



VISIÓN
ELECTRÓNICA

ALGORITMOS DE PROCESAMIENTO DE IMÁGENES SATELITALES CON TRANSFORMADA HOUGH

IMAGE PROCESSING ALGORITHMS FOR SATELLITE WITH HOUGH TRANSFORM

Frank N. Giraldo¹

M. Fernando González²

Esperanza Camargo C.³

Fecha de envío: Mayo de 2011

Fecha de recepción: Mayo de 2011

Fecha de aceptación: Julio de 2011

Resumen:

La segmentación es uno de los pilares del tratamiento digital de imágenes que involucra un conjunto de técnicas con el objetivo de mejorar la calidad o facilitar la búsqueda de información; múltiples procesos son llevados a cabo dentro de ese marco, utilizados actualmente en el desarrollo de la visión artificial. El grupo de investigación DIGITI, de la Universidad Distrital Francisco José de Caldas (Colombia), se ha interesado en incorporar características tolerantes a cierto nivel de distorsión que, en definitiva, proporcionen robustez en los resultados investigados sobre imágenes, por tal motivo, el presente artículo describe la implementación de un modelo de análisis para imágenes satelitales de la malla vial usando la Transformada Hough, técnica que requiere mucho tiempo de computación para conseguir resultados óptimos, pero que entre otras ventajas, permite subdividir el espacio paramétrico en celdas acumuladoras, creando un plano independiente para ello, por lo que se convierte en un método robusto frente al ruido, dotado de procesos estadísticos para el análisis de la imágenes.

Palabras clave:

Transformada Hough, algoritmo, espacio paramétrico, celdas acumuladoras, imagen satelital.

Abstract:

Segmentation is one of the pillars of digital image processing that involves a set of tech-

¹ Ingeniero de Control e Instrumentación, MSc. (c) en Automatización, Universidad Nacional de Colombia. Docente de la Universidad Distrital Francisco José de Caldas. e-mail: fngiraldor@udistrital.edu.co.

² Tecnólogo en Electrónica, Universidad Distrital Francisco José de Caldas. Investigador del grupo DIGITI. e-mail: manfego6@hotmail.com

³ Ingeniera en Control electrónico e Instrumentación, Esp. en Instrumentación Electrónica, Universidad Santo Tomás, PhD.(c) en Ingeniería Pontificia Universidad Javeriana. Directora del grupo de Investigación DIGITI. Docente de la Universidad Distrital Francisco José de caldas. e-mail: ecamargoc@udistrital.edu.co.

niques in order to improve quality or facilitate the search for information in your shots, multiple processes are carried out within this framework currently used in the development of artificial vision. DIGITI research group of the Distrital University “Francisco José de Caldas” (Colombia), has been interested in tolerant features include some level of distortion that ultimately provide robustness of the results investigated over images, for this reason, this article describes and implements a model for analyzing satellite imagery of the road network using the Hough Transform, a technique that requires much computing time to achieve optimal results, but among other advantages, allows subdividing the parameter space into accumulator cells, creating a This separate level, so it becomes a robust method against noise, statistical process equipped to analyze the images.

Keywords:

Hough transform, algorithm, parameter space, accumulating cells, satellite image.

1. Introducción

La idea de considerar a la visión artificial como una ciencia se fundamenta en el gran número de problemas que se pueden estudiar, desde diversos puntos de vista, asociados a un sinnúmero de conceptos de múltiples áreas del conocimiento, entre estos se tiene el concepto de procesamiento digital, mediante el cual se agrupan varios métodos que partiendo de unas bases matemáticas, plantean la consecución y organización de información respecto a una imagen.

La Transformada de Hough trata de detectar formas geométricas sencillas en una imagen, en su procesamiento emplea el conjunto total de la imagen, haciéndolo robusto ante la

presencia de ruido o discontinuidades de las etapas previas. De hecho, para su ejecución requiere de una imagen binarizada a la que se han seleccionado previamente los bordes. Por lo tanto, la entrada al algoritmo de Hough es una imagen a la que se ha aplicado un proceso de segmentación [1].

Es difícil satisfacer todas las propiedades para el análisis óptimo de las imágenes satelitales, por esto interesa incorporar características tolerantes a cierto nivel de distorsión que, en definitiva proporcionen robustez en los resultados. Como la Transformada de Hough tiene potente capacidad para integrar varias informaciones visuales, ha sido aplicada a problemas de visión tales como detección y restauración de imágenes.

2. Técnicas clásicas de segmentación de imágenes

La segmentación de imágenes permite dividir una imagen, para aislar segmentos, regiones u objetos individuales, que son homogéneos con respecto a algún rasgo; de tal forma que se puedan extraer características o identificar y clasificar los objetos presentes en la imagen. Estos datos pueden ser útiles para cuantificar regiones, tomar medidas sobre los objetos, discriminar automáticamente los elementos presentes, definir bordes y formas, diferenciar tonalidades, etc. Con base en ello, se han desarrollado aplicaciones, por ejemplo, en imágenes de cultivos en agricultura [2].

Los criterios generales en los cuales se basan la mayoría de técnicas para la segmentación de imágenes son:

Similitud: cada uno de los píxeles de un elemento tiene valores parecidos con respecto a alguna propiedad determinada.

Discontinuidad: los objetos y más concretamente las regiones que los diferencian destacan del entorno y tienen por tanto bordes bien definidos.

Conectividad: los píxeles pertenecientes al mismo objeto o región tienen que ser contiguos, es decir, tienden a agruparse constituyendo regiones homogéneas.

En la segmentación de imágenes en escala de grises el criterio de discriminación es la intensidad o el tono de gris para buscar agrupar los píxeles de similar comportamiento. En algunas aplicaciones las imágenes pueden binarizarse (umbralizar dejando solo dos tonos: blanco y negro) para hacer más sencillo y veloz el procesamiento computacional [3].

En la segmentación de imágenes a color, y si para la aplicación no es tan relevante la información de color que contiene, se puede simplificar el procesamiento haciendo una conversión a escala de grises para así segmentarla. También se pueden emplear la mayoría de métodos que se han desarrollado para el caso de imágenes en escala de grises, aplicándolos de forma independiente a cada componente de color y procesándolos por separado como si fueran monocromáticos, solo teniendo en cuenta el espacio de color y las cualidades propias del color.

Un color usualmente se puede especificar por la teoría tri-cromática usando tres coordenadas o parámetros, es decir formado por tres luces por separado, roja verde y azul a partir de las cuales se puede visualizar cualquier color visible, esos tres parámetros definen el color en el denominado espacio de color, que es un modelo que permite especificar, crear y visualizar un color.

