Automatic Reading and Interpretation of an Antibiogram

Ahmad Diab[1,2], Abdelwahab Raad[1], Mashhour Chakouch[1,2], Marwan Osman[1], Monzer Hamze[1], Mohamad Khalil[1], and Walid Kamali[2]
[1]Faculty of Engineering, Faculty of Public Health, Azm center for research in Biotechnology and its Applications
Lebanese University, Lebanon
[2]Department of Biomedical Engineering, Faculty of Engineering and Maritime Studies
Al Manar University of Tripoli, Tripoli, Lebanon

Nowadays, the reading and interpretation of antibiogram tests is a frequently performed task by doctors, researchers and technicians at hospitals and laboratories. An antibiogram is a test of the sensitivity of a microorganism to given antibiotics. Reading an interpretation of the antibiogram results is usually performed manually, which leads to human errors and a great waste of time. There are few automated devices that reads and interprets antibiograms but they are expensive and costly to run. This paper aimed to present a prototype that reads and interprets antibiograms. In this paper, we show the algorithm of detection and interpretation using the recent image processing techniques. The two principle techniques used for circles detection are the Circular Hough Transform (CHT) and the Pixel Value Checking (PVC).

Keywords: Antibiogram, Image Processing, Circular Hough Transform, Pixel Value Checking, Inhibition Zone, GUI.

I. INTRODUCTION

The goal of microbiology is to diagnose the pathological infections in order to assign the best treatment for them [1]. An Antibiogram is a laboratory technique that studies the susceptibility of microorganisms to given antibiotics. Generally, an antibiogram test is recommended when an infection is severe or when the bacteria responsible of the infection are resistant to many antibiotics[2].

There is two widely used techniques to do an antibiogram: the dilution method and the diffusion method [2]. We focused on the diffusion method, and we try to provide a low-cost device that reads and interprets antibiograms. In diffusion method, antibiotic wafers are placed on an agar plate where bacteria have been placed and then the plate is left to incubate. If an antibiotic stops the growth of bacteria or kills it, there will be a circular area around the wafer where bacteria haven't grown. This area is called the inhibition zone. The diameter of the zone of inhibition indicates the degree of sensitivity of bacteria to a drug. In general, a bigger area of bacteria-free media surrounding an antibiotic disk means the bacteria are more sensitive to the drug the disk contains. These diameters are usually measured manually with a ruler. Then bacteria are classified into three clinical categories: resistant, intermediate and susceptible[2]. The aim of this work is to design a low-cost prototype that automatically reads and interprets an antibiogram. The concept of the prototype is to capture an image of the antibiogram plate and then process it via our developed software.

II. MATERIALS AND METHODS

A. Prototype design

We made a rectangular wooden box equipped with a LED dimmer strip to adjust the lightening inside. Wood is used to avoid light reflection inside the box. The box was also equipped with a drawer, a 1080p webcam and movable stands for the camera and the plate inside the box.

B. Procedure

The block diagram below shows the steps followed in our methodology:

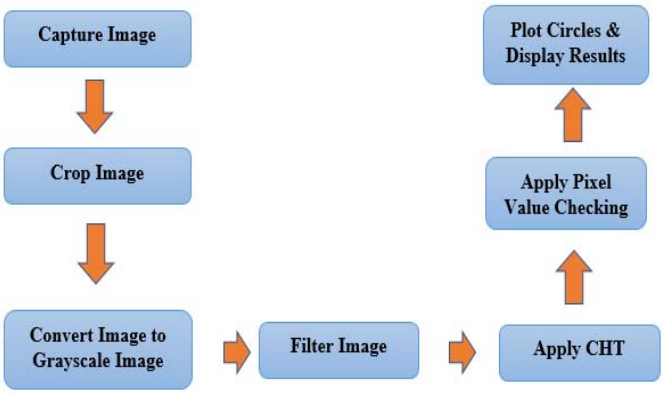


Figure 1: Block Diagram of the Work

First of all, the antibiogram plate is placed on the drawer inside the box and the camera captures an image of it to be processed via a MATLAB Graphical User Interface(GUI).

The image then cropped and converted to a grayscale image. This step converts the true color image RGB (Red Green Blue) to the grayscale intensity image. This is done by eliminating the hue and saturation information while retaining the luminance. Doing so helps us in the pixel checking step.

