

# Comparación cualitativa y cuantitativa de las técnicas básicas de umbralización global basadas en histogramas para el procesamiento digital de imágenes

## Qualitative and quantitative Comparison of Basic histogram-based global Thresholding techniques for Digital Image Processing

Jimmy Alexander Cortés Osorio<sup>1</sup>, Agustín Muriel<sup>2</sup>, Jairo Alberto Mendoza Vargas<sup>3</sup>

*Universidad Tecnológica de Pereira, Pereira, Colombia*

*jacoper@utp.edu.co*

*jam@utp.edu.co*

*agustin.muriel@utp.edu.co*

**Resumen**— La umbralización es una técnica en el tratamiento digital de las imágenes que permite, entre otras, la separación de los objetos de interés con el fondo. La selección de la estrategia más adecuada para cada caso se hace crítica dependiendo del objetivo principal del procesamiento siendo el tiempo de ejecución una de las mayores limitantes.

Este trabajo realiza un análisis de las técnicas básicas de umbralización conducentes a la binarización de las imágenes en el desarrollo de la investigación del estudio del movimiento oscilatorio mediante visión por computadora del grupo Robótica Aplicada de la Universidad Tecnológica de Pereira.

**Palabras clave**— bimodal, binarización, entropía, global, histograma, mínimo, Otsu, pixel, triangulo, umbralización.

**Abstract**— The thresholding is a technique in digital image processing that allows, among others, the separation of the objects of interest from the background. Selecting the most appropriate strategy for each case is critical depending on the primary objective of the processing time which is one of the main constraints.

This paper provides an analysis of the basic techniques of thresholding leading to the binarization of the images in the research of Oscillatory Motion using computer vision of the Research Group Robótica Aplicada of la Universidad Tecnológica de Pereira.

**Key Wor**— bimodal, binarization, entropy, global, histogram, minimum, Otsu, pixel, thresholding, triangle.

### I. INTRODUCCIÓN.

Para lograr la binarización de una imagen es importante definir cuál es el objeto de interés y cuál es el fondo. De manera iterativa, se puede procurar encontrar un umbral  $T$  el cual defina las regiones, pero este proceso no es siempre posible aplicarlo cuando el entorno no es controlado. En 1979 el investigador japonés Phd Nobuyuki Otsu diseñó una estrategia estadística que no requiere intervención humana la cual utiliza la variancia como medida de la dispersión de los niveles de gris para determinar un único umbral adecuado global [1][2].

Esta técnica puede fracasar cuando no se puede garantizar una luz homogénea sobre el objeto en estudio, o cuando existe deterioro sobre la imagen fuente causado por suciedad o desgaste o simplemente las condiciones en que se ha capturado la fotografía no resultan óptimas. Por lo anterior se ha sido necesario implementar técnica de umbralización locales que particularizan cada zona de la foto para descubrir un umbral independiente en cada pixel de la imagen.

<sup>1</sup>Ingeniero Electricista, Magíster en Instrumentación Física.

<sup>2</sup>Ingeniero Mecánico, Magíster en Sistemas Automáticos de Producción

<sup>3</sup>Ingeniero Electricista, Magíster en Instrumentación Física

II. PIXEL

El pixel es la unidad básica que permite representar un punto localizado en  $p$  donde coinciden las coordenadas  $r$  y  $c$  con información  $I(p)=I(r,c)$  sobre la intensidad del punto la cual va desde 0 hasta  $L$ . Usualmente un pixel requiere de 8 bits para la almacenar su intensidad  $I(r,c)$ , permitiendo una escala de  $L=0$  hasta  $L=255$  posibles valores de gris (Figura 1).

$$Pixel : [p, I(p)] = [(r, c), I(r, c)]$$

Donde:

$$I(r, c) \in \{0, 1, 2...L\}$$

Las coordenadas  $r$  y  $c$ , respectivamente, representan la fila y la columna donde se encuentra el elemento de la imagen con intensidad  $I(r,c)$ . Si  $I(p)$  es solo un número, entonces se dice que el pixel es monocromático, pero si es a color se describe como un vector de 3 elementos.