De acuerdo a la aplicación será más conveniente usar uno u otro espacio de color.

2.1 Espacios de color:

RGB (Red Green Blue): es un sistema de color basado en la teoría tri-cromática, es un método que describe la cantidad de luz que debe ser emitida para producir un color, tiene como colores primarios al rojo, el verde y el azul, cuya fusión recompone la luz blanca, por eso a esta mezcla se la llama aditiva. Pero este espacio no es uniforme, en el sentido de que las diferencias de color percibidas como iguales por el ojo humano no corresponden a iguales distancias euclídeas. Es empleado por los sensores de las cámaras digitales y tubos de rayos catódicos [4].

HSL, HSB o HSV, HSI (Hue Saturation and Lightness/Brightness/Intensity): es un sistema que representa intuitivamente el color basado en su percepción, es decir la forma como lo percibe el ser humano. Puede entenderse como una transformación del espacio RGB al espacio perceptual. Tiene tres variantes en cuanto a si el color se representa por luminancia, brillo o intensidad [4].

CIE (CIELuv, CIEUvw and CIELab): es el sistema de color más cercano a la percepción visual del hombre, se basa en el sistema de medición CIE el cual está relacionado directamente con la visión humana [4].

CMY(K) (Cyan Magenta Yellow (Black)): es un sistema de color que describe la cantidad de la luz reflejada o sustraída desde la superficie del pigmento, la mezcla de los tres colores primarios pigmento en teoría debería producir el negro, el color más oscuro y de menor o nula cantidad de luz, por lo cual esta mezcla es conocida

como sustractiva, es usado en impresoras y copiadoras [4].

YIQ, YUV, YCbCr, YCC, YPbPr (Luminance - Chrominance): estos espacios de color son usados por las transmisiones de Televisión (NTSC, PAL, SECAM) y para compresión de imágenes y video.

Una alternativa para la segmentación de imágenes a color y más robusta a cambios de iluminación, consiste en tomar la imagen del espacio RGB (en el que están las imágenes captadas por los sensores de las cámaras) y convertirla al espacio HSI (perceptual basado en la intensidad) y aplicar los métodos para segmentación de imágenes en escala de grises en ese espacio de color para cada componente de color por separado.

Otras cualidades del color también pueden ser muy útiles para encontrar similitud, discontinuidad y conectividad en imágenes a color como el tono, la saturación, el brillo y la luminosidad.

Las técnicas clásicas para la segmentación de imágenes se pueden dividir a grandes rasgos en seis métodos [4]:

2.1.1 Método 1: Segmentación basada en el valor del píxel.

Segmentación por umbralización (basada en el histograma). La forma más simple de segmentar una imagen es en el caso que los píxeles o regiones de una imagen se pueden diferenciar y caracterizar de acuerdo a su escala de gris. Para ello se emplea el histograma, el cual indica el número de puntos de la imagen que poseen un determinado nivel de gris. El objetivo es la determinación del umbral que separa los niveles de intensidad en cada píxel de la imagen. Existen multitud

de algoritmos en la literatura que intentan estimar automáticamente el umbral en un histograma. El método no paramétrico más extendido es el conocido como método de OTSU. Los métodos paramétricos consisten en ajustar el histograma a una serie de funciones conocidas [4], [5], [6].

Para imágenes a color, cada píxel está caracterizado por tres valores RGB, se puede obtener un histograma en 3D y determinar un valor de umbral por cada componente de color y de esta forma agrupar o formar cluster de píxeles de regiones similares.

2.1.2 Método 2: Segmentación basada en el área o por regiones.

Dentro de este tipo de segmentación, se encuentran los métodos de crecimiento de regiones y división-fusión. La segmentación basada en regiones busca dividir la imagen en particiones que comparten ciertas características comunes.

Segmentación por crecimiento de regiones desde semillas. Se basa en los criterios de conectividad y similitud de puntos. Para cada región que se desea segmentar, el crecimiento se inicia a partir de uno o más puntos iniciales, conocidos como germen o semilla. A partir de estos, la región progresivamente va creciendo, al agregar aquellos vecinos sucesivos que satisfacen el criterio de aceptación prefijado. El crecimiento de esta región termina cuando no hay elementos que cumplan tal condición. Al finalizar el proceso, se tiene un conjunto de regiones que engloban un conjunto de puntos que la definen.

El principal problema en el caso del crecimiento de regiones es la ubicación de las semillas sobre la imagen de forma que a partir

Una alternativa para la segmentación de imágenes a color y más robusta a cambios de iluminación, consiste en tomar la imagen del espacio RGB y convertirla al espacio HSI y aplicar los métodos para segmentación de imágenes en escala de grises en ese espacio de color para cada componente de color por separado.

de esas semillas se generen las regiones. Por ello es importante que las semillas se especifiquen en posiciones de la imagen que corresponden a puntos de interés. La selección de las semillas puede hacerse de forma automática o con la ayuda de un experto que las seleccione de forma manual, que sería otra posible opción para que el método se orientara de una forma semiautomática [4], [5], [6].

Segmentación por división y fusión (*Split & Merge*). Se basa en la división y unión de regiones. Normalmente las dos fases del algoritmo se efectúan en ese orden, primero una división en regiones para posteriormente unir aquellas regiones vecinas similares.

Inicialmente la imagen es considerada como una única región cuadrada. Iterativamente, cada una de las regiones que componen la imagen son analizadas comprobando si dicha región es uniforme o no, según algún criterio previamente establecido, tal como que la diferencia de los valores de intensidad de los píxeles no deben diferenciarse entre sí más de un cierto valor respecto al de uno central de referencia. Si es uniforme esta región se mantiene intacta. Si no es uniforme, la región es dividida en cuatro partes iguales que serán analizadas en etapas posteriores. Este proceso iterativo continúa, hasta que las regiones son uniformes o demasiado pequeñas.

Cuando el proceso de división termina tiene lugar la fase de unión de las regiones obtenidas. En este caso, se ha de comprobar si dos bloques consecutivos, adyacentes, es decir, que estén en contacto espacialmente, cumplen el criterio anterior. Si es así, dichos bloques quedaran fusionados, formando un único bloque. Dicho proceso se repite hasta llegar a la idempotencia, es decir, el momen-

to en el cual ya no pueden fusionarse mas bloques.

