

CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS
AVANZADOS DEL INSTITUTO POLITÉCNICO
NACIONAL

Unidad Zacatenco

Departamento de Control Automático

“Circularidad en imágenes digitales”

Tesis

Que presenta

Guillermo García Jiménez

para obtener el Grado de

Maestro en Ciencias

en la Especialidad de

Control Automático

Directora de Tesis

Dra. Petra Wiederhold Grauert

Ciudad de México

Noviembre, 2017

Dedicatoria

Con dedicatoria a mis padres, por su gran amor y su incondicional apoyo. Siempre me han guiado, me han orientado para ser una persona de bien y correcta.

Agradecimientos

Quiero agradecer al CONACYT por brindarme una beca para poder sustentar mis gastos a través de todo este proceso de aprendizaje y desarrollo.

También quiero agradecer al Departamento de Control Automático y al CINVESTAV por dejarme formar su comunidad estudiantil y permitirme realizar la maestría. Por brindarme una gran calidad educativa y ofrecerme las mejores instalaciones. También, agradezco a todos los doctores que pusieron esfuerzo y me brindaron su conocimiento.

Agradezco a mi directora de tesis, la Dra. Petra Wiederhold Grauert por brindarme la oportunidad de trabajar con ella. Por su compromiso, paciencia, dedicación y su gran ayuda para poder llevar acabo esta tesis. Además, le agradezco mucho por sus enseñanzas, consejos y su incondicional apoyo.

Le agradezco a los sinodales, la Dra. Martha Rzedowski Calderón y al Dr. Wen Yu Liu, por formar parte de mi jurado, por tomarse el tiempo de revisar mi tesis y sugerirme correcciones para mejorar el trabajo final de tesis.

Le agradezco a toda mi familia, a mi madre Maria Guadalupe Jiménez Hernández; a mi padre, Guillermo García Rodríguez; y a mi padrastro, Julio César López Aguilar, que sin su apoyo, amor y orientación, no habría podido haber llegado a donde estoy ahora. Y a mis hermanas, Amayraní y Estrella por su amor y apoyo.

Le agradezco a mi novia, Diana Lissette Flores Rivas, por su apoyo a lo largo de este camino, por su amor y paciencia, además también por impulsarme siempre a ser mejor.

Le agradezco a todos mis amigos y compañeros con los que compartí esta etapa de mi vida, por hacer más ameno el recorrido, por su ayuda, también por los buenos y difíciles momentos que pasamos en el transcurso de toda la maestría.

Resumen

Esta tesis presenta las bases matemáticas y algunos métodos de la detección de círculos y elipses y de arcos circulares en imágenes digitales bidimensionales. La detección y el reconocimiento de formas circulares y elípticas es de gran importancia en muchas áreas de aplicación del análisis de imágenes, por ejemplo, en la ingeniería, incluyendo la robótica, en medicina y biología, y en la astronomía.

El enfoque principal de la tesis son métodos que usan la acumulación de evidencia en espacios de parámetros, basados en la transformada de Hough. La tesis presenta en detalle las bases y varias versiones de este método, incluyendo el aprovechamiento de la imagen gradiente y de propiedades geométricas para bajar la dimensión del espacio de parámetros o para descomponerlo. De manera breve es presentado también el enfoque de ajuste de curvas, donde la adaptación de círculos o elipses a un conjunto dado de puntos es logrado al resolver un problema de minimización de errores. La gran ventaja de estos métodos y de los que se relacionan con la transformada de Hough, consiste en su utilidad para tratar datos de entrada desorganizados, es decir, donde no se requiere que los puntos dados sean ordenados o que pertenezcan a un contorno o curva digital. El problema de detección de círculo es muy distinto cuando los puntos dados forman una curva digital, es decir, siguen un orden de trazado, y forman un camino con ciertas propiedades en una gráfica vecina que proporciona conectividad al plano discreto. Entonces, para detectar o decidir si esta curva corresponde a un círculo, o para segmentarla en arcos circulares, se conocen métodos eficientes que aprovechan esta conectividad y orden. La tesis presenta uno de estos algoritmos, el cual se basa en propiedades de los números enteros cuadrados y conceptos de la geometría digital, y logra una eficiencia impresionante tanto para generar círculos digitales como para reconocerlos o para segmentar una curva digital en arcos circulares. Varios algoritmos de detección de círculos, principalmente del método de Hough, fueron programados. La tesis presenta experimentos que fueron efectuados con diversos tipos de imágenes, por ejemplo, dibujos con varios círculos de diferentes radios, incluyendo concéntricos, y una imagen real que tuvo que ser preprocesada primero. Se experimentó también con imágenes de nubes de puntos, donde los píxeles se encuentran muy dispersos y los círculos están presentes no por fragmentos de curvas que se forman a partir de puntos de borde, sino por una densidad más alta de la nube alrededor de la circunferencia a detectar.

Abstract

This thesis presents the mathematical bases and some methods for the detection of circles, ellipses, and circular arcs, in two dimensional digital images. Detection and recognition of circular and elliptical forms are important in many application areas of digital image analysis, for example, in engineering, including robotics, in medicine and biology, and in astronomy. The thesis is mainly focused on methods which use evidence accumulation in parameter spaces, based on the Hough transform. The document presents in detail the bases and various versions of this method, including parameter space decomposition based on the gradient image or using geometrical properties. Much more briefly, the curve fitting approach is presented, where the adaptation of a curve to the given point set is achieved by solving an error minimization problem. The great advantage of these methods, as well as of those based on the Hough transform, consists in the fact that they can be applied to unorganized input data, that is, where the given points are not required to be ordered or to belong to some contour or digital curve. The problem of circle detection looks quite distinct when the input points form a digital curve, that is, when they follow some tracing order, when they belong to some trajectory with special properties, within a neighboring graph which provides connectivity to the discrete plane. In this case, in order to detect a circle, or to decide whether the given curve corresponds to a circle, or to segment the curve into circular arcs, there are known methods which use connectivity and order. The thesis presents one of these algorithms, which is based on properties of the integer square numbers and concepts from digital geometry. It achieves an amazing efficiency for generating digital circles, and also for circle detection and curve segmentation into circular arcs. Several algorithms of circle detection, mainly of the Hough method, were implemented. The thesis describes experiments performed with images of diverse kinds, for instance, circle drawings containing circles of various radius, including concentric circles, and a real image where some preprocessing was needed. Special experiments were performed with 2D images containing point clouds, where the pixels are more distant from each other, and where the circles make themselves evident by a higher density of the cloud rather than by curve fragments connecting edge points.

Índice general

Resumen	vii
Abstract	ix
Introducción	1
Motivación	1
Justificación del tema de tesis y antecedentes	1
Objetivos y metas de la tesis	5
Resultados y aportaciones de la tesis	5
Estructura de tesis	6
1. Imagen digital, imagen gradiente y puntos de borde	9
1.1. Imagen digital y plano discreto	9
1.2. Conectividad en el plano discreto	10
1.3. Mapeos de discretización	11
1.4. Líneas rectas y círculos digitales	12
1.5. Imagen de gradiente y puntos de borde	17
2. Circunferencias y elipses en el plano euclidiano	21
2.1. Círculo o circunferencia	21
2.1.1. Definición geométrica	21
2.1.2. Representación algebraica	22
2.1.3. Representación paramétrica	23
2.2. Elipse	24
2.2.1. Definición geométrica	24
2.2.2. Representación algebraica	25
2.2.3. Representación paramétrica	27
3. Detección de círculos y elipses mediante acumulación de evidencia en espacios de parámetros	29
3.1. Método general de la transformada de Hough	29
3.2. Detección de líneas mediante la transformada de Hough	30
3.3. Detección de círculos	33

3.3.1.	Método general (Hough 3D) para círculos	33
3.3.2.	Método para radio conocido (Hough 2D)	37
3.3.3.	Método para centro conocido (Hough 1D)	38
3.3.4.	Método práctico para un rango conocido de radios	40
3.3.5.	Variantes para conocimiento sobre el centro	41
3.3.6.	Métodos para la reducción de la dimensión o para la descomposición del espacio de Hough	45
3.4.	Detección de elipses	51
3.4.1.	Aplicación de la transformada de Hough para la detección de elipses .	51
3.4.2.	Método geométrico con dirección de gradiente	54
4.	Circularidad digital	63
4.1.	Circularidad digital versus detección de círculos basada en geometría euclidiana	63
4.2.	Definición de circularidad digital	64
4.3.	Caracterización y detección de circularidad digital	66
5.	Ajuste de círculos a conjuntos de puntos	73
5.1.	Ajuste de una curva a un conjunto de puntos	73
5.2.	Ajuste algebraico del círculo	75
5.3.	Ajuste geométrico del círculo	76
5.4.	Determinación del centro del círculo a partir de tripletas de puntos	78
6.	Experimentos	81
6.1.	Programas de cuatro métodos para la detección de circunferencias en imágenes digitales	81
6.2.	Detección de circunferencias continuas	84
6.3.	Comparación con otras implementaciones del método de Hough en MATLAB	91
6.4.	Imágenes artificiales de nubes de puntos	94
6.5.	Tratamiento de una imagen astronómica	99
6.6.	Programa para ilustrar la Versión 1 del Algoritmo de Bhowmick	103
	Conclusiones	107
	Bibliografía	109

Índice de figuras

1.1. (a) Código de Freeman. (b) El código de cadena del segmento de línea, iniciando con el píxel izquierdo de color gris, es $F = (07007)$, su código de longitudes de corridas es $S = (2, 3, 1)$	14
1.2. Discretización de la intersección con la retícula sobre el primer octante para círculos de radios $r = 1, \dots, 15$	16
1.3. Discretización de intersección con la retícula en el primer octante de una circunferencia de radio $r = 13$	17
2.1. Características del círculo: (a) ángulo central, (b) ángulo inscrito, (c) recta tangente, (d) recta secante.	22
2.2. Representaciones del círculo.	23
2.3. Definición de una elipse.	24
2.4. Propiedades de una elipse.	25
2.5. Deducción de la ecuación algebraica de la elipse.	27
2.6. Una elipse con sus círculos que la inscriben y la suscriben.	27
3.1. Representación paramétrica de una línea recta: (a) en MATLAB, (b) en un sistema de coordenadas estándar.	31
3.2. Posibles líneas rectas que pasan por un punto (x, y) : (a) en el espacio imagen, (b) su representación en el espacio de parámetros.	31
3.3. Varios puntos de una misma línea (a) generan, en el espacio de parámetros (b), un conjunto de curvas sinusoidales, llamado en la literatura mariposa, con un solo punto de intersección.	32
3.4. Arreglo y matriz acumulador Ac del espacio de Hough.	33
3.5. Representación de un círculo: (a) cuando sus tres parámetros (h, k, r) son conocidos, y, (b) en el espacio de Hough 3D.	34
3.6. Un punto en el plano imagen y su representación en el espacio de Hough 3D, vea texto.	35
3.7. División del espacio de Hough 3D en celdas correspondientes a los elementos de la matriz acumulador Ac	36
3.8. Tres puntos en la imagen sobre una circunferencia (a), y los conos correspondientes en el espacio de Hough (b), vea texto.	37