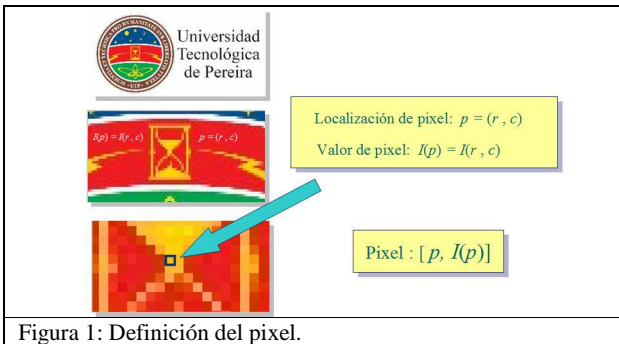


Figura 1: Definición del pixel.

III. IMAGEN EN ESCALA DE GRIS

Un imagen  $I$  es un arreglo matricial bidimensional de  $N$  filas por  $M$  columnas de pixels  $[p, I(p)]$  de información (Figura 2). Es decir:

$$I(r, c) \in \{0, 1, 2...L\}$$

Donde:

$$r = 1, 2...N \quad c = 1, 2...M$$

Se llega a:

$$I = \begin{bmatrix} I(1,1) & I(1,2) & \dots & I(1,N) \\ I(2,1) & I(2,2) & \dots & I(2,N) \\ \vdots & \vdots & \ddots & \vdots \\ I(M,1) & I(M,2) & \dots & I(M,N) \end{bmatrix}$$

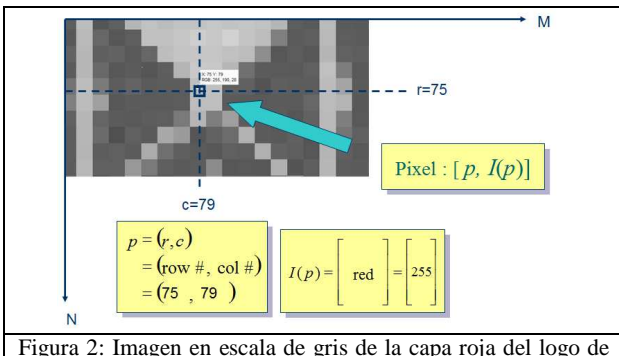


Figura 2: Imagen en escala de gris de la capa roja del logotipo de

la Universidad Tecnológica de Pereira.

IV. IMAGEN A COLOR RGB

V.

Una imagen a color del espacio del color RGB está dada por la parametrización de tres imágenes en escala de grises del mismo orden  $N \times M$ , con intensidades  $I(r,c,k)$ .  $k$  representa la capa del color que se describe en su correspondiente escala de grises para las intensidades del color  $k=1$  rojo,  $k=2$  verde y  $k=3$  azul (Figura 3 y Figura 4).

$$I(r, c, k) \in \{0, 1, 2...L\}$$

Donde:

$$r = 1, 2...N \quad c = 1, 2...M \quad k = 1, 2, 3$$

$$I(r, c, 1) \rightarrow \text{Capa Roja}$$

$$I(r, c, 2) \rightarrow \text{Capa Verde}$$

$$I(r, c, 3) \rightarrow \text{Capa Azul}$$

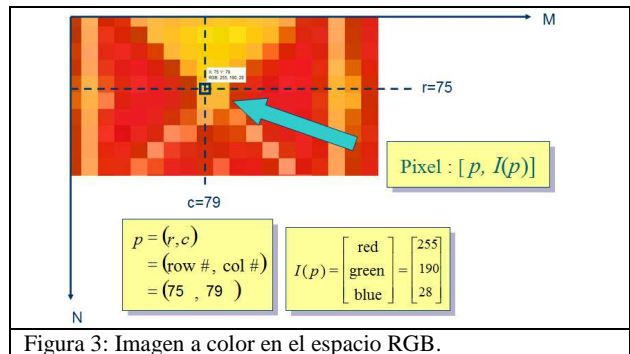


Figura 3: Imagen a color en el espacio RGB.

La imagen a color se genera como la superposición de las tres matrices  $I_R(r,c,1)$ ,  $I_G(r,c,2)$  y  $I_B(r,c,3)$  en escala de grises que entrega el aporte ponderado de cada una de las capas  $k$ .