Filtering the image and eliminating unwanted artifacts will reduce errors in results and ease the processing. In our project, we used the *Wiener* filter. The Wiener Filter is a noise filter based on Fourier iteration. Its main advantage is the short computational time it takes to find a solution. The formula of the wiener filter is shown in equation (1):

978-1-5386-1642-0/17/$31.00 ©2017 IEEE

$$G(u,v) = \frac{H*(u,v)Ps(u,v)}{|H(u,v)|2P(u,v)+Pn(u,v),} \quad (1)$$

u and v are the coordinate of the image.
H(u,v): Fourier transform of the point-Spread function.
Ps(v,v): Power spectrum of the signal process obtained by taking the Fourier transform of the signal autocorrelation.
Pn(u,v): Power spectrum of the noise process obtained by taking the Fourier transform of the noise autocorrelation [3].

The CHT can detect circular shapes in an image referring to given parameters.
The Hough transform can be described as a transformation of a point in the *x,y*-plane to the parameter space. The parameter space is defined according to the shape of the object of interest [4].
A straight line passing through the points (x1, y1) and (x2, y2) can in the *x,y*-plan be described by: $y = ax + b$ This is the equation for a straight line in the Cartesian coordinate system, where *a* and *b* represent the parameters of the line. The Hough transform for lines don't use this representation of lines, since lines perpendicular to the *x*-axis will have an a-value of infinity. This will force the parameter space *a,b* to have infinite size. Instead a line is represented by its normal, which can be represented by an angel *q* and *a* length *r (Equation 2)*.

$$r = x\cos q + \sin q \quad (2)$$

The parameter space can now be spanned by *q* and *r*, where *q* will have a finite size, depending on the resolution used for *q*. The distance to the line *r* will have a maximum size of two times the diagonal length of the image.
The circle is actually simpler to represent in parameter space, compared to the line, since the parameters of the circle can be directly transferred to the parameter space. The equation of a circle is:

$$r^2 = (x-a)^2 + (y-b)^2 \quad (3)$$

As it can be seen the circle got three parameters, *r*, *a* and *b*. Where a and *b* are the center of the circle in the *x* and *y* direction respectively and where *r* is the radius. The parametric representation of the circle is:

$$x = a + r\cos q \quad (4)$$

$$y = b + r\sin q \quad (5)$$

Thus, the parameter space for a circle will belong to R3 whereas the line only belonged to R2. As the number of parameters needed to describe the shape increases as well as the dimension of the parameter space R increases so do the complexity of the Hough transform. Therefor is the Hough transform in general only considered for simple shapes with parameters belonging to R2 or maximum R3. In order to simplify the parametric representation of the circle, the radius can be held as a constant or limited to number of known radiuses (Figure 2) [4].

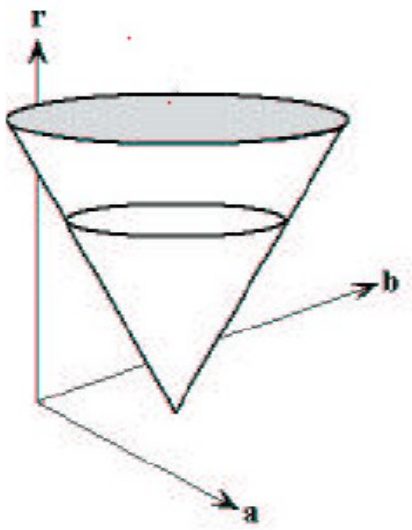


Figure 2: Parameter Space for CHT[5]

First, we find all edges in the image. This step has nothing to do with Hough Transform and any edge detection technique of your desire can be used. It could be Canny, Sobel or Morphological operations (Figure 3).