3.9.	Ejemplo de varios puntos candidatos pertenecientes a la misma circunferencia en la imagen (a), y los círculos generados por ellos en el plano de Hough (b). El punto de intersección de todos estos círculos determina las coordenadas del centro de la circunferencia buscada en la imagen.	38
3.10.	El punto p sobre la circunferencia en (a) es mapeado al espacio de Hough 3D (b), donde es representado por un punto del segmento de recta correspondiente a valores r en el intervalo $[0, r_{\text{máx}}]$	39
3.11.	Ejemplo de varios puntos candidatos de la misma circunferencia, con su mapeo al espacio de Hough, vea texto.	41
3.12.	Representación de dos puntos en el espacio de Hough. La parte sombreada se desprecia y solamente se considera parte en blanco; (a) espacio de Hough 2D sólo se considera $h \in [h_{\text{mín}}, h_{\text{máx}}]$, (b) los conos generados en el espacio de Hough 3D únicamente se valoran los puntos dentro de los intervalos $[h_{\text{mín}}, h_{\text{máx}}]$ y $[k_{\text{mín}}, k_{\text{máx}}]$	42
3.13.	Restricción del espacio de Hough por intervalos, vea texto.	43
3.14.	Representación de un punto del espacio imagen al plano de Hough (h, r) . . .	44
3.15.	Representación de un punto del espacio imagen al plano de Hough (k, r) . . .	45
3.16.	Representación gráfica del método basado en la magnitud y la dirección del gradiente	47
3.17.	Representación gráfica del método de dos puntos finales de una cuerda. . . .	49
3.18.	Componentes de los semiejes a, b de una elipse.	52
3.19.	Una elipse y los elementos geométricos para el uso del método geométrico con dirección de gradiente.	55
3.20.	Ángulos importantes en la elipse.	55
3.21.	Elipse rotada por un ángulo θ	56
3.22.	Relación entre los vértices v_1, v_2, v_3, v_4 de la elipse y sus parámetros θ, h, k, a, b . .	59
4.1.	Discretización de intersección con la retícula de circunferencias euclidianas con radios $r = 1, \dots, 15$	65
4.2.	Ejemplo de curva digital.	69
4.3.	Curva digital con siete corridas ($k = 0, \dots, 6$) horizontales.	71
4.4.	Tabla de radios del segmento $S = \{8, 5, 4, 2, 4, 3, 2\}$	71
5.1.	Tres puntos sobre una circunferencia forman un triángulo cuyo circuncentro coincide con el centro del círculo.	79
6.1.	Interfaz gráfica.	82
6.2.	Imágenes de entrada, a ser analizadas por los cuatro métodos H3D, 2P, DG, 4P. .	84
6.3.	Resultados de las matrices acumuladoras correspondientes a los métodos H3D, DG, 2P y 4P, para la imagen (a) de la figura 6.2.	85
6.4.	Círculos detectados en la imagen (a) de la Fig. 6.2.	86
6.5.	Resultados de las matrices acumuladoras correspondientes a los métodos H3D, DG, 2P y 4P para la imagen (b) de la figura 6.2.	87

6.6. Círculos detectados en la imagen (b) de la figura 6.2.	88
6.7. Resultados de las matrices acumuladoras correspondientes a los métodos H3D, DG, 2P y 4P para la imagen (c) de figura 6.2.	89
6.8. Resultados de la imagen (c).	89
6.9. Resultados de círculos detectados en la imagen (d) de figura 6.2.	90
6.10. Matriz acumulador y resultado de circunferencias encontradas con el método 2P3D.	91
6.11. Comparación de resultados del ejemplo (b) con otros programas en MATLAB.	92
6.12. Comparación de resultados del ejemplo (c) con otros programas en MATLAB.	92
6.13. Imagen con circunferencias concéntricas.	93
6.14. Resultados de detección de círculos concéntricos.	93
6.15. Nubes de puntos.	94
6.16. Imágenes con nubes de puntos.	95
6.17. Ejemplo (a).	95
6.18. Ejemplo (b) con resultados no idóneos.	96
6.19. Ejemplo (b).	97
6.20. Ejemplo (c).	97
6.21. Ejemplo (d).	98
6.22. Imágenes preprocesadas y resultados.	99
6.23. Imagen simulada astronómica (vea texto).	100
6.24. Resultados cuando se exige un 75 % de puntos sobre las circunferencias.	101
6.25. Resultados cuando se considera un 60 % de puntos sobre las circunferencias.	101
6.26. Resultados cuando se considera un 50 % de puntos sobre las circunferencias.	102
6.27. Interfaz gráfica para la Version 1 del Algoritmo de Bhowmick.	103
6.28. Segmento de curva digital dado por su código de longitudes de corridas $S = \{8\ 7\ 6\ 5\ 5\ 4\ 2\}$. (a) Representación de la curva digital S y de los círculos digitales correspondientes a las partes circulares de S . (b) Tabla de radios asociados a las corridas.	104
6.29. Segmentos de la curva de $S = \{8\ 7\ 6\ 5\ 5\ 4\ 2\}$, correspondientes a sus arcos circulares.	105

Introducción

Motivación

Esta tesis se dedica a estudiar algunos métodos de detección de círculos y elipses y de arcos circulares en imágenes digitales bidimensionales. La importancia de eso en diversas áreas de aplicación del procesamiento digital de imágenes y del reconocimiento de patrones resulta evidente de la presencia de formas circulares y elípticas en muchas imágenes. Contornos circulares o elípticos, importantes de ser detectados o reconocidos, pueden pertenecer, por ejemplo, a piezas industriales presentadas en dibujos técnicos o en fotografías de escenas en procesos de manufactura o de supervisión de calidad o de separación de piezas. También, en el entorno de trabajo de un robot, puede ser esencialmente importante detectar o reconocer formas circulares, sea para que el robot pueda determinar su ubicación y orientación en este entorno, o para hacer posible que cumpla ciertos objetivos o tareas. Para aplicaciones médicas o en investigaciones biológicas, resulta importante que muchos tipos de células en imágenes microscópicas tienen formas elípticas, lo cual permite, por ejemplo, distinguirlas de células enfermas o anormales o de otros objetos. Formas circulares y elípticas son de interés también en imágenes astronómicas conteniendo galaxias y otras estructuras cosmológicas.

Los métodos estudiados en este trabajo de tesis son esencialmente útiles en el análisis de imágenes digitales y en el reconocimiento de formas. Éstos son partes del área de la visión artificial y robótica, cuyos aspectos ingenieriles, matemáticos y computacionales han sido cultivados en el Departamento de Control Automático del CINVESTAV-IPN desde hace muchos años.

El modelado y la detección de formas y propiedades circulares en imágenes digitales son relacionados con la geometría euclidiana, la geometría discreta y digital, la teoría de números, las gráficas y la teoría de palabras. Entonces, a parte de la motivación por la importancia práctica de describir y detectar circularidad en imágenes digitales, esta tesis se motiva también por la curiosidad de aprender sobre modelos y métodos de detección de circularidad, en el contexto de las áreas matemáticas mencionadas.

Justificación del tema de tesis y antecedentes

La detección de los contornos de objetos en una imagen digital es un aspecto básico en el procesamiento de imágenes y es útil para separar los objetos con respecto al fondo de la

imagen. Un contorno de objeto representa su frontera, y, en una imagen digital, comúnmente es generado a partir de puntos de discontinuidad llamados puntos de borde. En un punto de borde, la imagen cambia dramáticamente su valor de gris o su color. El aislamiento de objetos de interés con respecto al fondo de la imagen a través de los contornos es uno de los enfoques principales de la segmentación de imágenes, llamado *segmentación basada en bordes*. Otro enfoque muy importante es el de la *segmentación basada en regiones*, donde mediante un proceso, comúnmente iterativo, de descomposición y unificación de partes de la imagen que satisfacen ciertos criterios de uniformidad, todas las regiones de interés en la imagen son identificadas. El método más sencillo de este enfoque consiste en la umbralización, separando la imagen entera en dos partes, una de los puntos cuyo valor sobrepasa cierto umbral predefinido, la otra siendo el conjunto de los puntos restantes de la imagen. Más generalmente, pueden ser usados varios umbrales que definen intervalos de valores, por ejemplo de gris, para generar una descomposición de la imagen en regiones, donde los puntos de cada una tienen valores dentro de uno de estos intervalos. Cuando se tiene conocimiento de los valores del fondo de la imagen, se puede utilizar umbralización combinada con la sustracción del fondo, método que permite retirar el fondo para dejar en primer plano los objetos relevantes y así lograr su aislamiento con respecto al fondo.

En muchas aplicaciones del procesamiento y análisis de imágenes digitales, los objetos de interés pueden ser modelados por figuras geométricas simples o son compuestos de partes de tales figuras. Eso hace la búsqueda y la identificación de figuras en imágenes interesante e importante. La detección de figuras se relaciona con la búsqueda de objetos de cierto tipo. Para tratar muchos problemas en la práctica, es útil detectar líneas rectas, círculos o elipses, puesto que, éstos sirven como modelos de los objetos de interés presentes en la imagen. Un ejemplo es la búsqueda de árboles para determinar el estado forestal de una región boscosa, ésta puede ser realizada asociando círculos o elipses a los tallos de árbol residual [26]. La detección de figuras también resulta de gran utilidad en la medicina, para encontrar tumores o malformaciones [17], o en la detección de conos sobre la retina [2]. En el área ingenieril, por ejemplo, en la interferometría Speckle [8], se usa una técnica para la extracción de fases que busca ajustar un círculo en una nube de puntos generada por una señal, permitiendo encontrar una relación entre la abscisa del centro del círculo y la determinación del fondo local, así como, el radio con la profundidad de modulación; a partir de esto se puede calcular la frecuencia instantánea.

La detección de figuras en imágenes digitales se puede llevar a cabo por diferentes métodos; uno es la transformada de Hough cuyo objetivo es la detección principalmente de líneas rectas, círculos y elipses [32]. Por ejemplo, un método de reconstrucción del espacio tridimensional que fue propuesto en [33], se basa en la detección de grandes círculos mediante la transformada de Hough, en imágenes esféricas generadas por una cámara central; estos círculos corresponden a líneas geodésicas sobre la esfera.

La transformada de Hough se basa en la representación de las figuras a detectar en la imagen mediante ecuaciones algebraicas o paramétricas que contienen un número pequeño de parámetros. Por ejemplo, un círculo tiene tres parámetros siendo éstos su radio y las dos coordenadas de su punto centro, mientras que una elipse necesita cinco parámetros: su punto

centro, las longitudes de sus ejes mayor y menor, y su ángulo de inclinación. Cada figura en la imagen es caracterizada por sus parámetros específicos, por lo tanto puede ser representada por un punto en un tal llamado espacio de Hough cuya dimensión, es decir, su número de coordenadas, coincide con el número de parámetros. La idea clave del método de Hough es interpretar a cada punto de la imagen donde es candidato de una figura a detectar, como elemento de todas las figuras del mismo tipo que podrían pasar por este punto. Este conjunto de posibles figuras es transformado hacia el espacio de Hough donde genera algún tipo de curva. De esta manera, cada punto candidato de la imagen original genera una curva en el espacio de Hough. Si en la imagen original, varios puntos candidato pertenecen a una misma figura, entonces sus correspondientes curvas en el espacio de Hough tienen un punto común, y detectar a este punto común es equivalente a descubrir a los parámetros específicos de dicha figura. Para fines prácticos y tratando con imágenes digitales, el espacio de Hough es cuantizado mediante un arreglo de celdas y representado por una matriz acumulador para poder contabilizar el número de veces que cada celda es pasada por alguna curva generada por un punto candidato. Al final, máximos locales en esta matriz son detectados, obteniendo así los parámetros de las figuras cuya presencia en la imagen original es más evidente.

El método de Hough no solamente fue generalizado desde un detector de líneas rectas, que fue la propuesta original, hacia un método para detectar figuras representables por ecuaciones con parámetros, sino más recientemente, fue relacionado con técnicas generales de emparejamiento de plantillas que sirven para detectar figuras y formas más generales [26],[11].

Se han propuesto en la literatura diversos enfoques y algoritmos para detectar círculos y elipses en imágenes digitales, donde los problemas principales son la reducción de la gran memoria necesitada y los retos de lograr tanto una eficiencia razonable como también una exactitud y confidencialidad suficientes de los resultados.

La detección de figuras también se hace necesaria e importante en nubes de puntos. En la literatura, bajo una nube de puntos se entiende un conjunto de píxeles o vóxeles los cuales son parte de una imagen de dimensión dos o tres, respectivamente, y que pueden pertenecer a algún objeto o superficie de interés. En esta tesis solamente trabajamos con imágenes bidimensionales. Lo interesante de una nube de puntos es que, a diferencia de imágenes comunes, no existen bordes o contornos ni subconjuntos conexos que puedan servir para encontrar figuras, puesto que cada punto de la nube se encuentra separado con cierta distancia “grande” de los otros. Esto obliga a buscar métodos nuevos para la segmentación de la imagen y para encontrar objetos o figuras, pues, la mayoría de métodos comunes de segmentación están basados en cierta conectividad entre los elementos de los objetos de interés. La localización de objetos o figuras en las nubes, igualmente como para cualquier imagen digital, proporciona información esencial para el análisis del contenido de la imagen, en particular, hace posible la descripción y el reconocimiento de formas. Especialmente, eso permite la aproximación o adaptación de formas primitivas simples como segmentos de líneas, círculos o elipses, cuadrados o rectángulos, a los objetos presentes en la imagen. Para realizar esta búsqueda y detección de figuras en nubes de puntos, principalmente de círculos y elipses, se han propuesto algunos métodos en el contexto de diversas áreas de aplicación [7],[20],[22],[35]. Círculos y elipses detectados en nubes de puntos son utilizados, por ejemplo,

para el análisis de terrenos que contienen árboles, arbustos, edificios, ríos, etc., con el fin de una planificación de urbanización o industrial [11],[23],[33], o para la calibración de sensores 3D y escáneres 3D [27],[27],[29].