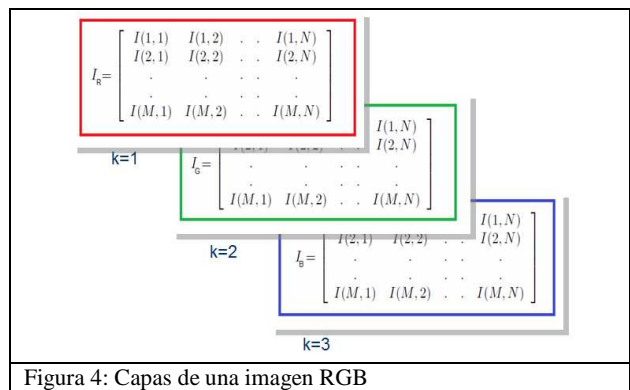


Figura 4: Capas de una imagen RGB

La figura 5 representa cada una de las capas de la imagen a color en RGB del logotipo de la Universidad Tecnológica de Pereira.

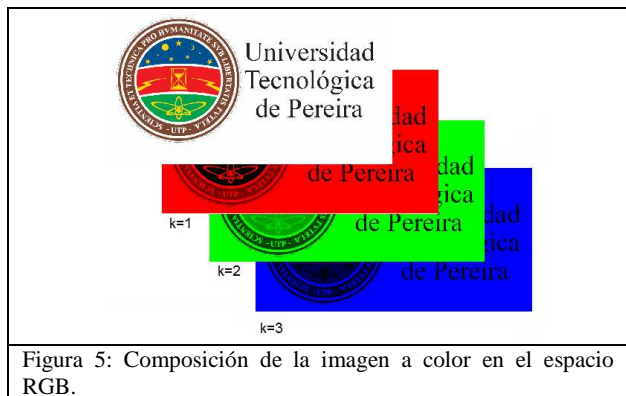


Figura 5: Composición de la imagen a color en el espacio RGB.

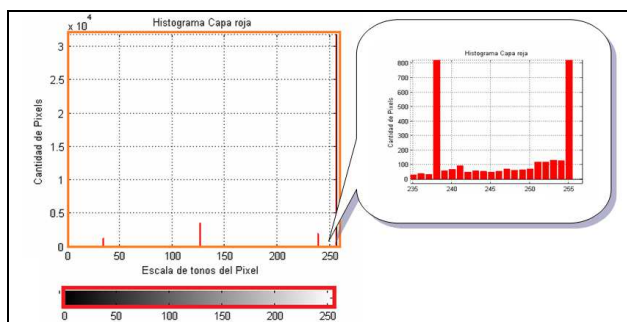


Figura 6: Histograma de la capa roja (en escala de grises) del logo de la Universidad Tecnológica de Pereira con zoom sobre la zona  $235 \leq k \leq 255$

### V. EL HISTOGRAMA[3]

El histograma  $h_I$  de una imagen  $I$  de escala de grises es un arreglo vectorial de orden  $L \times L$  elementos que representa gráficamente los posibles valores de intensidad desde 0 hasta  $L-1$  en el eje horizontal y con la cuenta  $h_I(k)$  de pixels con igual intensidad  $I(r,c)=k$  en el eje vertical mediante barras.

Sea  $s$  un conjunto, y  $\#s$  cada uno de los elementos de  $s$ . El histograma  $h_I(k)$  de una imagen  $I$  se define matemáticamente como:

$$h_I(k) = \# \{ (r,c) \mid I(r,c) = k, \quad r = 1, 2, 3 \dots N, \quad c = 1, 2, 3 \dots M \}$$

Donde:

$$k = 0, 1, 2 \dots L - 1$$

Se debe notar que la sumatoria de todos los elementos del arreglo que forma el histograma es igual a la totalidad de los pixels que forma la imagen (Ecuación 1):

$$\sum_{k=0}^{L-1} h_I(k) = N \times M$$

Ecuación 1: Total de la suma del arreglo de histograma.

Si se divide el histograma  $h_I$  sobre la totalidad de pixels  $N \times M$  de la imagen, cada cuenta  $h_I(k)$  representa la probabilidad de ocurrencia de de intensidad  $I(r,c)=k$  (Ecuación 2).

$$\sum_{k=0}^{L-1} \frac{h_I(k)}{N \times M} = \sum_{k=0}^{L-1} p_I(k) = 1$$

Ecuación 2: Histograma por probabilidad de ocurrencia.

El histograma proporciona información sobre la distribución de la luz en la imagen en forma de brillo o contraste y puede ser utilizado para ajustar dichos parámetros. La figura 6 muestra un histograma con su respectivo detalle.

Los autores implementaron una versión del histograma en Matlab 7, pero puede ser extendida a cualquier otro lenguaje. En la figura 7 se entrega el código de los autores.