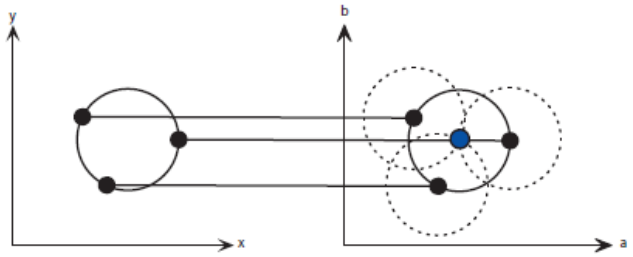


Figure 3: A Circular Hough Transform From The x,y-Space (Left) To The Parameter space[5]

At each edge point we draw a circle with center in the point with the desired radius. This circle is drawn in the parameter space, such that our *x* axis is the *a* - value and the *y* axis is the *b* value while the *z* axis is the radii. At the coordinates that belong to the perimeter of the drawn circle we increment the value in our accumulator matrix, which essentially has the same size as the parameter space. In this way, we sweep over every edge point in the input image drawing circles with the desired radii and incrementing the values in our accumulator. When every edge point and every desired radius is used, we can turn our attention to the accumulator. The accumulator will now contain numbers corresponding to the number of circles passing through the individual coordinates. Thus, the highest numbers (selected in an intelligent way, in relation to the radius) correspond to the center of the circles in the image.

In order to detect the center of the inhibition zones CHT is applied. The antibiotic disks have the same center as the inhibition zones so once detected the centers of the inhibition zones are detected too.
Once we have the center of each inhibition zone we start checking each pixel value around the center in eight directions:

up, down, left, right and diagonally in each of these directions (Figure 4).

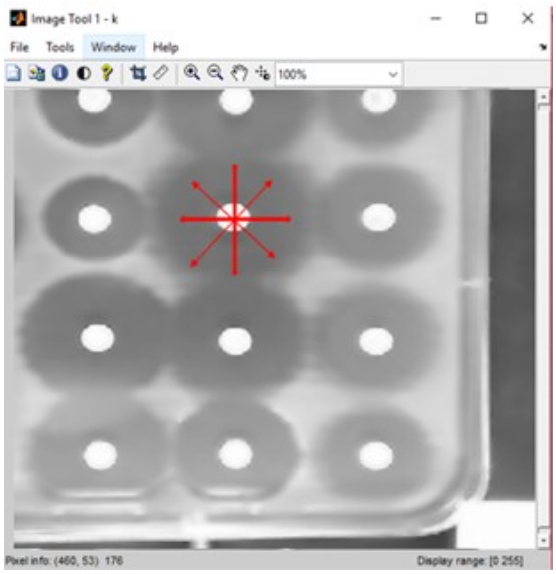


Figure 4: Pixel Checking in 8 Directions

And this is how we are able to detect easily where the inhibition zones start and end. The algorithm allows the users to choose to use either the average of the eight radiuses or the maximum of them. After detecting the center and calculating the radius of each zone, we are able to plot them.

Once we have the radius (diameter) of each zone, we compare the calculated data with a previously stored database that contains critical standard values for each antibiotic and bacteria. We should note that diameters detected are in pixels and not in mm, so in order to be able to compare them with the database a conversion from pixels to mm were done. We used a conversion factor of 0.158 to convert from pixel to mm. We chose this factor after capturing an image of a ruler and determining how much one cm equals in pixel. Then results are shown in a table and bacteria are classified into the three clinical categories: Resistant, Intermediate and Susceptible.

C. Graphical User Interface

In order to facilitate the work for the end user we designed a GUI that runs our algorithm. The user should select the type of bacteria then choose either to load a previously saved image of an Antibiogram or capture a new one using the prototype (Figure 5**Error! Reference source not found.**). The GUI allows the user to select between the average and maximum methods, select the threshold of difference between the pixel value and the maximum radius to be detected (Figure 6**Error! Reference source not found.**). Thus, we can ensure maximum accuracy and precision.

The camera interface shows a preview window directly linked to the camera to allow the user to set the plate and camera position perfectly then take a snapshot of the plate that will be processed or saved. In addition to that the user can choose the threshold of difference between the pixel value and the maximum radius to be detected (Figure 7).

