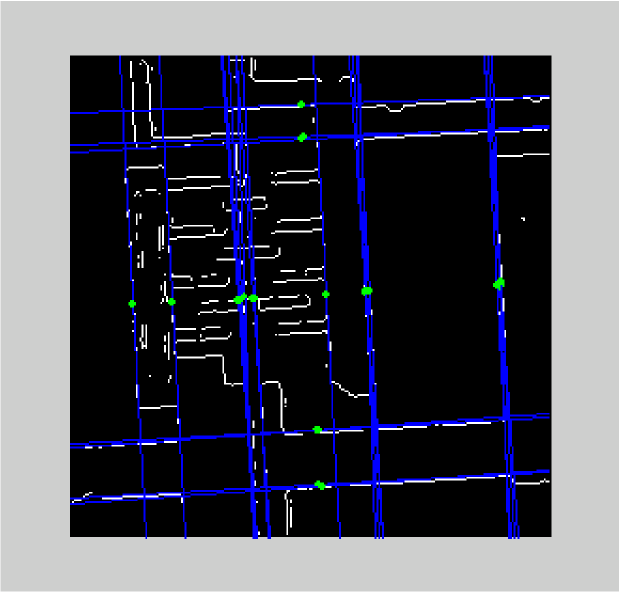


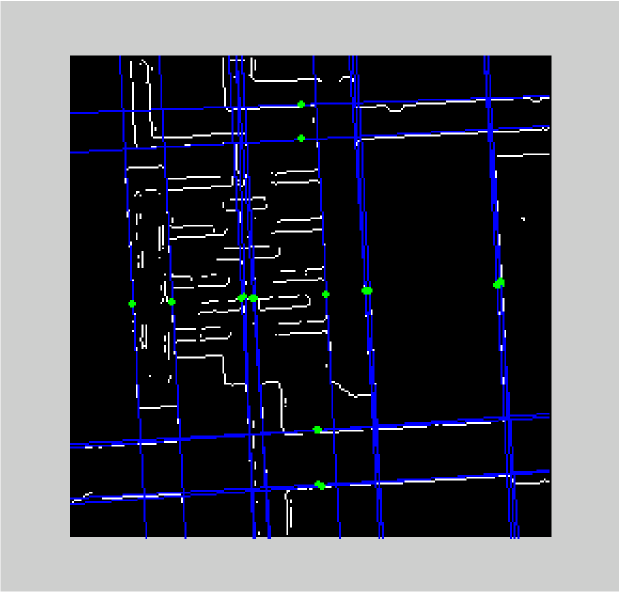
Figure (6) A graph of length against angle for our example (see Figure (5)).