Se debe hacer notar que Matlab 7, en su toolbox de Procesamiento de Imágenes, posee una función propietaria denominando *imhist()* que realiza el gráfico.

```
function histograma=mihist( RGB, tipo, color, graficar)
% Grafica el histograma de una imagen para 255 puntos. mihist( RGB, tipo, color, graficar)
% use tipo 'u' para graficar por unidad y 'p' como la cantidad de pixels. El tercer
% parametro es el color del histograma 'r'(rojo) y l para ver la grafica.
% Si se le asigna una variable, guarda el resultado en ella. El indice debe
% coincidir.
% f=imread('utp.png');
% z=mihist(f(:,:,1),'u','r',1);
factor=1;
formato=lower(tipo);
[fil,col]=size(RGB);
histograma(1:256)=0;
if formato=='u'
    factor=(fil*col)/100;
end
for nivel=0:255
    histograma(nivel+1)=length(find(RGB==nivel))/factor;
    % El vector no puede iniciar en "0"
end
xajustado=[0:255];
% Se ajusta porque la grafica inicia en "0" y no en "1"

if graficar==1
    bar(xajustado,histograma,color);
    title('Histograma de la Grafica')
    xlabel('Escala de tonos del Pixel')
    if formato=='u'
        ylabel('Valor Probabilidad (Por Unidad)')
    else
        ylabel('Cantidad de Pixels')
    end
    grid
    axis([0 256 0 max(histograma)])
    %axis tight
end
end
```

Figura 7: Función que genera el histograma

### VI. LA UMBRALIZACIÓN

La umbralización es una técnica de segmentación simple y eficiente que permite separar los pixel de la imagen en escala de grises en dos categorías a partir de un una ventana o borde umbral. El umbral puede ser función de la posición  $p(r,c)$ , de la vecindad  $N(r,c)$  y de la intensidad  $I(r,c)$  actual del pixel.

$$T = T( (r,c), N(r,c), I(r,c) )$$

#### A. Umbral global [3][4]

El umbral fijo o global es aquel que es único sobre toda la imagen.

$$b(r,c) = \begin{cases} t_0 & \text{si } I(r,c) < T \\ t_1 & \text{si } I(r,c) \geq T \end{cases}$$

Cuando el valor  $t_0=0$  y el valor  $t_1=1$  se dice que la imagen se ha binarizado. Esta estrategia puede resultar muy simple cuando se

posee un histograma que claramente se define como bimodal (Figura 7).

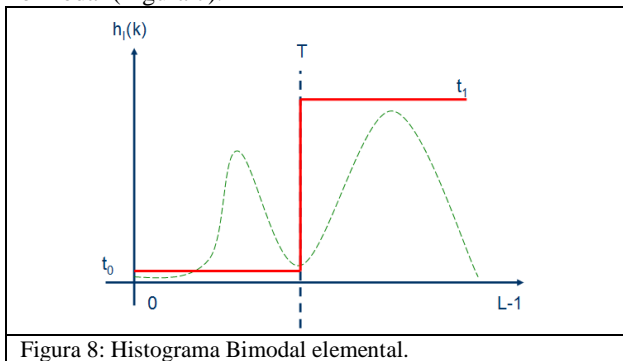


Figura 8: Histograma Bimodal elemental.

VII. BINARIZACIÓN GLOBAL DE LAS IMÁGENES EN ESCALA DE GRIS [5][6]

Las técnicas basadas en el histograma para la determinación del umbral más adecuado para la discriminación de los objetos de estudio y el fondo de la imagen resultan de gran utilidad ya que son fáciles de implementar, simples y generalmente rápidos.

La figura 9 representa un circuito impreso [10] sobre un fondo no uniforme. Este posee un histograma el cual se aprecia en la figura 10 .

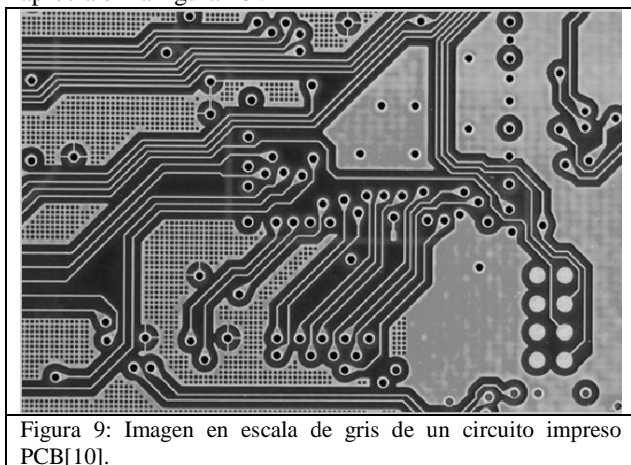


Figura 9: Imagen en escala de gris de un circuito impreso PCB[10].