El criterio adoptado en este proceso es igual que el aplicado en el caso anterior, con el apunte adicional de que el umbral de decisión puede ser variado entre la etapa de división y la de unión. Por lo general se suele tomar un umbral bastante estricto al realizar la división, suavizándolo a continuación en la fase de unión [4], [5], [6].

2.1.3 Método 3: Segmentación mediante detección de contornos.

Un contorno es un conjunto conectado de píxeles que tienen el mismo nivel de intensidad. Los contornos de una imagen se pueden clasificar en dos categorías contornos definidos por la intensidad de los píxeles y contornos definidos por la textura de una región. La detección de contornos se realiza mediante búsqueda de discontinuidades en los niveles de grises de la imagen, es decir, de píxeles en los cuales el nivel de gris varíe abruptamente respecto al de sus vecinos.

Métodos basados en la derivada. Las dos técnicas más conocidas de detección de píxeles en el contorno consisten en la búsqueda de puntos de ruptura en la imagen, como por ejemplo, detectar píxeles de gradiente elevado (utilizando la primera derivada) o al detectar cruces por cero de la laplaciana (segunda derivada).

En esta categoría quedan incluidos los operadores como Sobel (1968), Prewitt (1970) y Roberts (1965), que tiene cada uno su propio diseño de acuerdo a como definen las derivadas. Canny (1986) encuentra los bordes por la búsqueda de un máximo local del gradiente, aplicando la derivada de un filtro gaussiano. El operador Marr-Hildreth tiene

la ventaja de delimitar contornos cerrados, y Boman cuya variante es usada en segmentación de imágenes 3D [5], [6], [7].

Transformada Hough. Una vez detectados los puntos que definen un contorno mediante cualquiera de las técnicas basadas en la derivada, se pueden presentar discontinuidades de estos puntos por píxeles perdidos o ruido y el contorno puede quedar incompleto o abierto, existe la posibilidad de unir las posibles discontinuidades extrapolando los puntos a formas predeterminadas tales como, rectas, circunferencias, etc. La técnica para realizar esta unión es la transformada de Hough [3], [6], [8].

Modelos deformables. Tiene como ventaja que permite obtener contornos o superficies cerradas. Consiste en hacer evolucionar un objeto de una posición inicial a una posición de equilibrio realizando iteraciones para minimizar una función la cual evita tomar información a priori de los objetos en su forma o su contorno [6], [7].

2.1.4 Método 4: Segmentación basada en principios o modelos físicos.

La segmentación basada en principios o modelos ópticos y físicos, se fundamenta en el estudio del proceso de generación de la imagen misma, considerando la forma en que se da la reflexión de la luz sobre la superficie de interés de acuerdo a su material, de las características de la cámara y óptica empleada y la forma en que se ubican las luces y se proyectan las sombras. Este método es similar a los otros métodos de segmentación, en cuanto a la implementación de los algoritmos, sólo que difiere de ellos por el hecho de usar explícitamente los modelos de reflexión de las superficies como herramienta para segmentar las imágenes.

El objetivo de este método es segmentar imágenes de acuerdo al entorno real de los objetos, evitando que estén sobreexposados por sombras y sobresaltos en la imagen. Es un método complejo, que involucra un modelamiento del fenómeno de generación de una imagen, ya que las medidas provenientes de una sola superficie pueden variar mucho debido a inter-reflexiones, sombras, ruidos, iluminaciones no uniformes o superficies texturadas [6].

2.1.5 Método 5: Transformación Watershed.

La transformación Watershed es una aproximación matemática por morfología. Considera la magnitud del gradiente de una Imagen, interpretándola de forma similar a una superficie de contornos topográfica. El nivel de gris de los píxeles indicaría entonces la altitud y los píxeles conexos que tengan mayor magnitud de gradiente corresponden a las líneas que delimitan los contornos o watershed, que representan los límites de la región. Por tanto, se intenta dividir la imagen de grises en regiones o cuencas contenidas entre esas líneas de contorno.

La aplicación directa de este método generalmente provoca una sobre-segmentación debido al ruido y otras irregularidades locales del gradiente. Este problema puede ser resuelto empleando marcadores, los cuales son componentes conectados en la imagen el cual se define de acuerdo al problema a resolver [6], [7].

2.1.6 Método 6: Segmentación por clusterización.

Es un proceso en donde un conjunto de datos es reemplazado por un clúster o agrupamiento compacto. El objetivo es intentar

determinar cuáles componentes en el conjunto de datos naturalmente están integrados juntos.

Un clúster puede implementarse de dos maneras:

- **Particionamiento:** hay un grupo de datos que pueden separarse de acuerdo a la noción de asociación dentro del conjunto. Se puede descomponer una imagen en regiones que tengan un color coherente entre sí o la misma textura.
- **Agrupamiento:** en este método se tienen distintos grupos de datos, y se van recolectando en la medida que se van sensando su afinidad para estar juntos.

Segmentación por K-means clusterización. Efectúa una división iterativa de la imagen de interés en regiones por la media y termina cuando se alcanza el criterio de discriminación. Este algoritmo de clusterización busca clasificar objetos (o píxeles) basado en un conjunto de características en K número de clases. La clasificación se realiza minimizando la suma de cuadrados de las distancias entre los objetos y el correspondiente clúster o centroide. Su principal problema es que se debe saber el número de clases a priori [7].

Segmentación por fuzzy C-means clustering. La clusterización por Fuzzy C-Means (FCM), es una técnica en la cual el conjunto de datos es agrupado en K clúster, pero cada dato del conjunto es agregado al cluster con cierto grado de membresía por lógica difusa [5], [6].

Segmentación independiente del modelo por Mean - Shift clustering. Este algoritmo busca el máximo local en una distribución de densidad dada. No asume que los clústeres deban tener forma esférica, ya que se define una ventana en donde se

computa la media de los datos en ella iterativamente hasta que haya convergencia, es un buen método para encontrar características como color, gradientes o texturas [5], [6].

Segmentación probabilística por Expectación - Maximización (EM) clustering. Este algoritmo es usado para estimar la densidad probabilística en un conjunto de datos. El algoritmo se basa en encontrar la máxima probabilidad estimada de los parámetros cuando el modelo de datos depende de algunas variables latentes. En EM, se alternan los pasos de Expectación y Maximización, los cuales se efectúan iterativamente hasta que el resultado converja. El paso de expectación, calcula la probabilidad de incluir las variables latentes como si hubieran sido observadas; y el paso de maximización, calcula la máxima probabilidad estimada o la máxima probabilidad esperada del primer paso. Este proceso se ejecuta iterativamente hasta que converja [5], [6].