En la literatura se han propuesto también métodos los cuales no hacen uso de matrices de acumulación para determinar los parámetros, es decir, no se basan en transformaciones como la de Hough. Uno de estos métodos distintos es el descrito en [9], éste utiliza una *transformada de circlet* aplicada a la imagen gradiente, es decir, al conjunto de puntos cuyo valor de gradiente sobrepasa cierto umbral. La imagen entonces es representada como superposición de círculos con diferentes radios que forman un anillo con un cierto ancho, a esta formación de radios se le denomina circlets. Los circlets son caracterizados por el centro de los círculos, todos concéntricos, y la frecuencia finita que provee el ancho de estos circlets, es decir, la cantidad de radios presentes en el anillo. El objetivo principal es descomponer la imagen utilizando la transformada de Fourier. Dentro del dominio de Fourier se pretende obtener los coeficientes de la transformada de circlet que determinan los centros de los círculos y los radios asociados a los anillos. Por último, la imagen final es obtenida aplicando la transformada inversa de Fourier.

Otro enfoque se basa en la suposición que en la imagen ya fueron segmentados contornos o segmentos de curvas (digitales) los cuales supuestamente corresponden a arcos de círculos o de elipses. La conectividad dentro de la curva digital, la cual es proporcionada por los datos de entrada, es importante para que durante su seguimiento, arcos circulares son detectados o la curva es segmentada en tales arcos. Un método propuesto en [12] logra reconocer arcos circulares digitales de manera muy eficiente, a través del uso de la programación lineal en un determinado espacio de búsqueda.

Otro método para la detección de circunferencias o arcos circulares, dada una curva digital, es el propuesto en [5],[6], donde se hace uso de la definición precisa del círculo digital. Este método originalmente fue diseñado para la construcción de circunferencias en el plano discreto, y posteriormente fue extendido para determinar si una curva digital tiene circularidad digital. Este método resulta ser muy eficiente, sin embargo, a diferencia de otros métodos mencionados anteriormente aplicables a conjuntos desorganizados de píxeles, requiere una curva digital, es decir, una secuencia ordenada de píxeles formando un 8-camino irreducible, como dato de entrada.

Cuando se trata encontrar circunferencias en imágenes de conjuntos desorganizados de píxeles, por ejemplo, con nubes de puntos, una opción es el enfoque estadístico cuya finalidad es ajustar una circunferencia de forma óptima a los puntos de interés en la imagen (*circle fitting*). Es decir, los parámetros (centro y radio) de la circunferencia ajustada corresponden a la que pasa más cerca por la mayoría de los puntos dados. Los métodos para encontrar estos parámetros óptimos, comunmente, usan la representación algebraica del círculo y tratan de minimizar los errores de posición entre los puntos dados y puntos idealmente situados sobre la circunferencia. En general, eso corresponde a un problema no lineal de optimización, el cual depende de la métrica usada para medir distancias en el plano y del modelo del error. El enfoque estándar se basa en la métrica euclidiana y la minimización de los errores cuadráticos [10],[34]. Los métodos de esta clase también aprovechan diversas propiedades geométricas del

círculo. Por ejemplo, en los trabajos [25],[34], se determina una tripleta de puntos no alineados que forman un triángulo circunscrito en la circunferencia buscada. Para determinar el centro se aprovecha que la intersección de los bisectores de las mediatrices pertenecientes a los lados del triángulo corresponde al centro del círculo; luego se obtiene el radio calculando la distancia euclidiana de cualquiera de los tres puntos al centro.

Objetivos y metas de la tesis

Los **objetivos** de esta tesis son los siguientes:

- aprender sobre el modelado y la detección de círculos, elipses y arcos circulares en imágenes digitales bidimensionales,
- implementar algunos métodos de detección de circularidad, en el ambiente computacional MATLAB, y
- experimentar con estos programas computacionales,
- en particular, con imágenes de nubes de puntos, para evaluar el desempeño de los métodos implementados.

Para alcanzar estos objetivos, es necesario estudiar las bases matemáticas para la representación de círculos y elipses en el plano, tanto el euclidiano como también el discreto. La tesis analiza enfoques teóricos y métodos representativos conocidos de la literatura para describir y detectar circunferencias o arcos circulares.

El trabajo de implementación involucra completar el conocimiento sobre el ambiente computacional MATLAB, referente al manejo de imágenes digitales. Eso, conjunto con la fase de experimentación, permite entender mejor los métodos, pero también hace posible un análisis y una evaluación de los métodos, con respecto a su velocidad y exactitud.

En relación al último objetivo, una **meta** concreta y particular para esta tesis fue el desarrollo de un método para el tratamiento de una imagen de simulación proveniente del área de la astronomía. Esta imagen contiene muchos círculos a ser detectados, de diversos radios, los cuales están presentes en una nube de puntos. Para el problema de detección de dichos círculos, esta imagen es de alta complejidad.

Resultados y aportaciones de la tesis

Los objetivos de la tesis fueron cumplidos, como es explicado a continuación.

- La tesis presenta las bases matemáticas para la representación de círculos y elipses en el plano euclidiano y el plano discreto. La tesis describe enfoques teóricos y algunos métodos representativos de detección de círculos, elipses, y arcos circulares, conocidos

de la literatura. Se estudiaron principalmente métodos de acumulación de evidencia, para la detección de circunferencias y elipses. Además, el documento de tesis presenta un método interesante para determinar circularidad digital de una curva digital, que se basa en la geometría digital y en la teoría de números. La tesis contiene también una descripción introductoria y breve del enfoque de ajuste de círculos a conjuntos de puntos, basado en la optimización matemática.

- El trabajo de implementación y experimental se enfocó a cuatro métodos de acumulación de evidencias en un espacio de parámetros: el método general Hough 3D (H3D), un método para la detección de círculos basado en la transformada de Hough y valor y dirección del gradiente de la imagen (DG), un método basado en la detección de parejas de puntos finales de una cuerda (2P), y un método geométrico que usa la dirección del gradiente de la imagen (4P). Con el fin de entender mejor y de ilustración, también fue implementada una versión simplificada de un método de determinación de circularidad digital de una curva digital dada, la cual es representada por su código de longitudes de corridas. Todos los programas fueron desarrollados en el entorno computacional de MATLAB como GUI's, y resultaron funcionales. Además, los programas del método Hough realizados en esta tesis, ofrecen una alternativa para la detección de circunferencias en este entorno. La transformada de Hough es la base de muchos métodos para detectar circunferencias que son usados en la práctica; sin embargo, las funciones estándar disponibles en MATLAB para aplicar esta transformación, tienen muchas limitaciones.
- El trabajo experimental resultó en un estudio comparativo de los cuatro métodos programados (H3D, DG, 2P y 4P). Se concluyó que, para detectar circunferencias en imágenes digitales pequeñas, el método Hough 3D es el más eficiente al ofrecer los mejores resultados con respecto a su precisión y su robustez ante el ruido en la imagen.
- Se logró obtener un mejoramiento del método de detección de parejas de puntos finales de una cuerda (2P). Con la modificación realizada en éste, se logró contrarrestar el fallo que tiene el método original en la detección de circunferencias concéntricas, y se obtuvo disminuir considerablemente el tiempo de computo.
- Se realizaron experimentos con imágenes de nubes de puntos. Se desarrolló un método primero y preliminar para tratar una imagen de simulación proveniente del área de la astronomía, que presenta una alta complejidad para el problema de detección de círculos, debido a la presencia de muchos círculos de diversos radios en una nube de puntos.

Estructura de tesis

La tesis está estructurada como sigue: después de la introducción,

* el capítulo 1 proporciona preliminares sobre el plano discreto y la descripción de figuras básicas como líneas y círculos, y sobre los datos de entrada asumidos por los métodos estudiados en esta tesis.

* En el capítulo 2, las representaciones matemáticas de círculos y elipses en el plano euclidiano, son tratadas.

* El capítulo 3 está dedicado a la gran clase de métodos de detección de círculos y elipses basados en la transformada de Hough, es decir, en la acumulación de evidencia en un espacio de parámetros. Esta clase de métodos es la que fue estudiada con más detalle en esta tesis.

* Los capítulos 4 y 5 se dedican, de forma más breve, a métodos de otras dos clases basados en ideas muy distintas.

* En el capítulo 4 se aprovecha una definición formal de círculo digital y de circularidad, que dio lugar al desarrollo de un método muy eficiente tanto para graficar círculos en el plano discreto, como también para detectar círculos o arcos circulares en curvas digitales.

* El capítulo 5 presenta las ideas básicas del método de ajuste de círculos a conjuntos de puntos, siendo éste uno de los métodos más usados en la práctica para detectar círculos en imágenes digitales.

* El capítulo 6 describe programas computacionales que fueron elaborados sobre la plataforma MATLAB, como parte de esta tesis, y reporta sobre experimentos realizados que ilustran algunos métodos estudiados en esta tesis.

Capítulo 1

Imagen digital, imagen gradiente y puntos de borde

1.1. Imagen digital y plano discreto

Una imagen de dimensión n comúnmente es modelada por una función de n variables, con valores que son números reales o enteros o vectores. Una imagen puede presentar una escena del mundo real o alguna propiedad de la realidad que nos rodea, por ejemplo una propiedad física o química de la materia que fue medida por algún aparato. Fotografías, tomografías computarizadas, imágenes de la tierra generadas por un satélite, nubes de puntos generadas por un escáner 3D de Láser, o imágenes astronómicas generadas por un telescopio, son ejemplos de estos tipos de imágenes.

Cuando los valores de la función imagen son números, se dice que la imagen es monocromática, y los valores son interpretados como niveles de gris o de luz. En esta tesis se trabaja solamente con imágenes monocromáticas de dimensión dos, es decir, una imagen es una función f de \mathbb{R}^2 en \mathbb{R} . Para que una imagen pueda ser memorizada en una computadora, la imagen es digitalizada. Eso significa que su dominio de definición, \mathbb{R}^2 , es discretizado o muestreado, y acotado, y su dominio de valores, \mathbb{R} , es cuantizado y acotado [21],[36],[37]. El resultado es una imagen digital f_d definida sobre un subconjunto discreto y acotado de \mathbb{R}^2 , de manera estándar, una porción rectangular de \mathbb{Z}^2 , donde \mathbb{Z} denota al conjunto de los números enteros. Podemos suponer que la esquina inferior izquierda de este rectángulo sea el origen de un sistema cartesiano de coordenadas. El dominio de valores de una imagen digital es un intervalo de números enteros, en paquetes computacionales como MATLAB, por ejemplo, el intervalo $[0, 255]$, también llamado escala de gris, donde el valor de gris 0 corresponde al negro, y el valor 255 corresponde al blanco. La imagen digital puede ser vista como una matriz del tamaño del rectángulo sobre el cual está definida, con valores enteros entre 0 y 255. Los elementos del dominio de definición de una imagen digital son llamados píxeles. Para realizar estudios de carácter teórico, en general se supone que una imagen digital es definida sobre todo \mathbb{Z}^2 , también llamado el *plano discreto*.

Entonces, la *discretización* es un mapeo que transforma subconjuntos de \mathbb{R}^2 en subcon-

juntos del plano discreto \mathbb{Z}^2 , es decir, conjuntos de píxeles [37]. En la práctica, este mapeo es realizado por un escaneador o “frame grabber” el cual traza porciones de la imagen o señal original f , capta e integra los valores leídos para producir después un valor cuantizado y representativo de la porción trazada; este valor es asignado a un píxel. Eso permite la interpretación del plano discreto \mathbb{Z}^2 como una descomposición o un mosaiqueado de \mathbb{R}^2 en cuadrados unitarios: cada píxel siendo un punto en \mathbb{Z}^2 representa al cuadrado unitario centrado en dicho punto. Para definir diversos mapeos de discretización, esta identificación de cada píxel con su correspondiente cuadrado, “píxel=cuadrado”, es importante.