A. Método del Mínimo

El histograma de la figura 10 es de tipo bimodal, lo que hace relativamente fácil la selección del umbral más adecuado. De la inspección visual del histograma, se puede ver que una buena opción para la separación de fondo y los objetos es un lugar el cual corresponde al punto más bajo entre los dos picos principales. Para realizar el algoritmo se recomienda suavizar la curva reiteradas veces hasta lograr que solo existan 2 máximos locales.

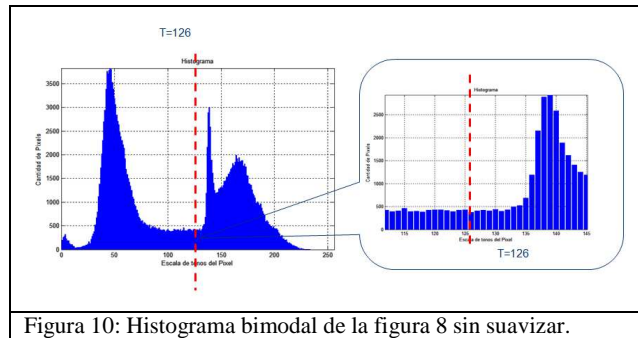


Figura 10: Histograma bimodal de la figura 8 sin suavizar.

Si se suaviza el histograma bimodal, se pueden definir más claramente el umbral adecuado ya que este se encuentra en el valle localizado entre los 2 únicos picos. Para localizar los máximos locales, se recurre a los criterios de la derivada que indican:

*Si  $h'(k)=0$  y  $h''(k)<0$  entonces es un máximo local*

El histograma debe generar dos únicos máximos locales. Seguidamente, se busca el valor mínimo entre los dos máximos locales siendo este punto el que posee el umbral adecuado T (Figura 11).

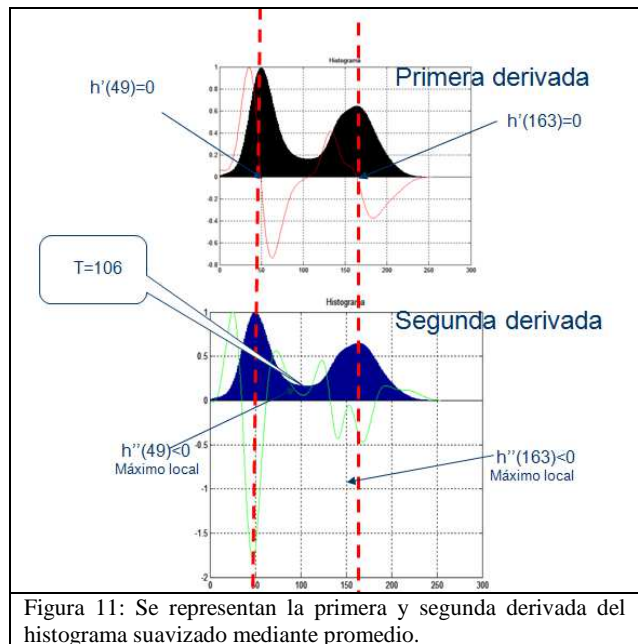


Figura 11: Se representan la primera y segunda derivada del histograma suavizado mediante promedio.

El algoritmo implementado, permite verificar que el umbral T=106 resulta adecuado (Ver figura 12).

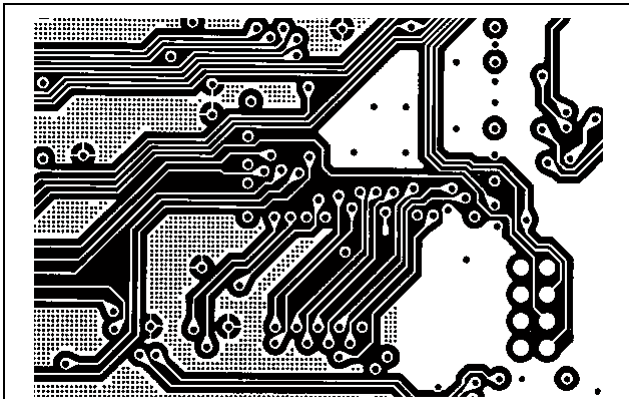


Figura 12: Resultado de la umbralización mediante el método del Mínimo T=106 suavizado.