Segmentación por particionamiento por grafos y Normalized Cuts-A clusterización. La imagen a ser segmentada puede ser modelada como si fuera un grafo con peso no direccionado. Cada píxel es un nodo en el grafo, y un contorno queda definido por cada par de píxeles. El peso del contorno es cuantificado por el grado de similitud entre los píxeles. La imagen se particiona en conjuntos disjuntos, para remover los contornos que conectan los segmentos débiles en peso y mantener los fuertes. El óptimo particionamiento o corte del grafo se efectúa minimizando los pesos de los contornos a ser removidos. Los algoritmos buscan minimizar el corte normalizado, la cual es la razón del corte para todos los contornos del conjunto [5], [6].

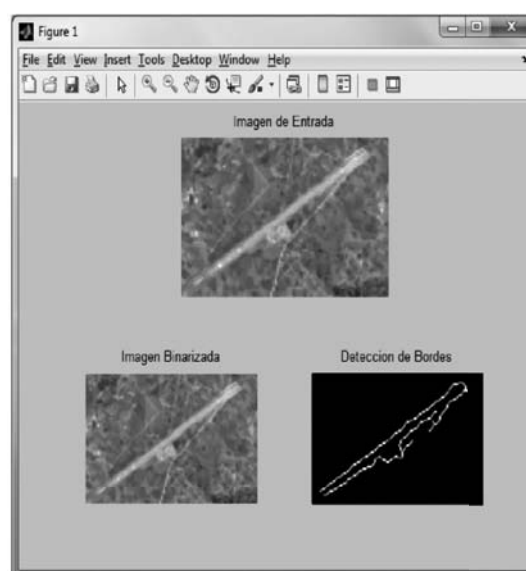
Segmentación por Partition Around Medoids (PAM) clusterización. La técnica PAM minimiza la suma del grado de no similitudes, en lugar de sumar el cuadrado de las distancias euclídeas. El algoritmo primero computa un número de objetos representativos, a los cuales se denomina como medoides o centrotipos. Que pueden definirse un objeto en un cluster el cual tiene un grado de no similitud mínimo respecto a los otros elementos del cluster. Después de encontrar el conjunto de medoides, cada objeto en el conjunto es asignado al medoi-de más cercano en similitud [5], [6].

3. Implementación de la transformada Hough para procesamiento de imágenes satelitales.

Para generar un algoritmo lo suficientemente capaz de detectar formas geométricas básicas, fue necesario realizar un proceso previo de acondicionamiento de la imagen, este proceso implica su transformación a tamaño 100x100, luego es necesaria su binarización y por último la detección de bordes, con lo que el análisis de la transformada de Hough se hace un poco menos complejo. La Figura 1 ilustra claramente los pasos que se desarrollan en esta etapa de acondicionamiento de la imagen.

Una vez realizado el proceso de acondicionamiento de la imagen, es necesario plantear la forma de ejecutar el algoritmo de la Transformada de Hough. Estos algoritmos son específicos respecto a lo que plantean detectar dentro de la imagen, lo que significa que no existe un algoritmo general que determine varias formas al tiempo. En este documento se detallan dos de los algoritmos más utilizados mediante esta

Figura 1. Proceso previo a la ejecución del algoritmo de la transformada, segmentación de la imagen original en dos pasos



técnica como lo son la Transformada de Hough para líneas rectas y para círculos.

3.1 Proceso de filtrado

Es el conjunto de técnicas englobadas dentro del preprocesamiento de imágenes cuyo objetivo fundamental es obtener, a partir de una imagen origen, otra final cuyo resultado sea más adecuado para una aplicación específica mejorando ciertas características de la misma que posibilite efectuar operaciones del procesado sobre ella. [1], [3].

El proceso de filtrado en una imagen de tipo digital busca reducir el ruido, suavizar las texturas o detectar bordes. Existen gran cantidad de métodos para realizar este tipo de procesos; para el caso de la aplicación de la Transformada Hough, se implementó el filtro Canny, debido a su robustez frente al

ruido y la versatilidad de tiempo computacional [8],[9].

3.1.1 Método Canny

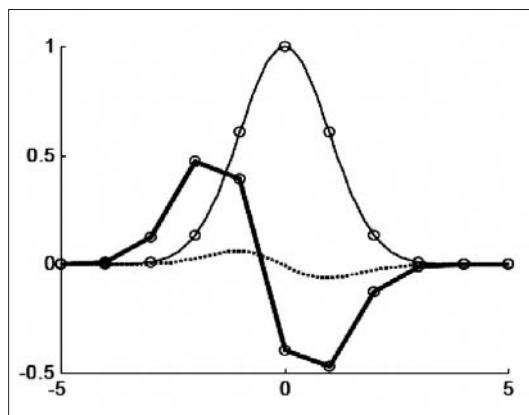
Esta técnica, que se caracteriza por estar optimizada para la detección de bordes diferenciales, consta de tres etapas principales: filtrado, decisión inicial, e histéresis [9], [10].

La primera etapa consiste en un filtrado de convolución de la derivada primera de una función gaussiana normalizada discreta sobre la imagen, realizada en dos direcciones: horizontal y vertical. La función gaussiana posee dos parámetros fundamentales, valor medio m , y desviación típica σ . En este caso, el valor medio es nulo, siendo la ecuación que define el filtro gaussiano de la forma:

$$g(x) = k \cdot \exp\left(-\frac{x^2}{2\sigma^2}\right) \quad (1)$$

donde k se elige de manera que el máximo de $g(x)$ sea la unidad en su versión discreta.

Figura 2. Filtro Óptimo Detector de Borde de Canny: $g(x)$ normalizada continua y discreta para $\sigma = 1$ (línea fina). $dg(x)$ (línea punteada) y $dg[x]$ (línea gruesa) a partir de $g[x]$.



Para la realización del filtro utilizado en la primera etapa, se deriva (1), obteniéndose:

$$g'(x) = \frac{d}{dx} g(x) = -\frac{k}{\sigma^2} x \cdot \exp\left(-\frac{x^2}{2\sigma^2}\right) \quad (2)$$

La versión discreta del filtro se define como $dg[x] = g'(x)$, donde x posee valores desde -5σ hasta 5σ , con diferencias de 1 pixel entre muestras consecutivas. Por razones de cómputo, es preferible derivar directamente la versión discreta de (1), con el fin de obtener el valor k . En la Figura 2. se muestran estas funciones para $\sigma=1$.