Algunas propiedades de \mathbb{R}^2 se mantienen en su subconjunto \mathbb{Z}^2 que es el plano discreto, por ejemplo, la estructura de espacio métrico. Recordemos que un *espacio métrico* es un conjunto arbitrario $A \neq \emptyset$ dotado de una función d definida en $A \times A$ con valores reales no negativos que para puntos cualesquiera $x, y, z \in A$ satisface lo siguiente: $d(x, y) = 0 \iff x = y$, $d(x, y) = d(y, x)$, $d(x, y) + d(y, z) \geq d(x, z)$. La función d entonces es llamada *métrica* sobre A . En el plano discreto, aparte de la métrica euclidiana, se trabaja con algunas otras métricas bien conocidas que son importantes en el contexto teórico pero también en el diseño de algoritmos para el tratamiento de imágenes digitales; mencionamos las dos más importantes:

$$d_1(p, q) = |x_1 - y_1| + |x_2 - y_2|, \text{ también denotada por } d_4,$$

$$d_\infty(p, q) = \max\{|x_1 - y_1|, |x_2 - y_2|\}, \text{ también denotada por } d_8,$$

para puntos $p = (x_1, y_1)$, $q = (x_2, y_2)$. Estas funciones son métricas tanto sobre \mathbb{R}^2 como también sobre \mathbb{Z}^2 .

1.2. Conectividad en el plano discreto

Una de las propiedades de \mathbb{R}^2 que son destruidas en el paso al subconjunto \mathbb{Z}^2 , es la conexidad. Un conjunto $M \subseteq \mathbb{R}^2$, intuitivamente, es conexo si consiste de una sola pieza. Por ejemplo, el conjunto total \mathbb{R}^2 , y cualquier línea recta, segmento de línea, disco, triángulo, rectángulo, punto solitario, curva o arco, son conexos. Formalmente, la conexidad es un concepto topológico, y M es **conexo** (en \mathbb{R}^2) si M no puede ser descompuesto mediante dos conjuntos abiertos $M_1, M_2 \subseteq \mathbb{R}^2$ tales que $M = (M_1 \cap M) \cup (M_2 \cap M)$, $M_1 \cap M_2 \cap M = \emptyset$. En caso de existir tales M_1, M_2 , M se llama **disconexo**. Considerando \mathbb{Z}^2 como subconjunto del espacio topológico \mathbb{R}^2 , es evidente que \mathbb{Z}^2 puede ser descompuesto mediante “muchos” conjuntos abiertos: Cada píxel $p \in \mathbb{Z}^2$ está en un disco abierto que contiene a ningún otro píxel, por ejemplo, en el disco centrado en p con radio $1/2$. Eso significa que no solamente el plano discreto \mathbb{Z}^2 es desconexo, sino además, cualquier conjunto de píxeles es desconexo. Pero entonces: ¿Cómo pueden ser modeladas figuras, líneas o curvas en un plano donde “nada” es conexo?

Para introducir al plano discreto \mathbb{Z}^2 un concepto de conexidad que hace posible modelar objetos de interés en imágenes digitales así como desarrollar métodos para encontrarlos, han sido definidas relaciones de adyacencia o de vecindad, por ejemplo, con la ayuda de las métricas arriba mencionadas:

Dos píxeles $p = (x_1, y_1), q = (x_2, y_2) \in \mathbb{Z}^2$ se llaman k -**vecinos**, para $k \in \{4, 8\}$, si $d_k(p, q) = 1$. Denotando al conjunto de los k -vecinos de un píxel $s = (x, y)$ por $N_k(s)$, resulta que $N_k(s)$ tiene k elementos:

$$N_4(s) = \{(x+1, y), (x-1, y), (x, y+1), (x, y-1)\},$$

$$N_8(s) = N_4(s) \cup \{(x+1, y+1), (x+1, y-1), (x-1, y+1), (x-1, y-1)\}.$$

Las relaciones de k -vecindad generan un concepto natural de camino siendo lo equivalente a una curva o trayectoria en el plano euclidiano: Un k -**camino** de p a q es una secuencia de píxeles $\{p = p_0, p_1, p_2, \dots, p_m = q\}$ tales que para todo $i = 0, 1, \dots, m-1$, p_i es k -vecino de p_{i+1} . Si existe tal camino, los puntos p, q se llaman k -**conectados** (en \mathbb{Z}^2). Estos caminos crean conexidad en el plano discreto: Un conjunto de píxeles $M \subseteq \mathbb{Z}^2$ se llama k -**conexo** si para cada dos puntos $p, q \in M$, existe un k -camino de p a q de píxeles de manera que todos pertenecen a M , es decir, p y q son k -conectados en M . Cuando un conjunto no es k -conexo, puede ser interesante estudiar sus partes conexas: Sea $M \subseteq \mathbb{Z}^2$, un subconjunto A de M se llama k -**componente (conexa)** de M si A es k -conexo y es maximal con esta propiedad, es decir, para cualquier píxel $p \in (M \setminus A)$, el conjunto $A \cup \{p\}$ no es k -conexo.

Para más detalles sobre estas definiciones vea [21],[36],[37]. Para una gran variedad de problemas de aplicación, es conveniente modelar a los objetos de interés presentes en una imagen digital mediante subconjuntos k -conexos de píxeles. Esta k -conexidad es esencial no solamente para describir a estos objetos, sino también para desarrollar métodos de segmentación que los detectan y los extraen de la imagen.

En algunas áreas específicas de aplicación del análisis de imágenes digitales, aparecen conjuntos de píxeles “altamente desconexos” llamados nubes de puntos, los cuales, sin embargo, pueden representar a uno o a varios objetos de interés, o los contienen. Una **nube de puntos** es un conjunto de píxeles cuyas k -componentes conexas, para cualquier k , consisten de muy pocos puntos; la mayoría de las k -componentes son píxeles aislados. Equivalentemente, cada píxel de una nube de puntos tiene una d_k -distancia estrictamente mayor a 1 a casi todos los otros píxeles de la nube. Para tratar o analizar nubes de puntos, la falta de conexidad causa que muchos métodos de procesamiento y de segmentación sean inservibles en su forma original. Los métodos tienen que ser adaptados a la situación especial de desconexidad, o la nube tiene que ser tratada previamente, por ejemplo agregando más puntos mediante interpolación y así generando objetos de interés conexos.

1.3. Mapeos de discretización

A continuación se mencionan los más importantes *mapeos de discretización* que transforman subconjuntos del plano euclidiano en conjuntos de píxeles, los cuales a la vez son subconjuntos de \mathbb{Z}^2 como también son conjuntos de cuadrados unitarios. Para eso necesitamos ampliar el modelo del plano discreto un poco: El conjunto de píxeles \mathbb{Z}^2 también es tratado en la literatura como el conjunto de puntos de intersección de la *retícula cuadrática* que se forma por todas las líneas rectas horizontales que pasan por puntos (x, y) con

$x \in \mathbb{R}$, $y \in \mathbb{Z}$ y las líneas rectas verticales pasando por puntos (x, y) con $y \in \mathbb{R}$, $x \in \mathbb{Z}$ en el plano euclidiano. Todas estas líneas son llamadas *líneas de retícula*, y sus puntos de intersección, siendo los píxeles, son entonces los **puntos de retícula** (*grid points*). Sea ahora $S \subseteq \mathbb{R}^2$, $S \neq \emptyset$.

* La **discretización de Gauss** de S , $G(S)$, es la unión de todos los píxeles contenidos en S , o equivalentemente, de todos los cuadrados cuyo punto centro está en S . Aun siendo S no vacío, $G(S)$ puede resultar vacío. En general, ni la conexidad ni la desconexidad de S es preservada bajo esta discretización.

* La **discretización de Jordan exterior** de S , $J^+(S)$, es la unión de todos los cuadrados que son intersecados por el conjunto S . Esta discretización preserva la conexidad pero no preserva la desconexidad, pues, puede rellenar huecos o cerrar agujeros; $J^+(S)$ es un conjunto no vacío.

* La **discretización de Jordan interior** de S , $J^-(S)$, es la unión de todos los cuadrados contenidos completamente en S . Esta discretización no preserva conexidad y no rellena huecos ni agujeros; $J^-(S)$ puede resultar vacío.

* La **discretización de intersección con la retícula** es adecuada para líneas, curvas o arcos: la discretización de una curva γ del plano \mathbb{R}^2 es el conjunto $R(\gamma)$ de todos los píxeles que son los más cercanos a los puntos de intersección entre una línea de retícula con la curva γ . $R(\gamma)$ es generado como una lista de puntos, según las siguientes reglas:

— Para cada punto s de intersección de γ con una línea de retícula, el píxel más cercano a s es agregado a la lista $R(\gamma)$.

— Si existen dos píxeles con la misma distancia euclidiana al punto s , sólo el píxel con la coordenada x mayor es registrado. Si la coordenada x es la misma para ambos píxeles, el píxel con la coordenada y mayor es agregado a $R(\gamma)$.

— Si dos puntos s_1 y s_2 fueron detectados de manera consecutiva, y tienen el mismo píxel más cercano, este píxel es agregado solamente una vez en la lista $R(\gamma)$.

El resultado de la aplicación de estas reglas para discretizar una curva es un 8-camino. Cuando γ es una línea recta, la lista $R(\gamma)$ corresponde a un 8-camino irreducible, es decir, en el cual, la eliminación de cualquier píxel no final rompe la 8-conexidad. Una aplicación literal de estas reglas para discretizar un segmento de línea recta es la base del muy conocido algoritmo de Bresenham [21], el cual se encuentra implementado en muchos paquetes computacionales como un estándar para dibujar líneas rectas. Para más detalles, vea [21],[36],[37].

1.4. Líneas rectas y círculos digitales

Toda figura geométrica en el plano puede ser discretizada para obtener su versión llamada discreta o digital. Para obtener una curva digital, una línea recta digital, o un círculo digital, el mapeo de la discretización de intersección con la retícula es usado, para lograr que estos objetos digitales tengan propiedades deseadas de conectividad en el plano discreto [36].

Una *curva* parametrizada γ en el plano euclidiano es la imagen continua de un segmento de línea: $\gamma = \{f(t) = (x(t), y(t)) \in \mathbb{R}^2 : t \in [a, b] \subseteq \mathbb{R}\}$.

La curva γ se dice *abierta* si $f(a) \neq f(b)$, *cerrada* si $f(a) = f(b)$; se denomina *simple* si $f(s) \neq f(t)$ para todos $s, t \in [a, b]$ tales que $a \leq s < t < b$, *curva de Jordan* si es simple y cerrada, y *arco de Jordan* si es simple y abierta.

En analogía a estas propiedades, una **curva digital** o k -curva C es un k -camino en el plano discreto \mathbb{Z}^2 , donde por ejemplo $k = 4$ o $k = 8$, vea sección 1.1. Es decir, C es una secuencia $C = \{p_1, p_1, \dots, p_k\}$ de píxeles tal que p_i es k -vecino de p_{i+1} , $i = 1, 2, \dots, k - 1$. Si además, p_k es k -vecino de p_1 , la curva C es *cerrada*. Entonces C es un conjunto k -conexo de píxeles, el cual puede contener *puntos finales* definidos por tener exactamente un k -vecino en C . Para describir que una curva es un conjunto “largo pero delgado”, se pide que una k -curva C cumpla que para cualquier punto $p \in C$ que no es final, $C \setminus \{p\}$ no es k -conexo; esta propiedad es la *k-irreducibilidad* mencionada en la sección anterior. Una *k-curva de Jordan* es definida como un k -camino (cerrado) C en el cual, cada punto tiene exactamente dos k -vecinos en C . La discretización $R(\gamma)$ (de intersección con la retícula) de una curva γ como fue detallada en la sección anterior, es un 8-camino pero en general, no es 8-irreducible.