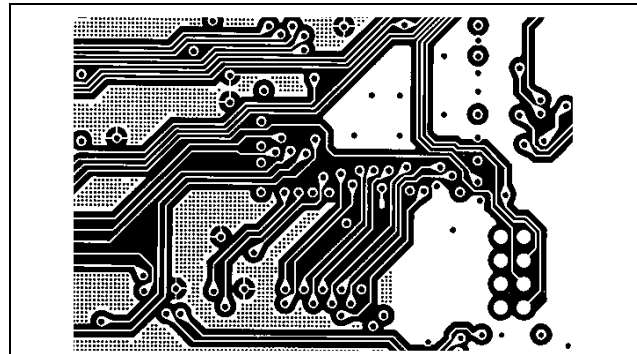


Figura 14: Resultado de la umbralización mediante el método de la Entropía T=103.

### B. Método de la Entropía Shannon [6]

Se usa la entropía  $H$  como una medida del contenido de la información. El umbral  $T$  separa la información en dos clases correspondientes al fondo y al objeto, determinando la entropía asociada a ellas. El umbral  $T$  se determina para aquel valor donde la entropía es máxima. La entropía se puede calcular a través de la ecuación 3.

$$H = - \sum_{k=0}^T p_1(k) \times \log_2(p_1(k)) - \sum_{k=T+1}^{L-1} p_1(k) \times \log_2(p_1(k))$$

Ecuación 3: Ecuación de la entropía.

La figura 13 presenta el código implementado en Matlab mediante el cual los autores calculan la entropía de una imagen para la determinación de su umbral.

```
function umbral=migraythresentropy(RGBgray)
% Algoritmo para determinar umbral por entropia (Ver entropy).
% Encuentra el umbral entre 0 y 255 para una imagen en
% escala de grises.

h=imhist(RGBgray,'p','z',0);
lh=length(h);
[filas,cols]=size(RGBgray);
tam=filas*cols;
maxH=0;

for T=1:lh-1
p1=sum(h(1:T))/tam;
h1=-sum(p1*log2(p1));

p2=1-p1;
h2=-sum(p2*log2(p2));
H=h1+h2;

if H >= maxH
maxH=H;
umbral=(T-1)/255;
end
end
end
```

Figura 13: Función para el cálculo del umbral por entropía.

En la figura 14 se muestra el resultado de la umbralización mediante el método de la entropía.

### C. Método del Triángulo

Para determinar el umbral adecuado mediante el método del triángulo, se deben determinar el valor máximo de la probabilidad  $P_{max}$  y mínimo  $P_{min}$  de la gráfica del histograma. El umbral se determina mediante otra línea recta la cual debe cortar perpendicular a la primera y rodear el histograma hasta encontrar la máxima distancia  $d$ . El punto donde la recta toca el histograma corresponde a umbral deseado  $T$ .

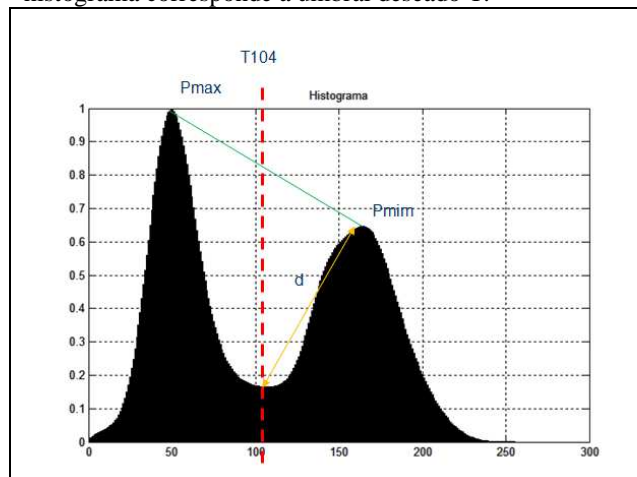


Figura 15: Procedimiento para la determinación del umbral mediante el método del Triángulo.

Esta técnica resulta adecuada cuando el histograma es de tipo bimodal y posee solo picos claramente definidos. En la figura 15 se ilustra el procedimiento cuando sobre una curva suavizada del histograma que permite definir mejor el histograma. La figura 16 presenta el resultado de la umbralización una vez aplicada la técnica.