El paso siguiente consiste en realizar la convolución de $dg[x]$ sobre la imagen U de tamaño $M \times N$, con $U(i,j) \in [0,1]$, obteniéndose dos nuevas matrices de la forma:

$$dx = \text{abs}(dg * U), \quad dy = \text{abs}(dg' * U) \quad (3)$$

donde el superíndice T indica una operación de trasposición, y $*$ es el operador de convolución discreta. Las matrices dx y dy son imágenes $M \times N$ donde el valor de nivel de gris de un pixel determina la variación o gradiente, en valor absoluto, con los pixeles colindantes, en las direcciones horizontal y vertical, respectivamente. Estas matrices son el punto de partida para la etapa siguiente.

La segunda etapa realiza una primera toma de decisiones sobre los posibles bordes de la imagen, [10], mediante el cálculo de los picos de las imágenes obtenidas en la primera etapa, dx y dy , realizando el siguiente grupo de cálculos:

- Un pixel de U , $U(i,j)$, se considera candidato a borde si $dx(i,j) > dx(i-1,j)$ y $dx(i,j) > dx(i+1,j)$, o bien $dy(i,j) > dy(i,j-1)$ y $dy(i,j) > dy(i,j+1)$, para $i = 1, \dots, M$ y $j = 1, \dots, N$. Los pixeles pertenecientes al borde de la imagen no se contabilizan en esta decisión, y se clasifican como pixeles de fondo.

- Para los píxeles (i,j) que cumplan la condición anterior, se calcula el valor $e(i,j) = (dx2(i,j)+dy2(i,j))/2$. Para el resto de píxeles, $e(i,j) = 0$. Para eliminar efectos de borde, se eliminan los valores de las filas 2, $M-2$ y $M-1$, así como los de las columnas 2, $N-2$ y $N-1$. La matriz $M \times N$ así obtenida, posee los candidatos de borde global (no orientado) de la imagen U .
- Para obtener una imagen con valores de gris en el rango $[0,1]$, se normaliza la imagen e .

La última etapa de procesamiento realiza una optimización de la decisión llevada a cabo en la etapa anterior, mediante la aplicación de una función de histéresis sobre la imagen e . Esta función se basa en la definición de dos umbrales, TL y TH , tales que $TL < TH$. Valores típicos para estos umbrales son 0.1 y 0.5, respectivamente, aunque se recomienda que TH y TL tengan una relación entre 2:1 y 3:1, dependiendo de la relación señal ruido, en el caso de que este valor sea conocido [10].

Esta etapa realiza los cálculos siguientes:

- Un píxel de e , $e(i,j)$, se considera borde definitivo si $e(i,j) \geq TH$.
- Un píxel de e , $e(i,j)$, se considera fondo definitivo si $e(i,j) < TL$.
- Todos los píxeles en un vecindario 3×3 de los píxeles considerados como borde definitivo, $e(k,l)$, se consideran también borde definitivo si $e(k,l) \geq TL$.

La información normalizada no binaria de salida del método de Canny indica la fuerza del borde a partir de un análisis del gradiente global entre píxeles vecinos. Este método es uno de los más robustos contra el ruido (filtro óptimo), en comparación con los métodos de Roberts, Sobel o Prewitt, por ejemplo [9], [10].

3.2 Transformada de Hough para líneas rectas.

Es la versión más simple de la transformada Hough, a continuación se explica de forma muy simplificada el método usado para detectar rectas: Considerando un píxel de coordenadas (x_i, y_i) , seleccionado como elemento que representa un borde; sobre éste pasarán infinitas rectas, el modelo de todas estas posibilidades queda definido por los infinitos valores de a y b .

$$y_i = ax_i + b \quad (4)$$

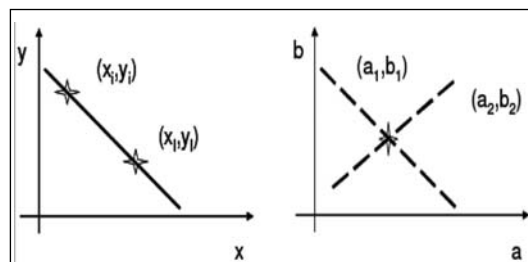
Haciendo la transformación al espacio paramétrico, mostrada en la Figura 3, los valores de a y b quedarán definidos como:

$$b = y_i - ax_i \quad (5)$$

Al variar el parámetro a desde $-\infty$ a $+\infty$ se obtendrán los infinitos valores de b en el espacio paramétrico y obtendremos infinitas rectas. La característica interesante de esta presentación consiste en que, si dos píxeles que pertenezcan a una misma línea son representados en el espacio paramétrico, el modelo de la recta estará definido en la intersección de las dos rectas del espacio paramétrico [11].

Sin embargo, el espacio paramétrico elegido no es el más correcto, ya que los rangos

Figura 3. Ejemplo de conversión hacia el espacio paramétrico



dinámicos de a y b no están limitados. En cambio, si se hace una representación en coordenadas polares, mostrada en la figura 4, el ángulo de la normal de la recta, θ , está limitado al rango de $[0, \pi]$. Ecuación de la recta en forma polar (6):

$$\rho = x \cos \theta + y \sin \theta \quad (6)$$

Por tanto hay que realizar una transformación entre el plano imagen (coordenadas x - y) y el plano o espacio paramétrico $[\rho, \theta]$, discretizado en una serie de celdas denominadas celdas acumuladoras. Esta discretización se realiza sobre los intervalos $[\rho_{min}, \rho_{max}]$ y $[\theta_{min}, \theta_{max}]$.

Figura 4. Espacio paramétrico en coordenadas polares

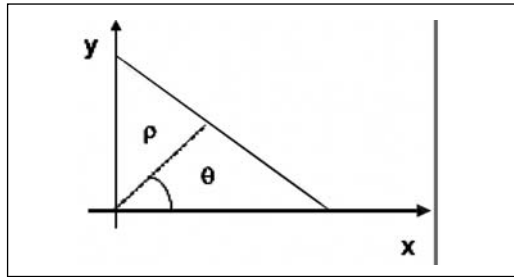


Figura 5. Código con que se realiza el recorrido de la imagen de entrada.

```

for(i=1:1:col)
    for(j=1:1:fil)
        if(img_filter(j,i)>=1)
            for(theta_gra=1:1:360)
                theta_rad=theta_gra*(pi/180);
                rho=round(i*cos(theta_rad)+j*sin(theta_rad));

                if (rho<1)
                    rho=1;
                    ma_hough(rho,theta_gra)=ma_hough(rho,theta_gra);
                else
                    ma_hough(rho,theta_gra)=ma_hough(rho,theta_gra)+1;
                end;
            end;
        end;
    end;
end;

```

Esta discretización del espacio también es conocida como matriz de Hough, en ella se graficará un grupo de líneas por cada pixel que se recorra de la imagen, y agregará un punto a la celda de acumulación correspondiente. Donde se evidencien más puntos de cruce, en esta coordenada existe una línea.