Sin embargo, si γ es una línea recta (o segmento de línea o rayo), entonces $R(\gamma)$ es un 8-camino irreducible llamado **línea digital** (DSL - *digital straight line*) o **segmento de línea digital** (DSS - *digital straight line segment*) o **rayo digital** (DR - *digital ray*). Toda línea recta $\gamma_{m,b}$ es especificada por una ecuación $y = mx + b$, donde m es la pendiente y b una constante de traslación: $\gamma_{m,b} = \{(x, mx + b), x \in I\}$, donde $I = \mathbb{R}$ para una línea recta, $I = [0, \infty)$ para un rayo, o $I = [s, t]$ un intervalo para un segmento de línea recta. El resultado de la discretización $R(\gamma)$ (vea sección anterior) de una línea recta $\gamma_{m,b}$ con pendiente $m \in [0, 1]$ que es trazada desde su lado izquierdo al lado derecho, es como sigue:

$$\text{DR: } \{(n, y_n), n \in \mathbb{N} \cup \{0\}, y_n = \lfloor mn + b + 0,5 \rfloor\}$$

$$\text{DSL: } \{(n, y_n), n \in \mathbb{Z}, y_n = \lfloor mn + b + 0,5 \rfloor\}$$

$$\text{DSS: } \{(n, y_n), n \in \mathbb{Z}, n_1 \leq n \leq n_2, y_n = \lfloor mn + b + 0,5 \rfloor\}$$

donde $\lfloor a \rfloor = \max\{z \in \mathbb{Z} : z \leq a\}$, $a \in \mathbb{R}$ (*función de redondeo hacia abajo*).

Una manera de representar un 8-camino $(p_1, p_2, \dots, p_{k-1}, p_k)$, es guardando al punto inicial p_1 y una secuencia $F = (f_1, f_2, \dots, f_{k-1})$ llamada **código de cadena según Freeman** (*chain code*), donde cada número $f_i \in \{0, 1, \dots, 7\}$ codifica al paso de p_i hacia su 8-vecino p_{i+1} , vea figura 1.1. En particular, eso es aplicable a un DSS, puesto que es un conjunto finito de píxeles y un 8-camino. El código de cadena de cualquier DSS contiene a lo más dos números distintos, por ejemplo 0, 7 si el segmento de línea cuya discretización es este DSS, tiene una pendiente entre -1 y 0 , vea figura 1.1, o 0, 1 para una pendiente entre 0 y 1 .

Algunas propiedades del código de cadena de un DSS son características para su linealidad o “rectitud”. Analizar un código de cadena significa analizar una palabra sobre el alfabeto de los números $0, 1, 2, \dots, 7$, eso se relaciona con áreas de las matemáticas discretas y de la computación.

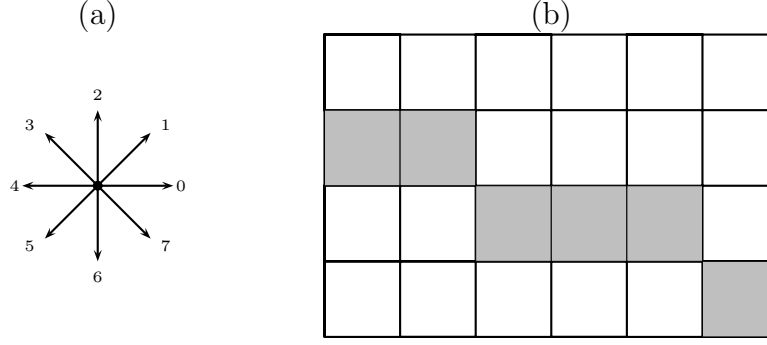


Figura 1.1: (a) Código de Freeman. (b) El código de cadena del segmento de línea, iniciando con el píxel izquierdo de color gris, es $F = (07007)$, su código de longitudes de corridas es $S = (2, 3, 1)$.

El código de cadena $F = \{f_1, f_2, \dots, f_k\}$ de cualquier DSS tiene las siguientes propiedades [21],[37]:

- (1) F contiene sólo un número, es decir, $f_1 = f_2 = \dots = f_k$, o, contiene dos números distintos r, s tales que $|r - s| = 1 \pmod{8}$.
- (2) Si F contiene dos números distintos, entonces uno de estos es singular, es decir, aparece en F siempre sin repetición inmediata.
- (3) Sea F' generado a partir de F como sigue: Si F no tiene elementos no-singulares, entonces $F' := \emptyset$. En otro caso, trazando F de la izquierda a la derecha,
 - cada elemento singular de F es ignorado,
 - para cada grupo de n ($n \geq 2$) elementos consecutivos de F que son el mismo número, el número n es agregado a F' .

Como resultado, F' es una secuencia de números naturales, en general, más corta que la secuencia F . Después de quitarle a F' su primera y su última letra, la secuencia restante F' cumple lo siguiente, entre otras propiedades: F' contiene sólo un número (como letra), o dos números distintos s, r tales que $|r - s| = 1$. Si F' contiene dos números distintos, entonces al menos uno de éstos es singular.

Propiedades de este tipo son usadas en algoritmos de detección de líneas rectas o de la segmentación de 8-caminos en DSS de máxima longitud. Cuando un código de cadena las cumple todas, queda verificado que este código representa un DSS. La secuencia F' de arriba cumple más propiedades, y los algoritmos de reconocimiento de DSS luego, iterativamente, derivan otras secuencias de números a partir de F' , cada vez más cortas. Cuando un código de cadena fue verificado como DSS, entonces se puede determinar la pendiente del segmento de línea recta original a partir de propiedades de periodicidad del código.

La construcción de F' se basa en observar cuántas veces se repite la misma dirección de paso en el 8-camino original. Eso es relacionado con otra representación útil para ciertos tipos de 8-camino: El **código de longitudes de corridas** (*run length code*) es una secuencia de números correspondientes a las longitudes de segmentos de línea que son todos horizontales o todos verticales, llamados **corridas**, donde la primera corrida empieza en un píxel fijado como inicial.

Por ejemplo, el 8-camino de la figura 1.1 con su píxel inicial y su código de cadena $F = (07007)$, puede ser igualmente representado por las longitudes de segmentos horizontales, cada longitud dada por el número de píxeles: 2, 3, 1. Esta información solamente es útil sabiendo que este camino siempre avanza o bien horizontalmente o bien, da un paso hacia abajo en dirección de Freeman 7. En general, un DSS con pendiente entre -1 y 0 , cuando es trazado desde la izquierda hacia la derecha, siempre da uno o varios pasos horizontales en dirección 0 , y ocasionalmente, da un paso hacia abajo en dirección 7 . Solamente por estas circunstancias especiales, su código de longitudes de corridas, cuyas corridas son segmentos horizontales, junto con el punto inicial, describe completamente al DSS.

Similarmente, un DSS con pendiente entre 0 y 1 , trazado desde la izquierda hacia la derecha, avanza en corridas horizontales, y ocasionalmente, da un paso hacia arriba en dirección 1 . Solo cuando se tiene la información adicional del sentido de trazado, y del intervalo $[0, 1]$ para su pendiente, su código de longitudes de corridas (horizontales), junto con el punto inicial, caracteriza al DSS. Una pendiente en el intervalo $[0, 1]$ significa una inclinación de la línea entre 0 y 45 grados. Un DSS con inclinación entre 45 y 90 grados avanza principalmente hacia arriba, en segmentos verticales en dirección 2 , y ocasionalmente da un paso hacia la derecha en dirección 0 . Entonces sus corridas son verticales, y su código de longitudes de corridas describe este DSS completamente, siempre y cuando se tenga la información adicional sobre el intervalo de su posible inclinación y su dirección de trazado hacia arriba.

Los códigos de longitudes de corridas son usados igualmente como los códigos de cadena en algoritmos de reconocimiento de líneas rectas o para segmentar 8-caminos en DSS [6].

Un **círculo digital** es el resultado de la discretización de la intersección con la retícula que fue detallada en la sección 1.3, de una circunferencia en \mathbb{R}^2 . Este modelo de círculo digital es la base del algoritmo de Bresenham para discretizar círculos o para graficar círculos en el plano \mathbb{Z}^2 de píxeles [21], el cual se encuentra implementado en paquetes computacionales como una herramienta estándar.

Un círculo digital con centro $o = (0, 0)$ y radio entero r puede ser separado en ocho octantes, los cuales presentan importantes propiedades de simetría, vea figura 1.2. El primer octante determina todo el círculo: cada punto (x, y) del círculo del octante 1 tiene ambas coordenadas positivas, y entonces, el punto (y, x) del octante 2 también es del círculo, igualmente los puntos $(y, -x)$ del octante 3, $(x, -y)$ del octante 4, $(-x, -y)$ del octante 5, $(-y, -x)$ del octante 6, $(-y, x)$ del octante 7, y $(-x, y)$ del octante 8.

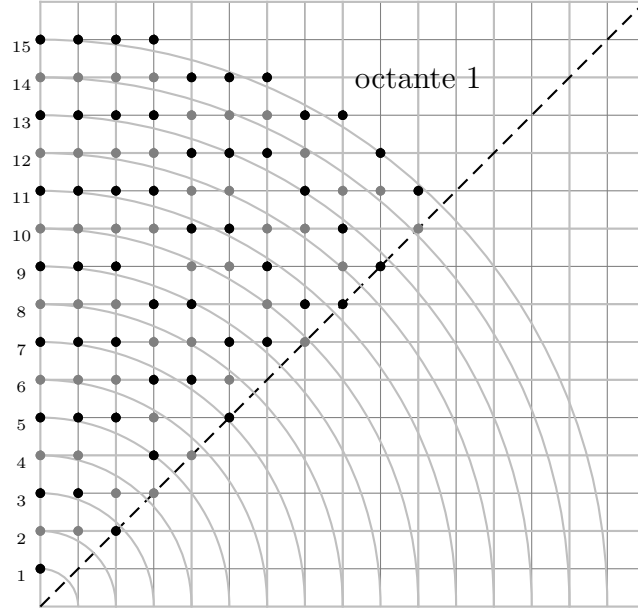


Figura 1.2: Discretización de la intersección con la retícula sobre el primer octante para círculos de radios $r = 1, \dots, 15$.

Recordemos la ecuación algebraica para todo punto $(x, y) \in \mathbb{R}^2$ de un círculo con centro $o = (0, 0)$ y radio r (vea también el siguiente capítulo),

$$x^2 + y^2 = r^2 \iff y = \sqrt{r^2 - x^2} \iff |y - \sqrt{r^2 - x^2}| = 0 .$$

Cuando (x, y) son los píxeles de un círculo digital, en general, no pueden cumplir exactamente la condición $|y - \sqrt{r^2 - x^2}| = 0$, pero entonces, la discretización impone el requerimiento que $|y - \sqrt{r^2 - x^2}| < 0,5$. En consecuencia, la parte del octante 1 de un círculo digital con centro $o = (0, 0)$ y radio entero r es dado por ([5],[6])

$$\mathcal{C}_{Oct1}^{\mathbb{Z}}(o, r) = \left\{ (x, y) \in \mathbb{Z}^2 : 0 \leq x \leq y \leq r, |y - \sqrt{r^2 - x^2}| < \frac{1}{2} \right\} .$$

Aprovechando las simetrías entre los octantes, el círculo digital completo entonces es dado por

$$\mathcal{C}^{\mathbb{Z}}(o, r) = \left\{ (x, y) \in \mathbb{Z}^2 : (|x|, |y|) \in \mathcal{C}_{Oct1}^{\mathbb{Z}} \text{ o } (|y|, |x|) \in \mathcal{C}_{Oct1}^{\mathbb{Z}} \right\} .$$

Más generalmente, un círculo digital con centro $p = (x_p, y_p) \in \mathbb{Z}^2$ y radio entero r es representado como sigue:

$$\mathcal{C}^{\mathbb{Z}}(p, r) = \left\{ (x + x_p, y + y_p) \in \mathbb{Z}^2 : (x, y) \in \mathcal{C}^{\mathbb{Z}}(o, r) \right\} .$$

La figura 1.3 muestra la parte del primer octante de un círculo digital de radio 13, siendo los cuadrados grises los píxeles del círculo. La curva del círculo forma un 8-camino, cuyo píxel

superior izquierdo consideramos como punto inicial. Entonces, este 8-camino en el octante 1, avanza en corridas horizontales hacia la derecha, y ocasionalmente baja en dirección de Freeman 7, para iniciar una nueva corrida de longitud al menos 1. Cualquier círculo digital $\mathcal{C}^{\mathbb{Z}}(p, r)$ tiene este comportamiento en su primer octante. En particular, el punto $(x_p, y_p + r)$ siempre pertenece a este círculo y es considerado su punto inicial para el trazado. Por eso, el código de longitud de corridas es un buen descriptor para el octante 1 del círculo digital. El ejemplo de la figura 1.3 tiene el código de longitud de corridas $S = (4, 3, 1, 1, 1)$.