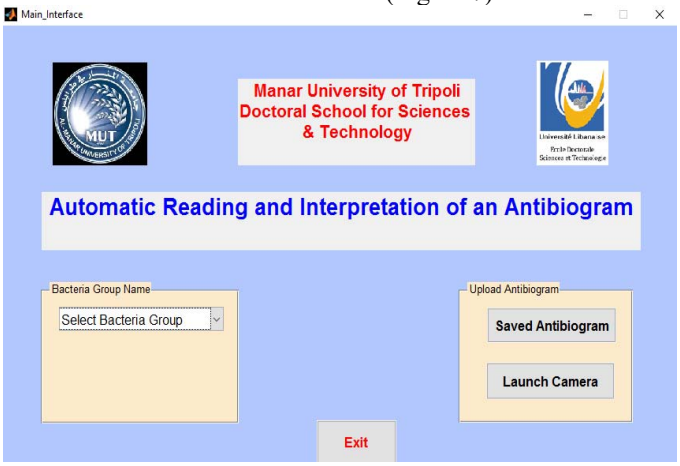


Figure 5: Main Interface

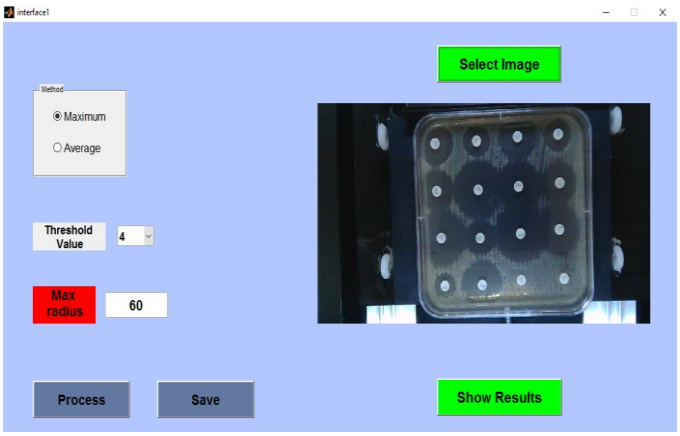


Figure 6: Process Interface

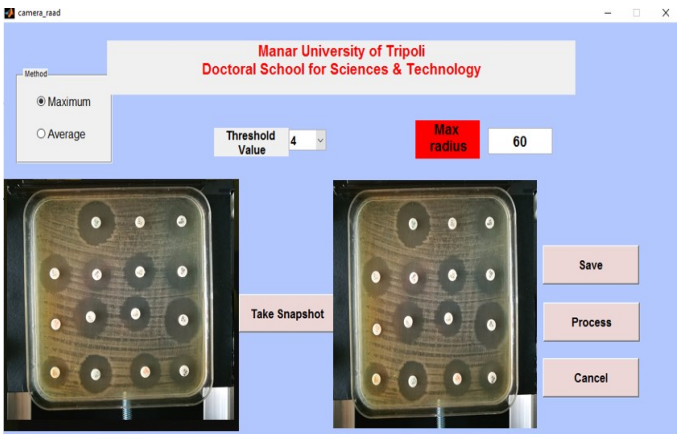


Figure 7: Camera Preview Interface

III. RESULTS

In this section, we will show the reults of the inhibition zones detection and of the clinical categorization of bacteria.