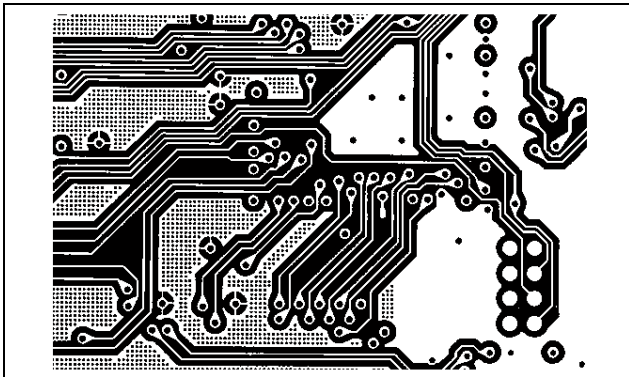


Figura 16: Resultado de la umbralización mediante el método del Triángulo T=104.

**D. Método de Otsu [6]**

El método de Otsu considera que el histograma es bimodal. Por otro lado, este itera a lo largo del histograma calculando para cada valor estimado posible de T [0 hasta 255] La varianza de pesos dentro de cada clase. El umbral óptimo se logra cuando la varianza entre clases entregue un valor mínimo. La varianza entre clases se calcula mediante la ecuación 4:

Donde las probabilidades de cada clase se determinan con la ecuación 5:

$$P_1(t) = \sum_{i=0}^t p(i) \quad P_2(t) = \sum_{i=t+1}^{255} p(i)$$

Ecuación 5: Probabilidades de cada clase

Y los promedios de clase a través de las ecuaciones en 6:

$$\mu_1(t) = \frac{\sum_{i=0}^t i \cdot p(i)}{P_1(t)} \quad \mu_2(t) = \frac{\sum_{i=t+1}^{G-1} i \cdot p(i)}{P_2(t)}$$

Ecuación 6: Promedios de clase

Con las varianzas de cada clase presentadas mediante las ecuaciones en 7:

$$\sigma_1^2(t) = \sum_{i=0}^t [i - \mu_1(t)]^2 \frac{p(i)}{P_1(t)} \quad \sigma_2^2(t) = \sum_{i=t+1}^{255} [i - \mu_2(t)]^2 \frac{p(i)}{P_2(t)}$$

Ecuación 7: Varianzas de cada clase

En la figura 17 se encuentra el algoritmo implementado para entregar el umbral mediante el método de Otsu. Matlab, dentro de su toolbox de Procesamiento de imágenes, posee la función *graythresh()* para su cálculo.

```
function umbral=migraythresh( RGBgray)
% Algoritmo de Otsu. Encuentra el umbral
% entre 0 y 255 para una imagen en
% escala de grises.
h=mihist( RGBgray, 'p', 'r', 0);
lh=length(h);
[filas,cols]=size( RGBgray);
tam=filas*cols;
maxV=0;
for T=1:lh-1
Wb=sum(h(1:T))/tam;
Ub=sum( ([0:T-1]).*h(1:T) )/sum(h(1:T));
%V2b=sum( ( ([0:T-1]-Ub).^2 ).*h(1:T) )/sum(h(1:T));
%Wf=sum(h(T+1:end))/(filas*cols);
Wf=1-Wb;
Uf=sum( ([T:lh-1]).*h(T+1:end) )/sum(h(T+1:end));
%V2f=sum( ( ([T:lh-1]-Uf).^2 ).*h(T+1:end) )/sum(h(T+1:end));
%WCV=Wb*V2b+Wf*V2f;
BCV=Wb*Wf*(Ub-Uf)^2;
if BCV >= maxV
maxV=BCV;
umbral=(T-1)/255;
end
end
end
```

Figura 17: Función de los autores para la determinación del umbral mediante el método de Otsu.

En la figura 18 se muestra el resultado de la aplicación del método de Otsu para la umbralización mediante la función implementada por los autores.