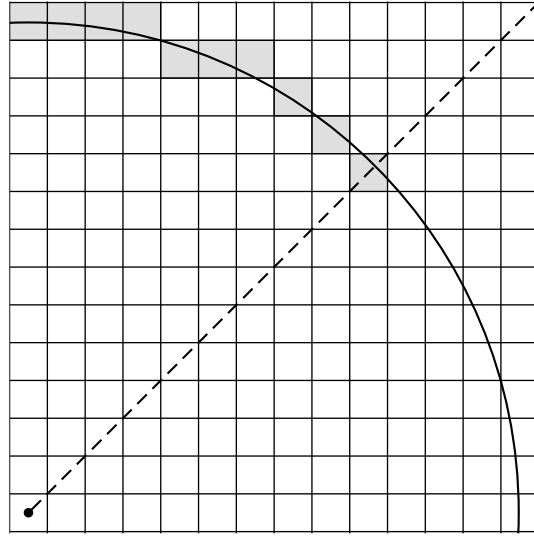


Figura 1.3: Discretización de intersección con la retícula en el primer octante de una circunferencia de radio $r = 13$.

1.5. Imagen de gradiente y puntos de borde

Todos los métodos de detección de círculos o de elipses expuestos en esta tesis, se aplican a una imagen en la cual los puntos candidatos a pertenecer a una de estas curvas de interés, ya se encuentran segmentados del resto de la imagen y marcados. Este es el caso cuando una imagen de entrada es binaria y sus puntos de la primera clase (por ejemplo, los negros) son estos puntos candidatos, y sus otros puntos (por ejemplo, los blancos) forman el fondo de la imagen. Vamos a suponer además que los puntos candidato pertenecen a curvas, entre las cuales se encuentran las curvas a detectar, o partes de ellas, pero que los puntos candidato no rellenan los interiores encerrados por estas curvas. Es decir, el conjunto de puntos candidato pertenece al conjunto de puntos de la curva de los elipses a detectar, o a las circunferencias, pero no existen esencialmente muchos puntos candidato en los interiores de tales elipses o dentro de los discos.

En general, una imagen dada no cumple esta condición, por lo cual es necesario darle primero un tratamiento de **detección de bordes**.

En el procesamiento de imágenes digitales, el preprocesamiento para la detección de bordes o cantos es un método estándar. Un **borde** o canto es un lugar en la imagen donde los valores de la función f que describe la intensidad o el valor de gris de la imagen, cambian notablemente. Eso puede ocurrir en un punto, entonces llamado **punto de borde**, o a lo largo de varios puntos que forman un conjunto llamado **borde** o canto. Comúnmente, los bordes forman los contornos o fronteras de objetos o figuras de interés. Un contorno de objeto, intuitivamente, delimita al objeto con respecto al fondo de la imagen, en otras palabras, se encuentra entre la región del objeto y la del fondo (o la de otro objeto).

El gradiente de una función $f(x, y) : \mathbb{R}^2 \rightarrow \mathbb{R}$ es definido como el siguiente vector:

$$\nabla f = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right) = (f_x, f_y)$$

El gradiente describe las variaciones locales de la imagen, es decir, cambios locales de la función f . Mientras más grande sea la variación alrededor de un punto, el valor del gradiente del punto será mayor. Otra información que brinda el gradiente, es la dirección hacia donde apunta la mayor variación. La magnitud del gradiente está dada por su norma euclidiana:

$$|\nabla f| = \sqrt{\left(\frac{\partial f}{\partial x} \right)^2 + \left(\frac{\partial f}{\partial y} \right)^2}$$

mientras que su pendiente puede ser calculada como

$$\tan \theta = \frac{f_y}{f_x} .$$

El gradiente de una función de imagen digital $f(x, y) : \mathbb{Z}^2 \rightarrow \mathbb{Z}$ no puede ser calculado como fue definido arriba, sino solamente por

$$\text{Grad}(f) = \text{Grad}(f(x, y)) = (G_x, G_y)$$

donde G_x es una aproximación digital de $\frac{\partial f}{\partial x}$, y G_y es una aproximación de $\frac{\partial f}{\partial y}$. El valor de este gradiente es su norma, y su pendiente $\tan \theta$ es dada como antes:

$$|\text{Grad}(f)| (x, y) = \sqrt{G_x^2 + G_y^2}, \quad (\tan \theta) (x, y) = \frac{G_y}{G_x}$$

Una versión sencilla para G_x y G_y es la siguiente:

$$G_x(x, y) = f(x, y) - f(x - 1, y), \quad G_y(x, y) = f(x, y) - f(x, y - 1)$$

donde el gradiente es medido en la posición interpolada $(x - \frac{1}{2}, y)$. Para calcular el gradiente en la posición (x, y) , sirve la siguiente versión:

$$G_x(x, y) = f(x + 1, y) - f(x - 1, y), \quad G_y(x, y) = f(x, y + 1) - f(x, y - 1)$$

Notemos que G_x y G_y son calculados en cada punto (x, y) de la imagen original, donde se involucran los valores de f en diversos puntos, para calcular al valor $G_x(x, y)$ que en general, es distinto al valor original $f(x, y)$. $G_x(x, y)$ y $G_y(x, y)$ son nuevas imágenes, llamadas **imágenes de gradiente en dirección x, y** , respectivamente. También $|\text{Grad}(f)|(x, y)$ cuyo valor en cada punto (x, y) es dado por $\sqrt{(G_x(x, y))^2 + (G_y(x, y))^2}$, es una nueva imagen llamada **imagen gradiente**. En esta tesis entendemos bajo **puntos candidato o puntos de interés** todos los puntos de la imagen original cuyo valor de gradiente $|\text{Grad}(f)|$ sobrepasa un umbral predefinido, estos también son llamados **puntos de borde**. Además, para varios métodos bajo estudio en esta tesis, se supone que en cada punto candidato se ha calculado al ángulo de la dirección o a la pendiente del gradiente en este punto.

La fórmula $G_x(x, y) = f(x+1, y) - f(x-1, y)$ puede ser escrita como la convolución de la imagen original f con una imagen h la cual tiene valor 1 en una posición correspondiente a $(x+1, y)$, valor -1 en la posición correspondiente a $(x-1, y)$, y valor cero en cualquier otro punto. La porción interesante de h que tiene valores distintos de cero es llamada **máscara** para el siguiente **filtrado lineal** (usando un sistema cartesiano de coordenadas (x, y) estándar), usando máscaras h_1, h_2 :

$$\begin{aligned} G_x(x, y) &= (f * h_1)(x, y) = (f * [-1 \ 0 \ 1]) (x, y) \\ &= -1 \cdot f(x-1, y) + 0 \cdot f(x, y) + 1 \cdot f(x+1, y) = f(x+1, y) - f(x-1, y). \end{aligned}$$

Similarmente,

$$\begin{aligned} G_y(x, y) &= (f * h_2)(x, y) = \left(f * \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} \right) (x, y) \\ &= -1 \cdot f(x, y-1) + 0 \cdot f(x, y) + 1 \cdot f(x, y+1) = f(x, y+1) - f(x, y-1). \end{aligned}$$

La siguiente versión toma en cuenta mejor que f es de dimensión dos, y usa las conocidas *Máscaras de Prewitt*, vea [31]:

$$\begin{aligned} G_x(x, y) &= \left(f * \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \right) (x, y) = -f(x-1, y-1) + f(x+1, y-1) \\ &\quad -f(x-1, y) + f(x+1, y) - f(x-1, y+1) + f(x+1, y+1) \\ G_y(x, y) &= \left(f * \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \right) (x, y) = -f(x-1, y-1) - f(x, y-1) \\ &\quad -f(x+1, y-1) + f(x-1, y+1) + f(x, y+1) + f(x+1, y+1) \end{aligned}$$

El operador G_x mide la diferencia en dirección horizontal, por lo cual, tendrá valores resultantes altos al encontrarse con un borde vertical en la imagen. Es decir, G_x detecta bordes

verticales. Similarmente, G_y cuya dirección es vertical, detecta bordes horizontales. La pendiente del gradiente siempre es perpendicular a la dirección del borde que detecta. Pueden ser diseñadas máscaras de filtrado tipo gradiente que logran detectar o resaltar (para valores altos a la salida) bordes de ciertas direcciones en la imagen.

En la práctica del procesamiento de imágenes, son usadas también otras versiones del valor del gradiente $|\text{Grad}(f)|$ o de su dirección, por ejemplo:

$$|\text{Grad}(f)|_1(x, y) = |G_x| + |G_y|, \quad |\tan \theta|(x, y) = \left| \frac{G_y}{G_x} \right| = \frac{|G_y|}{|G_x|}$$

Cuando G_y, G_x ambos son positivos o ambos son negativos, entonces en este punto (x, y) vale que $|\tan \theta| = \tan \theta$. Sin embargo, para G_y, G_x de diferentes signos, se tiene que

$$|\tan \theta| = \left| \frac{G_y}{G_x} \right| = -\frac{G_y}{G_x} = -\tan \theta.$$

Eso significa que, si un programa computacional de procesamiento de imágenes calcula la pendiente del gradiente no como G_y/G_x sino como $|G_y/G_x|$, el error consiste únicamente en que ya no se podrá distinguir entre una pendiente y la pendiente opuesta; ambos resultados de pendiente saldrán iguales y positivos. Tales gradientes siempre detectan un borde de la misma orientación. La conclusión importante es que, dadas las circunstancias de la plataforma computacional que se usa para procesar imágenes, una estrategia para detectar dos bordes paralelos en la imagen, consiste en buscar dos puntos con pendientes de gradiente iguales, o con pendientes de gradiente opuestos. Eso se usará en un método de detección de elipses que será tratado en esta tesis.

Capítulo 2

Circunferencias y elipses en el plano euclidiano

2.1. Círculo o circunferencia

2.1.1. Definición geométrica

De acuerdo a la geometría euclidiana del plano \mathbb{R}^2 , un **círculo**, también llamado **circunferencia**, es definido como un conjunto de puntos en el plano que son equidistantes desde un punto fijo, denominado *centro* del círculo. La distancia r entre el centro y un punto $p = (x, y)$ cualquiera de la circunferencia es conocido como *radio*, $2r$ es su *diámetro*. De forma que, un radio $r \in \mathbb{R}, r \geq 0$ y un punto centro $q \in \mathbb{R}^2$ determinan a una única circunferencia dada por

$$C(r, q) = \{p \in \mathbb{R}^2 : d(p, q) = r\} ,$$

donde d es una métrica sobre \mathbb{R}^2 , comúnmente la métrica euclidiana. La circunferencia delimita un área llamada disco. El conjunto $B(r, q) = \{p \in \mathbb{R}^2 : d(p, q) < r\}$ es llamado **disco abierto**, y $D(r, q) = \{p \in \mathbb{R}^2 : d(p, q) \leq r\}$ es un **disco cerrado**. Claro que $C(r, q) = D(r, q) \setminus B(r, q)$.

Como es bien conocido, el perímetro de una circunferencia con radio r está dado por $Perimetro = 2\pi r$, mientras que el área A del disco correspondiente puede ser calculado por $A = \pi r^2$. Históricamente fue importante la observación que la razón entre perímetro y diámetro permanece constante con el valor π aunque cambie el tamaño del círculo; en la antigüedad eso fue usado para calcular aproximaciones del número π , aplicando métodos aproximativos y experimentales para determinar el perímetro.

En el círculo se pueden identificar las siguientes características:

- * *Cuerda*: es cualquier segmento de línea recta que une dos puntos de la circunferencia $C(r, q)$.
- * *Arco*: es la parte de la circunferencia $C(r, q)$ que une dos puntos distintos de ésta.
- * *Punto interior* de la circunferencia $C(r, q)$: es cualquier punto que pertenece a $B(r, q)$.

- * *Punto exterior* de la circunferencia $C(r, q)$: es cualquier punto que no pertenece al disco cerrado $D(r, q)$, y por tanto, tiene una distancia hacia el centro q mayor que r .
- * *Ángulo central*: es el generado por dos cuerdas que coinciden con el centro de la circunferencia q .
- * *Ángulo inscrito*: es el generado por dos cuerdas que coinciden en la circunferencia en un punto $p = (x, y)$.
- * *Recta tangente*: es una línea recta que incide en un único punto en la circunferencia.
- * *Recta secante*: es una línea recta que incide en la circunferencia en dos de sus puntos.