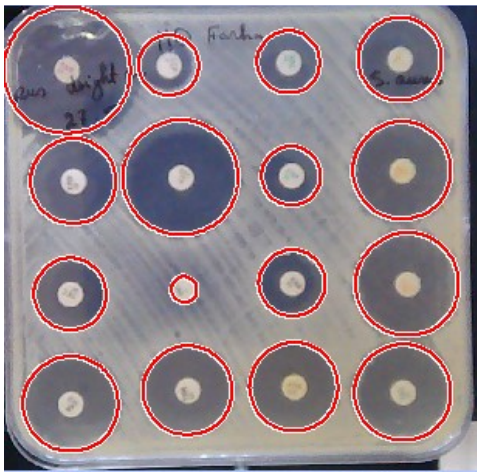


Figure 8: Inhibition Zones Detected Correctly

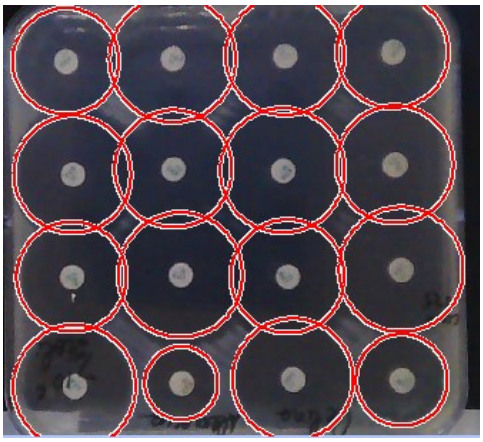
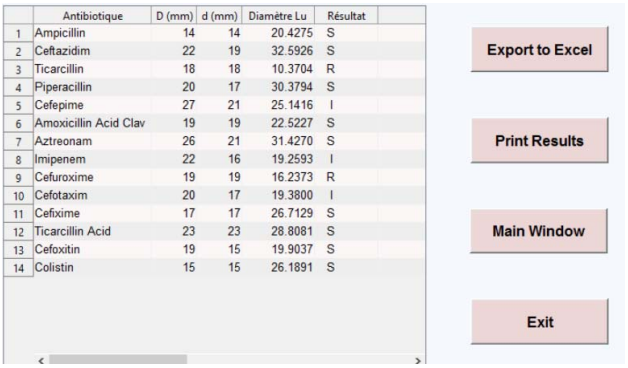


Figure 9: Inhibition Zones Detected Correctly in case of overlapping



	Antibiotique	D (mm)	d (mm)	Diamètre Lu	Résultat
1	Ampicillin	14	14	20.4275	S
2	Ceftazidim	22	19	32.5926	S
3	Ticarcillin	18	18	10.3704	R
4	Piperacillin	20	17	30.3794	S
5	Cefepime	27	21	25.1416	I
6	Amoxicillin Acid Clav	19	19	22.5227	S
7	Aztreonam	26	21	31.4270	S
8	Imipenem	22	16	19.2593	I
9	Cefuroxime	19	19	16.2373	R
10	Cefotaxim	20	17	19.3800	I
11	Cefixime	17	17	26.7129	S
12	Ticarcillin Acid	23	23	28.8081	S
13	Cefoxitin	19	15	19.9037	S
14	Colistin	15	15	26.1891	S

Figure 10: Results Interface

Figures 8 and 9 shows how inhibition zones are detected correctly even in the case of overlapping while figure # shows the interface of the final results.

In order to validate our results, we tested our prototype on 10 Antibiogram plates that contain 120 antibiotic disks. We compared results found by our prototype to results found by manual method. A professional lab technician did the manual measurement, which we compared with our results. The results were impressive. The accuracy of circular shapes detection was almost 100% while the precision in diameters measured varied between 2 and 3 mm.

Despite that the slight difference in diameter measurement rarely affected the clinical categorization of the bacteria. Ten errors were found in the clinical categorization of the bacteria, which means an error percentage of 8%. The errors were as follows:

Table 1: Errors

Susceptible	Intermediate	Resistant
4	4	2

IV. DISCUSSION

The results were very good considering that they were automatically acquired without user intervention for correction. These errors are due to:

1. Non-uniformity of the circular shape of the inhibition zones
2. Turbulence in the quality of the image due to bacteria incubation
3. Severe deterioration in the inhibition zones due to overlapping
4. Lack of accuracy in conversion of the diameters distance from pixels to mm

Despite great results provided by our prototype, it presented some limitation that should be resolved in future work. These limitations were:

1) Inability of the user to select via the interface the antibiotics present on the plate
2) Lack of precision in converting the diameter from pixels to mm
3) Longer processing duration compared to expensive commercial devices in the market

V. CONCLUSION

Reading and interpretation of Antibiogram test is a very important task in healthcare and microbiology fields and manual reading and interpretation of it consumes a lot of time. Our project provided an efficient low-cost prototype to overcome this issue. Our prototype provided satisfying results according to its cost, ease of use and reliability.

REFERENCES

[1] S. H. Gillespie and K. B. Bamford, *Medical microbiology and infection at a glance: [with website]*, 4. ed. Oxford: Wiley-Blackwell, 2012.

[2] N. Burnichon and A. Texier, “L’antibiogramme: la détermination des sensibilités aux antibiotiques,” *Bactériologie–Semestre Été*, 2003.

[3] Veldhuizen, Todd. "The Wiener filter." *Grid filters for local nonlinear image restoration* (1998).

[4] S. Liangwongsan, B. Marungsri, R. Oonsivilai, and A. Oonsivilai, “Extracted circle Hough Transform and circle defect detection algorithm,” *World Acad. Sci. Eng. Technol.*, vol. 5, pp. 432–436, 2011.

[5] H. Rhody, “Lecture 10: Hough circle transform,” *Chester F Carlson Cent. Imaging Sci. Rochester Inst. Technol.*, 2005.