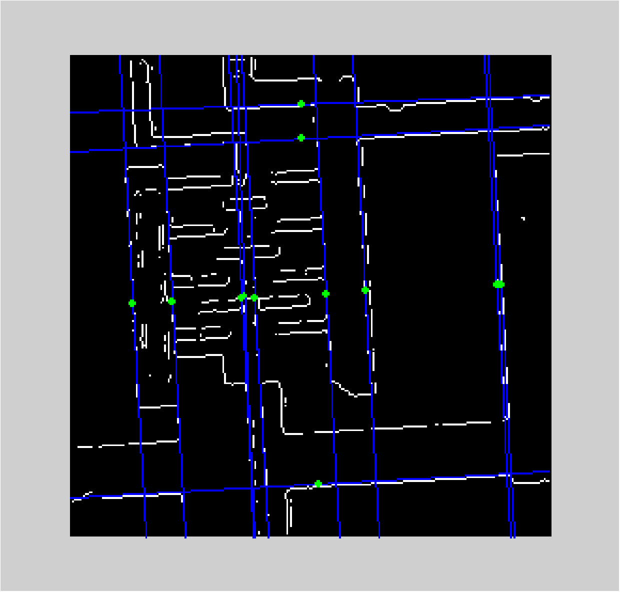
(a) **Threshold=0.5**



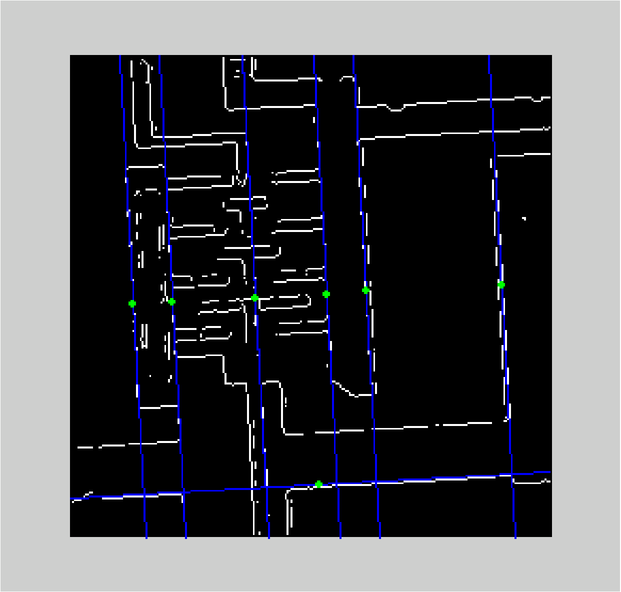
(b) **Threshold=0.55**



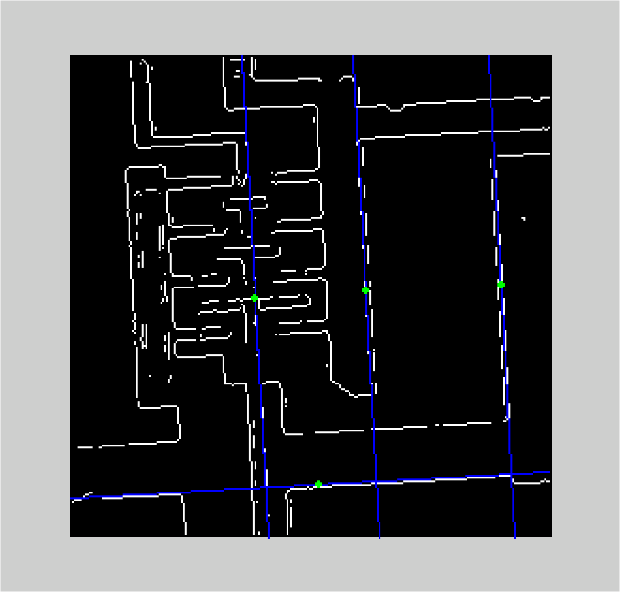
(c) **Threshold=0.6**



(d) **Threshold=0.65**

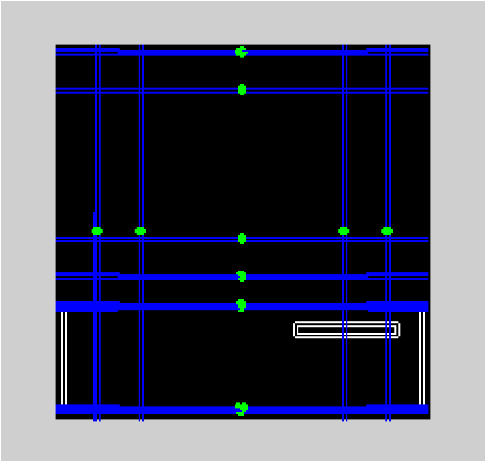


(e) Threshold=0.7

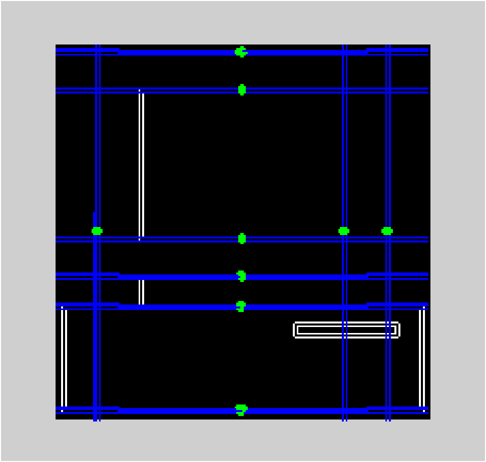


(f) Threshold=0.75

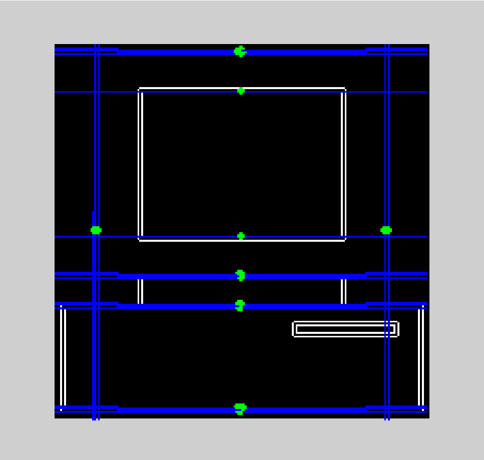