$$\sigma_w^2(t) = P_1(t)\sigma_1^2(t) + P_2(t)\sigma_2^2(t)$$

Ecuación 4: La varianza entre clases

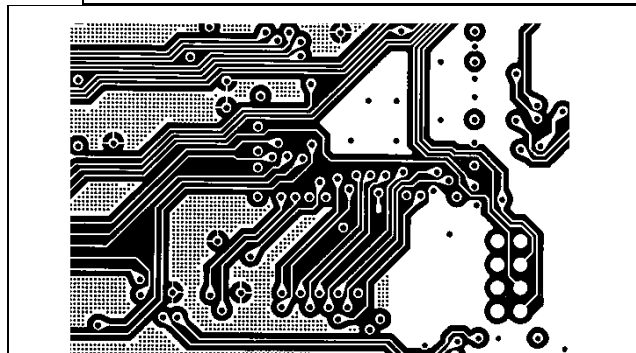


Figura 18: Resultado de la umbralización mediante el método de Otsu T=107

**VIII. METODOLOGÍA**

Las investigaciones realizadas por el grupo y el semillero de investigación en robótica para la elaboración de este documento, son de tipo descriptivo, y evaluativo ya que se pretende mostrar las características que tiene Matlab para desarrollar aplicaciones de visión por computador.

Para la realización de esta investigación se llevaron a cabo algunos procesos, a saber:

- Se realizó una búsqueda de las principales funciones que posee Matlab para el tratamiento de imágenes.
- Se evaluaron dichas funciones y posteriormente se implementaron funciones propias para el tratamiento de imágenes.

- Se midió el tiempo de operación de cada algoritmo implementado.

Para esta investigación se usó Matlab versión 7.0 gracias a su flexibilidad en lo que tiene que ver con el análisis numérico. Pese a que Matlab 7.0 posee un gran número de las funciones para el tratamiento, preprocesado, segmentación y reconocimiento de los objetos las funciones fueron implementadas a partir de los algoritmos existentes para estas tareas.

## IX. RESULTADOS

La umbralización global resulta muy eficiente en términos del tiempo y la simplicidad de su implementación. Existe abundante literatura al respecto y se aprecia que pese a esto no es un tema finalizado en su estudio y tratamiento. El análisis cualitativo se resume en la tabla 1:

algoritmo	Tiempo Ejecución
Mínimo	0.864461
Entropía	0.420397
Triángulo	0.422047
Otsu	0.428861

Tabla 1. Comparativo de la rapidez de los algoritmos de umbralización global.

Estas pruebas experimentales fueron realizadas sobre una computadora portátil HP 672 Notebook PC, Petium dual Core de 2.3 GHz, RAM de 3G, Sistema Operativo Windows 7 a 64 bits Home Premium, con Matlab 7.10 (R2010a). Es de notar que las pruebas realizadas no fueron ejecutadas bajo condiciones de laboratorio, por lo que los resultados pueden variar.

## X. CONCLUSIONES

Durante las pruebas de binarización de la tarjeta de circuito impreso, se pudo verificar que el método de la entropía fue muy acertado y rápido. En general, los autores pueden considerar que cuando las imágenes son de tipo bimodal, es posible recurrir a métodos simples de binarización como lo es la entropía, pero si por el contrario no lo es, estos métodos globales fallan y se debe recurrir a estrategias de umbralización local que analicen la imagen teniendo en cuenta la vecindad de cada pixel y no vista como un todo.

## REFERENCIAS

- [1]. Prof Nobuyuki Otsu, [En línea] disponible en: <http://www.see.ed.ac.uk/bliss08/keynotes/otsu.html> (Consultado el 13 de julio de 2011)
- [2]. Vassilios Vonikakis, Ioannis Andreadis, Nikolaos Papamarkos, "ADAPTIVE DOCUMENT BINARIZATION" [En línea] disponible en <http://robotics.pme.duth.gr/pubs/Conferences/ADAPTIVE%20DOCUMENT%20BINARIZATION%20A%20human%20vision%20approach.pdf> (Consultado el 13 de julio de 2011)
- [3]. Wilhelm Burger • Mark J. Burge, Principles of Digital Image Processing, Springer-Verlag London Limited 2009.
- [4]. Handbook of video and imagen processing, University of Texas, Elsevier Academic Press, USA, 2005.
- [5]. Mahboob H. Chowdhury and Warren D. Little, "Image thresholding techniques", [En línea] disponible en: [http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=519599](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=519599) (Consultado el 13 de julio de 2011)
- [6]. Tübitak Marmara Research Center, "Survey over image thresholding techniques and quantitative performance evaluation", [En línea] disponible en: <http://web.cs.wpi.edu/~trascrizione/MethSurvey.pdf> (Consultado el 13 de julio de 2011)
- [7]. "printing circuit board", [En línea] disponible en: <http://electroniccircuit1.com/wp-content/uploads/2010/12/Printing-Circuit-Board.jpg> (Consultado el 20 de Agosto de 2011)