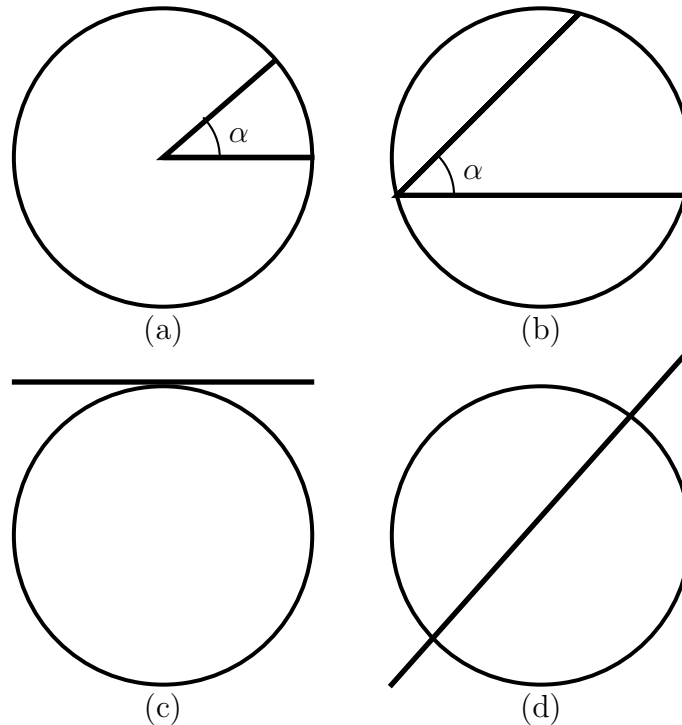


Figura 2.1: Características del círculo: (a) ángulo central, (b) ángulo inscrito, (c) recta tangente, (d) recta secante.

2.1.2. Representación algebraica

La ecuación algebraica del círculo puede ser deducida a través del teorema de Pitágoras: Para cualquier punto de una circunferencia, $p = (x, y) \in C(r, q)$, con radio r y centro en el origen, $q = (0, 0)$, se tiene lo siguiente (vea figura 2.2):

$$x^2 + y^2 = r^2 .$$

Cuando el círculo es trasladado por el vector (h, k) , es decir, $q = (h, k)$, se obtiene la **ecuación algebraica del círculo**:

$$(x - h)^2 + (y - k)^2 = r^2 .$$

La ecuación general de un círculo puede ser escrita también como sigue:

$$x^2 + y^2 + ex + fy + g = 0 ,$$

donde e, f, g son constantes. Esto es una ecuación algebraica en dos variables de segundo orden.

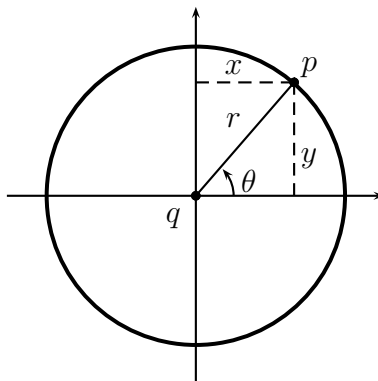


Figura 2.2: Representaciones del círculo.

2.1.3. Representación paramétrica

La circunferencia es una curva, es decir, $C(r, q)$ es la imagen continua de un intervalo I (por ejemplo $I = [0, 1]$ o $I = [0, t_{\text{máx}}]$), bajo una función f definida sobre I y con valores en \mathbb{R}^2 .

Una de las representaciones paramétricas puede ser deducida de la definición de las funciones trigonométricas seno y coseno, vea figura 2.2, donde $p = (x, y) \in C(r, q)$ se parametriza en función de $\theta \in [0, 2\pi)$, debido a que $\sin \theta = y/r$ y $\cos \theta = x/r$, considerando que el ángulo θ pertenece a un triángulo rectángulo con catetos x, y e hipotenusa r . Por tanto, para $q = (0, 0)$, se obtiene que

$$x = r \cos \theta , \quad y = r \sin \theta .$$

Para el caso general de un punto centro $q = (h, k)$, se obtienen las siguientes **ecuaciones paramétricas de un círculo** con radio r :

$$x = r \cos \theta + h , \quad y = r \sin \theta + k , \quad \theta \in [0, 2\pi) .$$

Notemos que aquí hay dos tipos de parámetros: los parámetros característicos de la curva que son las coordenadas h, k del punto centro, y el radio r , y un parámetro libre θ que sirve para generar todos los puntos de la curva.

2.2. Elipse

2.2.1. Definición geométrica

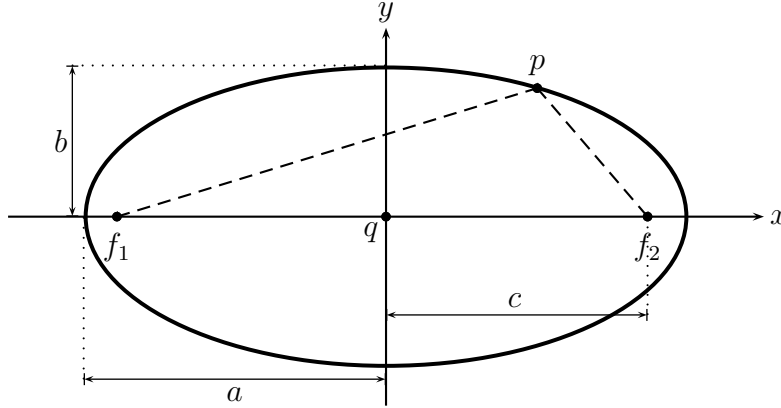


Figura 2.3: Definición de una elipse.

Una elipse es definida como un conjunto de puntos en el plano \mathbb{R}^2 , que generan una curva plana cerrada, considerando dos puntos fijos denominados focos, f_1 y f_2 , donde, para cualquier punto $p = (x, y)$ sobre la curva, la suma de las distancias de p a f_1 y de p a f_2 , es constante e igual a un número $2a$. De manera formal, utilizando la distancia euclidiana d y suponiendo $a \in \mathbb{R}, a \geq 0$, una **elipse** es dada como

$$E = \{p \in \mathbb{R}^2 : d(p, f_1) + d(p, f_2) = 2a\} .$$

Una elipse es una curva en forma ovalada con dos ejes de simetría, ver figura 2.3: su *eje mayor* contiene ambos focos y tiene longitud $2a$, y su *eje menor* de longitud $2b$, donde $0 \leq b \leq a$. Ambos ejes se intersectan de manera perpendicular en su punto centro q , llamado el *centro* de la elipse.

Una elipse E tiene las siguientes características (vea figura 2.4):

- * El número $c = d(f_1, q) = d(q, f_2)$ es llamado *semi-distancia focal* de E , $2c$ es la *distancia focal*, obviamente $0 \leq c < a$.
- * La *excentricidad* de E es definida como $\epsilon = c/a$, claro que $0 \leq \epsilon \leq 1$. Una elipse será más redonda si ϵ se aproxima a cero. El caso extremo $\epsilon = 0$ se da para $c = 0$, cuando los puntos focos coinciden con el centro, $f_1 = f_2 = q$, entonces E es un círculo con el radio $a = b$.
- * Los vértices v_1, v_2, v_3, v_4 son los puntos donde se intersectan el eje mayor y el eje menor con la elipse. El rectángulo $[v_2, v_1] \times [v_3, v_4]$ circunscribe la elipse.
- * El *lado recto* es la magnitud de un segmento paralelo al eje menor y que atraviesa a un punto foco. Existen dos tales segmentos de la misma magnitud, representados por $\overline{r_1 r_2}$ y $\overline{u_1 u_2}$.

* La *directriz* es una línea recta paralela al eje menor y asociada a cada foco. La relación entre la directriz y el foco, está dada por una razón que es constante y es igual a la excentricidad para cualquier punto $p \in E$: $\epsilon = d(p, f_2)/d(p, s)$, donde s es un punto en la directriz.

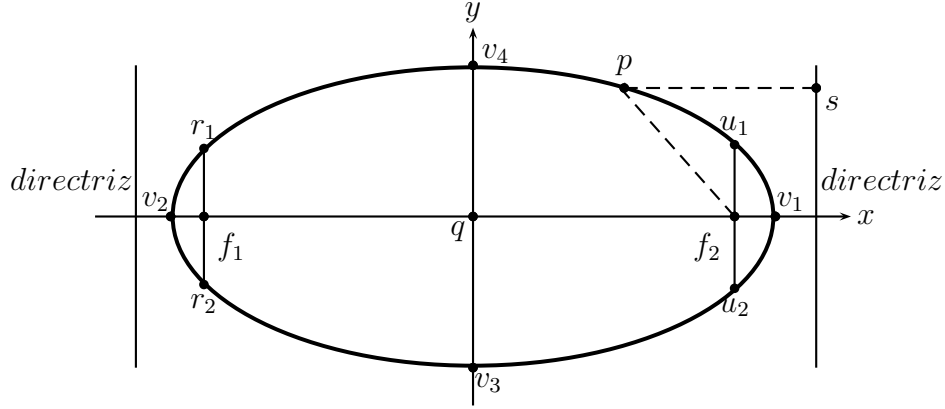


Figura 2.4: Propiedades de una elipse.

Es bien conocido que el perímetro de una elipse puede ser calculado aproximadamente por $\pi[3(a+b) - \sqrt{(3a+b)(a+3b)}]$, mientras que el área está dada por $A = \pi ab$ [38].

2.2.2. Representación algebraica

La ecuación algebraica de la elipse puede ser deducida de su definición $E = \{p \in \mathbb{R}^2 : d(p, f_1) + d(p, f_2) = 2a\}$. Considerando primero el punto centro $q = (0, 0)$ y que el eje mayor se encuentra sobre el eje de coordenadas x , además recordando que c es la semi-distancia focal, se tiene lo siguiente (ver figura 2.3):

$$\sqrt{(x-c)^2 + y^2} + \sqrt{(x+c)^2 + y^2} = 2a .$$

Despejando y elevando al cuadrado obtenemos:

$$\begin{aligned} (x-c)^2 + y^2 &= 4a^2 - 4a\sqrt{(x+c)^2 + y^2} + (x+c)^2 + y^2 \\ x^2 - 2xc + c^2 + y^2 &= 4a^2 - 4a\sqrt{(x+c)^2 + y^2} + x^2 + 2xc + c^2 + y^2 \\ -4xc &= 4a^2 - 4a\sqrt{(x+c)^2 + y^2} \\ -xc &= a^2 - a\sqrt{(x+c)^2 + y^2} \\ xc + a^2 &= a\sqrt{(x+c)^2 + y^2} \end{aligned}$$

Nuevamente, elevando al cuadrado y, al final, factorizando, se obtiene:

$$\begin{aligned}
(xc + a^2)^2 &= (a\sqrt{(x+c)^2 + y^2})^2 \\
x^2c^2 + 2a^2xc + a^4 &= a^2x^2 + 2a^2xc + a^2c^2 + a^2y^2 \\
x^2c^2 + a^4 &= a^2x^2 + a^2c^2 + a^2y^2 \\
a^4 - a^2c^2 &= a^2x^2 - x^2c^2 + a^2y^2 \\
a^2(a^2 - c^2) &= x^2(a^2 - c^2) + y^2a^2
\end{aligned}$$

Luego, dividiendo entre $a^2(a^2 - c^2)$:

$$\frac{x^2}{a^2} + \frac{y^2}{(a^2 - c^2)} = 1 .$$

El punto p de la figura 2.5, en particular, satisface la definición de la elipse, pero también tiene la propiedad especial de que $d(p, f_1) = d(p, f_2) = a$. Considerando al triángulo $\triangle pqf_2$, se obtiene entonces que $b^2 = a^2 - c^2$. Eso implica

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1 .$$

Cuando el centro $q = (h, k)$ y toda la elipse están desplazados, se obtiene la siguiente **ecuación algebraica de la elipse**:

$$\frac{(x - h)^2}{a^2} + \frac{(y - k)^2}{b^2} = 1 .$$

Como una variante, si el eje mayor de la elipse está sobre el eje y , y $q = (0, 0)$, entonces la elipse tiene la ecuación

$$\frac{y^2}{a^2} + \frac{x^2}{b^2} = 1 ,$$

o más generalmente, para un centro arbitrario $q = (h, k)$:

$$\frac{(y - k)^2}{a^2} + \frac{(x - h)^2}{b^2} = 1 .$$

La ecuación algebraica de la elipse puede ser escrita también de la siguiente forma:

$$ex^2 + fy^2 - gx - my + n = 0 ,$$

donde e, f, g, m, n son constantes que conforman una ecuación algebraica en dos variables de segundo orden.