La siguiente, es la descripción del código en MATLAB®, ver Figura 5, con el cual se realiza el recorrido en la imagen analizada y paralelamente se le asigna un punto a cada celda de acumulación en donde se reconozca la existencia de una línea [12]:

La Figura 6 muestra claramente la matriz de acumulación en una imagen con una línea, mientras que la Tabla 1 muestra en detalle un fragmento de la matriz de acumulación y los valores de cada celda de acumulación luego del barrido de la imagen.

A continuación se procederá a realizar el proceso inverso de la Transformada de Hough, transformando el plano parametrizado a los

Figura 6. Ejemplo de Acumulación en la matriz Hough

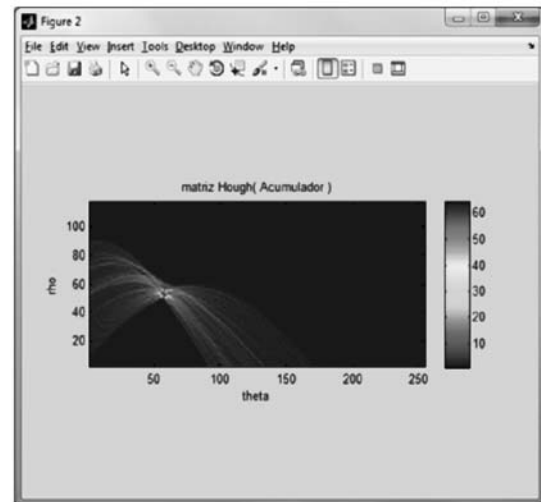


Tabla 1. Extracción de celdas de acumulación de la matriz Hough.

9	6	2	0	0	0	2	8	21	11	9	17	16	18	14	13	11	10	8
9	12	14	13	7	2	13	28	15	16	19	21	19	18	13	11	9	9	8
10	10	12	16	28	55	61	37	26	20	26	20	15	11	10	8	8	5	4
10	11	14	21	36	26	8	9	15	30	22	15	15	9	9	11	9	11	12
17	21	20	25	14	0	0	2	23	14	11	10	9	13	15	12	12	11	9
19	18	26	20	5	6	9	28	10	8	7	10	12	12	10	13	14	15	17
10	19	18	13	19	33	52	30	27	20	17	17	17	16	18	15	14	14	12
21	16	8	12	26	27	12	17	17	20	24	21	18	14	9	9	10	8	9
9	9	18	21	18	10	6	4	4	7	8	10	9	13	16	14	11	8	6
14	14	11	11	7	6	6	5	10	8	5	6	7	5	4	5	5	6	5
11	9	9	10	7	5	5	10	8	8	7	3	2	3	1	0	0	0	1
6	6	12	8	6	13	16	10	5	2	2	3	1	0	0	0	0	0	0
8	17	13	13	15	8	2	2	2	3	1	0	0	0	0	0	0	0	0
21	15	7	8	4	3	3	2	1	0	0	0	0	0	0	0	0	0	0

términos de $[x,y]$, con esto se logran recrear los grupos de líneas encontrados en una imagen en blanco en el plano $[x,y]$ llamada imagen Hough. Para realizar este procedimiento, despejamos y de la ecuación (6).

$$y = \frac{\rho}{\sin\theta} - \frac{x\cos\theta}{\sin\theta} \tag{7}$$

Donde,

$$b = \frac{\rho}{\sin\theta} \tag{8}$$

$$a = -\frac{\cos\theta}{\sin\theta} \tag{9}$$

con estos valores podemos utilizar la ecuación general de la recta (1) para graficar nuestros grupos de líneas pixel por pixel recorriendo la matriz imagen Hough, además, para fines comparativos, también los pintamos sobre una copia de la imagen original denominada imagen de salida. Para optimizar el algoritmo y evitar la intromisión de líneas que no corresponden a las relacionadas en la imagen, es importante que el usuario final, partiendo de un conocimiento previo, ajuste manualmente el valor de inicio

a partir del cual serán graficadas las líneas, teniendo en cuenta las coordenadas en las celdas de acumulación.

Las Figuras 7 y 8 permiten detallar la reconstrucción de un grupo de líneas según la Transformada Hough.

Figura 7. Reconstrucción de la Imagen según el algoritmo Hough.

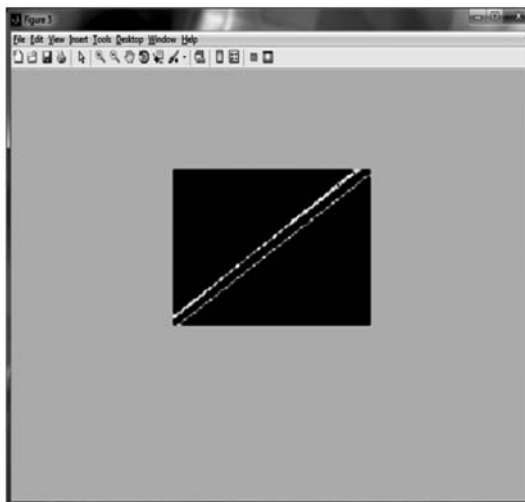
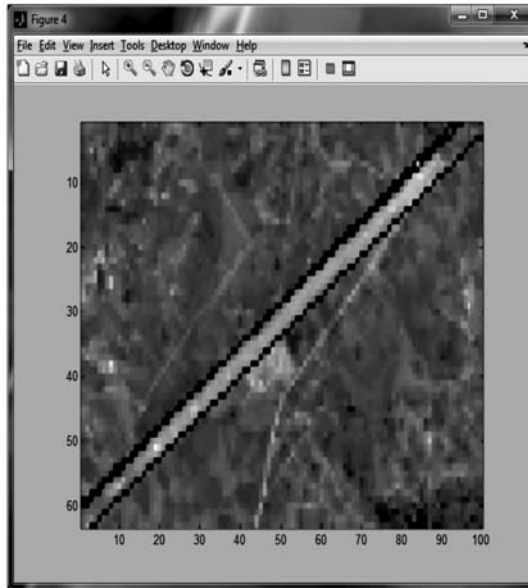


Figura 8. Relación entre la imagen de entrada y el resultado del algoritmo Hough



3.3 Transformada de Hough para círculos

El proceso para el desarrollo del algoritmo para la detección de círculos en la transformada Hough sigue el mismo proceso que para las líneas, con la variable que para este caso, el espacio paramétrico será tridimensional [13], correspondiente a la posición del centro del círculo, (x_0, y_0) , y al radio, r (10):

$$(x_i - x_0)^2 + (y_i - y_0)^2 = r^2 \quad (10)$$

Para pasar al espacio paramétrico es necesario aplicar la ecuación del círculo en este tipo de coordenadas, y queda de la siguiente forma:

$$x_i = a + r \cos \theta \quad (11)$$

$$y_i = b + r \sin \theta \quad (12)$$

Donde a y b corresponden al centro del círculo y r al radio (11) (12).