Figure (7) The first case study results for different threshold values.



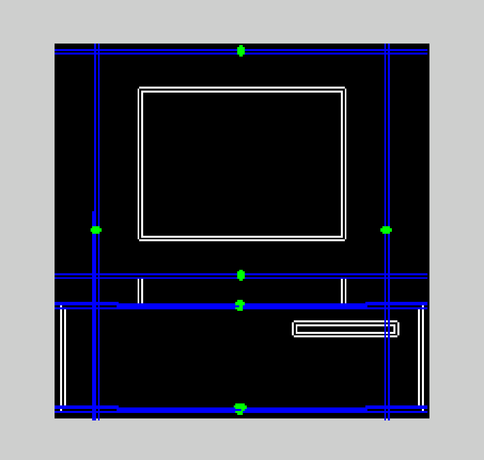
(a) **Threshold=0.5**



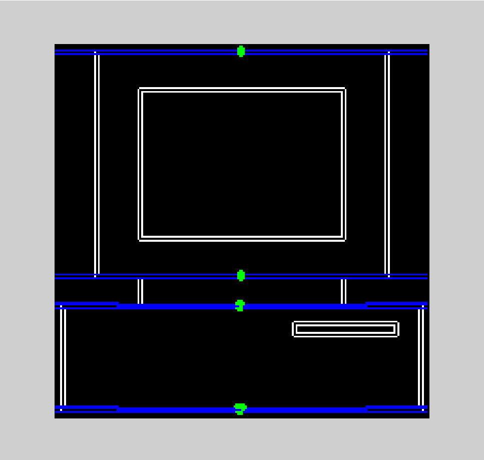
(b) **Threshold=0.55**



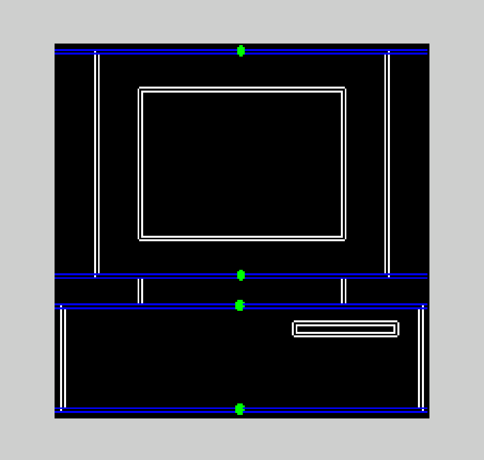
(c) **Threshold=0.6**



(d) **Threshold=0.65**



(e) **Threshold=0.7**



(f) **Threshold=0.75**

Figure (8) The second case study results for different threshold values.