Las votaciones de cada celda del espacio paramétrico se realizan variando la posición del centro del círculo, (a, b) , para cada píxel etiquetado, (x_i, y_i) , obteniendo los distintos valores del radio r . Por tanto, entrado en la rejilla de acumulación por posición del centro del círculo y del radio calculado se procede a depositar los votos correspondientes. Los círculos seleccionados son aquellos que superen un cierto umbral de votos recibidos [13], [14], [15].

La siguiente, es la descripción del código Desarrollado, mostrado en la figura 9, con el que se implementa la transformada para círculos; a continuación se muestra el proceso de recorrido de la imagen y acumulación de puntos en las celdas de la matriz Hough de la misma manera que para las líneas rectas. El radio máximo corresponde al valor máximo desde el origen hasta el círculo más lejano que se pudiera construir dentro de la imagen con el fin de evitar desbordamientos en la matriz.

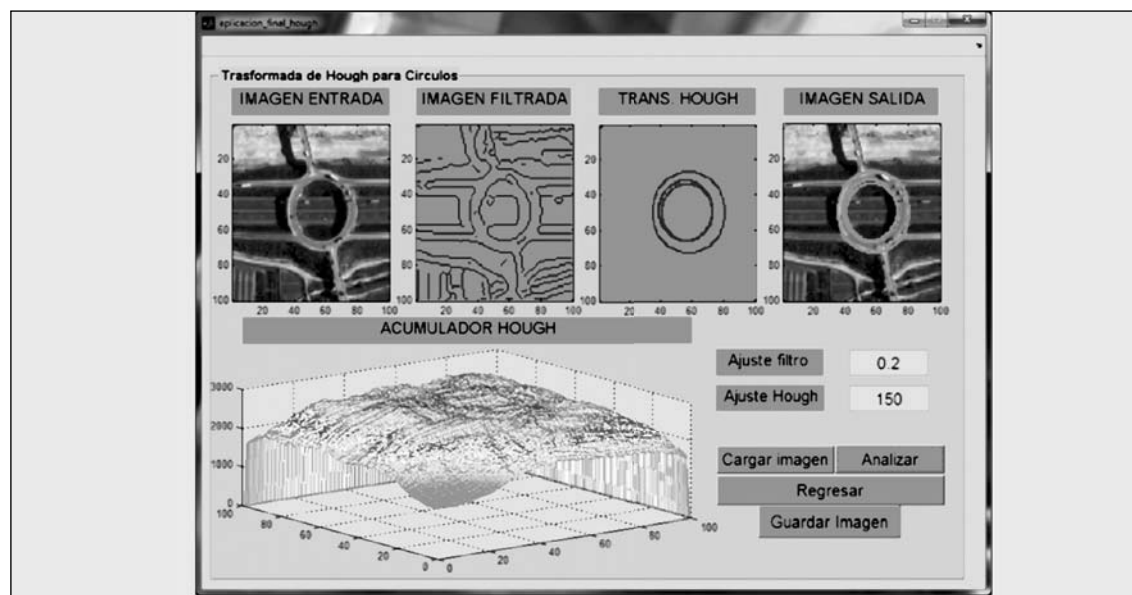
Figura 9. Código con que se realiza el recorrido de la imagen de entrada.

```

for(radius=1:1:radius_max)
    for(i=1:1:col)
        for(j=1:1:fil)
            if(img_filter(j,i)>=1)
                for(theta_gra=1:1:360)
                    theta_rad=theta_gra*(pi/180);%conversion a radianes
                    x0=round(i-(radius)*cos(theta_rad)); % x centro
                    y0=round(j-(radius)*sin(theta_rad)); % y centro

                    if((y0 > 0) && (y0 < ymax) && (x0 > 0) && (x0 < xmax))
                        ma_hough(x0,y0,radius)=ma_hough(x0,y0,radius)+1;
                        ma_hough1(x0,y0)=ma_hough1(x0,y0)+1;
                    end;
                end;
            end;
        end;
    end;
end;
end;

```

Figura 10. Desarrollo de la Transformada de Hough para círculos

A este punto se ha comprobado pixel por pixel la imagen creando círculos de varios radios, por cada uno que concuerde con un pixel positivo, se aumenta en uno la matriz de acumulación en la posición $[x,y,r]$. El siguiente paso es realizar el proceso inverso y recrear los círculos hallados en el proceso anterior en la imagen Hough y también en la imagen de salida.

La Figura 10 muestra el desarrollo de la Transformada Hough para círculos, se detallan exactamente los mismos parámetros que en el caso de las líneas, solo que denotados dentro de una interfaz gráfica que permite realizar el proceso de manera más sencilla, aunque más lenta, contrario al método de compilación simple del código que es mucho más complejo pero más ágil.

Se puede observar en la gráfica el umbral de sensibilidad o ajuste Hough, que corresponde al valor de la matriz de acumulación que

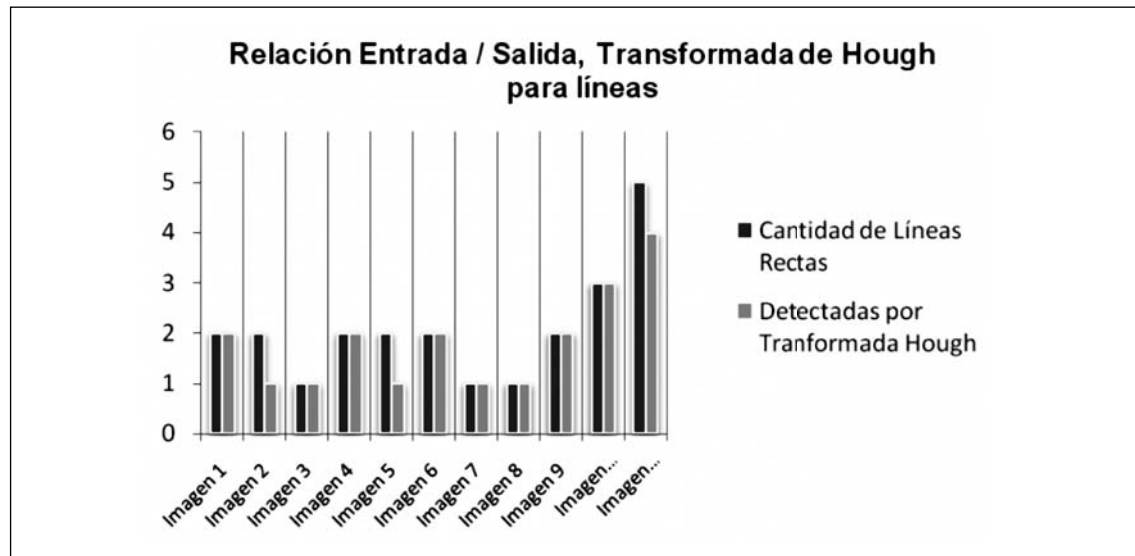
se grafica, para este caso, solo las celdas de acumulación superiores a 150 son los pixeles de las coordenadas que se dibujan.

Las ventajas de esta detección de círculos, provienen de las características inherentes de las transformadas de Hough, la inmunidad al ruido y que es capaz de detectar círculos, aunque exista oclusión del mismo. Por otro lado, en este tipo de curvas, al ser cerradas, no hay problemas del carácter infinito de las líneas rectas. Los círculos están definidos en la imagen, sin ninguna expansión.

4. Análisis de Resultados

Con el desarrollo de los algoritmos de la Transformada Hough para línea rectas y círculos en imágenes satelitales, se ha logrado detectar gran cantidad de patrones que mediante otros métodos podrían haber sido imperceptibles, aunque también se han generado ciertos valores que corresponden a

Figura 11. Relación entre la cantidad de líneas existentes en las imágenes de entrada y las encontradas mediante el algoritmo Hough para líneas rectas.



ruido o errores en el procesamiento. La Figura 11 muestra una comparación con una secuencia de imágenes realizada durante los procesos de evaluación.

Se muestra claramente que en ocho de las imágenes de entrada, el algoritmo logró detectar la misma cantidad de líneas que las que se podían percibir a simple vista, mientras que para las otras tres no.

La gran mayoría de las muestras analizadas mostraron resultados óptimos, sin embargo existe un parámetro que el algoritmo es incapaz de regular por sí mismo y es el ajuste de sensibilidad que determina el umbral que filtrará cada una de las celdas de acumulación con el fin de graficar un grupo de líneas o círculos, según sea el caso.

5. Conclusiones

- Una limitante en la Transformada Hough consiste en que los algoritmos son específicos respecto a lo que plantean detec-

tar dentro de la imagen, lo que significa que no existe un algoritmo general que determine varias formas al tiempo.

- Se comprobó la efectividad de la Transformada de Hough para el reconocimiento de líneas rectas y círculos y su alta inmunidad al ruido, lo que convierte los algoritmos en medios fiables para el análisis de las imágenes.
- Se lograron obtener una serie de algoritmos capaces de recrear el modelo de la transformación al espacio paramétrico, aunque su desarrollo implica un gran número de procesos.
- El mayor inconveniente de la Transformada Hough es su alto costo computacional. Una forma de ahorro en el cálculo en el algoritmo de detección de círculos es fijar el valor del radio del círculo. Si se conoce el radio, junto con la información de la orientación del gradiente del píxel etiquetado, (x_p, y_p) , las posibles coordenadas del centro del círculo darán una nube de puntos limitados.

- Debido a que estos algoritmos ejercen procesos netamente estadísticos, convierten la Transformada Hough en un robusto método de sustracción de primitivas en el análisis de imágenes, que no solo pueden ser satelitales, sino que también pueden ser de cualquier tipo, pues estos han sido diseñados de una manera generalizada para tal fin.
- Se recomienda implementar la Transformada de Hough para futuros análisis dentro del grupo de investigación como parte integral dentro de los procesos de segmentación que sean convenientes en los avances de visión artificial, gracias a sus potentes características.

Referencias Bibliográficas

- upm.es/acceso_profesores/asignaturas/tdi/tdi/transformadas/pdf/hough.pdf
- [1] C. Platero, "Apuntes de Visión Artificial", pp. 149-154. Consultado Octubre de 2008. Disponible en: <http://www.elai.upm.es:8009/spain/Asignaturas/Robotica/ApuntesVA/cap5VASegmentacion.pdf>
 - [2] G. Molinero, "Segmentación de imágenes en color basada en el crecimiento de regiones" Biblioteca de la Universidad de Sevilla. Consultado Noviembre 2010. Disponible en <http://bibing.us.es/proyectos/abreproy/11875/direccion/Proyecto+Fin+de+Carrera%252F>
 - [3] Universidad de Jaén, "Transformada De Hough (HGT)". Consultado Octubre de 2008. Disponible en <http://www.diac>
 - [4] R. E. Woods; R. C. González; S. L. Eddins, "Digital image processing using Matlab" Gonzalez, Woods, & Eddins, 2009
 - [5] T. Acharya; A. K. Ray, "Image processing: principles and applications" John Wiley & Sons, 2005
 - [6] R. C. González; P. Wintz, "Procesamiento digital de imágenes" Addison-Wesley, 1996
 - [7] R. C. González; R. E. Woods, "Digital Image Processing", Addison-Wesley, 1993
 - [8] V. F. Leavers. "Shape Detection in Computer Vision Using the Hough Transform". Oct. 2010.
 - [9] M. A Jaramillo; J. A. Fernández; E. Martínez de Salazar, "Implementación del Detector de Bordes de Canny sobre Redes Neuronales Celulares" Universidad de Extremadura, 2010
 - [10] J.F. Canny, "A Computational Approach to Edge Detection", IEEE PAMI, 8(6), 1986, pp. 679-698.
 - [11] E. Ospina, "Implementación de la Transformada de Hough para la detección de líneas para un sistema de visión de bajo nivel". Consultado Febrero de 2009. Disponible en <http://pci.unalmzl.edu.co/Tesis/emmanueljuanpablo.pdf>