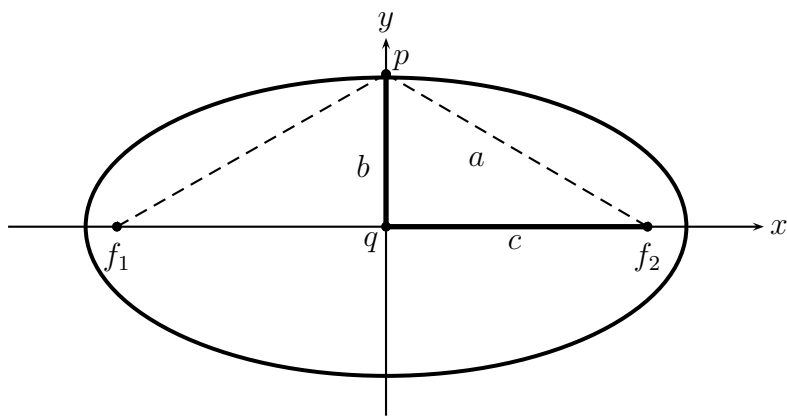


Figura 2.5: Deducción de la ecuación algebraica de la elipse.

2.2.3. Representación paramétrica

Como el círculo, la elipse es una curva y por eso es la imagen continua de un intervalo I , bajo una función definida sobre I y con valores en \mathbb{R}^2 .

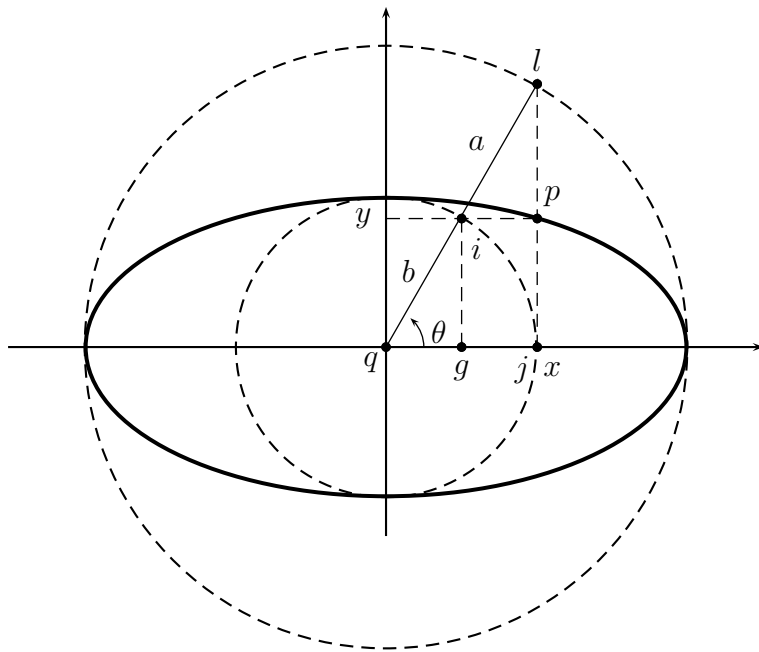


Figura 2.6: Una elipse con sus círculos que la inscriben y la suscriben.

Observemos en la figura 2.6 una elipse E con centro q , eje mayor de longitud $2a$, eje menor de longitud $2b$ y los círculos centrados en q con radios a y b . Tomando en cuenta el

punto $p = (x, y) \in E$ y suponiendo $q = (0, 0)$, se puede parametrizar la elipse utilizando al ángulo $\theta \in [0, 2\pi)$. Para la obtención de las componentes x del punto p , se considera la función trigonométrica $\cos \theta = x/a$ en el triángulo $\triangle qjl$. Para la componente y , se utiliza la función $\sin \theta = y/b$ en el triángulo $\triangle qgi$. Así, se obtienen las siguientes ecuaciones:

$$x = a \cos \theta , \quad y = b \sin \theta .$$

Para el caso más general si $q = (h, k)$, cada punto (x, y) de la elipse tiene que ser trasladado por el vector (h, k) . En consecuencia, las siguientes ecuaciones constituyen la **representación paramétrica de una elipse** con un eje mayor de longitud $2a$ y situada sobre el eje x , y un eje menor de longitud $2b$:

$$x = a \cos \theta + h , \quad y = b \sin \theta + k , \quad \theta \in [0, 2\pi) .$$

Notemos aquí que los parámetros son de dos tipos: las coordenadas h, k del punto centro y los semi-distancias focales a, b son los parámetros característicos de la curva, mientras que θ es un parámetro libre que sirve para generar a todos los puntos de la curva. Es importante notar que en general, una elipse puede ser rotada, entonces su inclinación con respecto al eje x sería el quinto parámetro característico de esta curva, el cual no está contemplado en nuestras ecuaciones arriba.

Capítulo 3

Detección de círculos y elipses mediante acumulación de evidencia en espacios de parámetros

3.1. Método general de la transformada de Hough

La transformada de Hough es una herramienta para la detección de figuras en imágenes digitales, la cual fue diseñada y patentada por Paul Hough en 1962 [18]. La transformada de Hough esencialmente fue propuesta para la detección de líneas rectas. Pero dada su gran utilidad en el procesamiento de imágenes, posteriormente fue generalizada para la detección de círculos y elipses, y llamada entonces transformada de Hough Generalizada [3].

Se supone que la figura de interés que debe ser detectada, está representada en el espacio imagen por puntos con coordenadas x, y , las cuales satisfacen una ecuación paramétrica o algebraica con pocos parámetros. Al utilizar la transformada de Hough, la figura es mapeada del espacio imagen al *espacio de parámetros*, llamado *espacio de Hough*, cuya dimensión coincide con el número de parámetros; cada parámetro corresponde a una coordenada en el espacio de Hough. Cada punto de la imagen que pertenece a la figura a detectar, en el espacio de Hough es considerado como la representación de todas las rectas, circunferencias o elipses que podrían pasar por ahí. Eso significa fijar al punto (x, y) pero considerar libres los parámetros de la figura en su ecuación. Entonces es generada una curva en el espacio de Hough. Cuando se intersecan dos curvas en el espacio de Hough es porque existe una coincidencia de parámetros entre estas dos curvas, es decir, los puntos de la imagen que generaron estas curvas, pertenecen a la misma línea recta, círculo o elipse en el espacio imagen. Los parámetros dados por el punto de intersección en el espacio de Hough, son analizados para corroborar si realmente representan a la figura de interés.

Para llevar a cabo este procedimiento en la práctica, primero, el espacio de Hough es subdividido en porciones formando las celdas de un arreglo denominado acumulador. Los valores de cada parámetro son considerados dentro de algún intervalo acotado. Este intervalo

es cuantizado en subintervalos. Procediendo así para cada parámetro, el espacio de Hough es descompuesto en celdas, por ejemplo, en rectángulos si este espacio tiene dimensión dos. Dicha descomposición o arreglo es representado por una matriz, asociando un elemento de la matriz a cada celda. Esta tal llamada **matriz acumulador** Ac es iniciada con el valor 0 para todos sus elementos.

La detección de la figura en una imagen digital inicia con un proceso de selección de píxeles que son candidatos a pertenecer a esta figura. Para cada píxel candidato p , las ecuaciones correspondientes a la figura son consideradas con las coordenadas de p fijadas pero con los parámetros característicos de la figura como libres. Eso genera una curva en el espacio de Hough, la cual corresponde al conjunto de todas las figuras que podrían pasar por el punto p en la imagen. Cada celda en el arreglo del espacio de Hough registra cuando alguna de las curvas generadas pasa por ella, y la matriz acumulador contabiliza el número de curvas que pasan por esta misma celda; cada vez un tal llamado *voto* es sumado. Al final, los máximos locales de los valores de la matriz Ac son determinados. Los parámetros específicos representados por una celda cuyo valor es uno de estos máximos, corresponden a una figura de mucha evidencia en la imagen, pues debe contener a muchos puntos candidatos.

La transformada de Hough originalmente fue desarrollada para detectar líneas rectas en imágenes digitales. La siguiente sección explica esta aplicación.

3.2. Detección de líneas mediante la transformada de Hough

La transformada de Hough para la detección de líneas rectas puede ser basada en la representación $y = mx + b$, sin embargo, para líneas verticales la pendiente m no está definida. Por eso, es más conveniente el uso de una representación paramétrica:

$$\rho = x \cos \theta + y \sin \theta ,$$

donde ρ es la distancia perpendicular del punto origen de la imagen a la línea recta, x, y son las coordenadas de un punto de la línea recta en el espacio imagen, θ es el ángulo entre la línea perpendicular a la línea recta y el eje x . El ángulo, de manera estándar y en un sistema de coordenadas cartesianas, ver figura 3.1(b), se mide en sentido matemático positivo, es decir, con respecto al sentido contrario del movimiento de las manecillas del reloj. Sin embargo, en algunos paquetes de programas computacionales, como MATLAB, el sistema de coordenadas tiene su origen en la esquina superior izquierda, ver figura 3.1(a), entonces la medición del ángulo se realiza en el sentido del movimiento de las manecillas del reloj.

Así, cada línea recta en el espacio imagen está definida por los dos parámetros θ, ρ . La pareja (θ, ρ) define un punto en el espacio de Hough el cual tiene θ, ρ como coordenadas. Cada punto (x, y) en el espacio imagen es mapeado en forma de una senoide al espacio de parámetros, como se muestra en figura 3.2. Para generar esta curva sinusoidal en el espacio de parámetros, el punto (x, y) es considerado como elemento de todas las líneas rectas que

podrían pasar por él. Es decir, en la ecuación (4), x, y son fijados (conocidos) pero se asumen libremente valores para θ dentro del intervalo $[0, \pi]$ y se calcula el valor de ρ para cada tal θ .

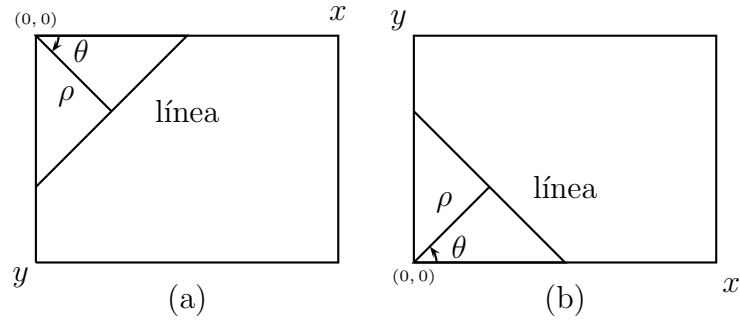


Figura 3.1: Representación paramétrica de una línea recta: (a) en MATLAB, (b) en un sistema de coordenadas estándar.

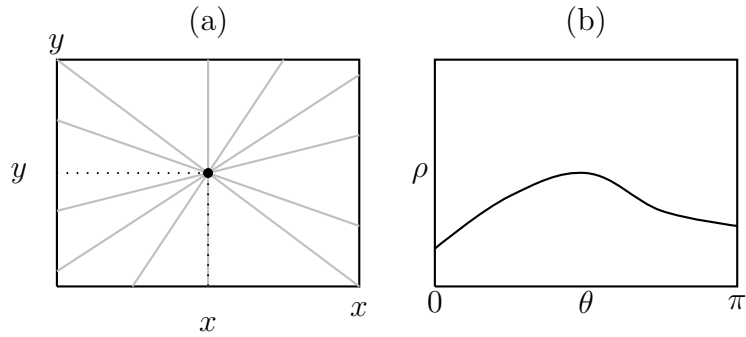


Figura 3.2: Posibles líneas rectas que pasan por un punto (x, y) : (a) en el espacio imagen, (b) su representación en el espacio de parámetros.

Si dos puntos en el espacio imagen pertenecen a la misma línea recta que tiene los parámetros θ, ρ , entonces, el mapeo al espacio de parámetros produce dos sinusoides que se interceptan en las coordenadas (θ, ρ) , ver figura 3.3. Para muchos puntos sobre una línea recta, el mapeo al espacio de parámetros da como resultado muchas sinusoides (una senoide por cada punto), las cuales, todas se intersecan en el mismo punto (θ, ρ) .

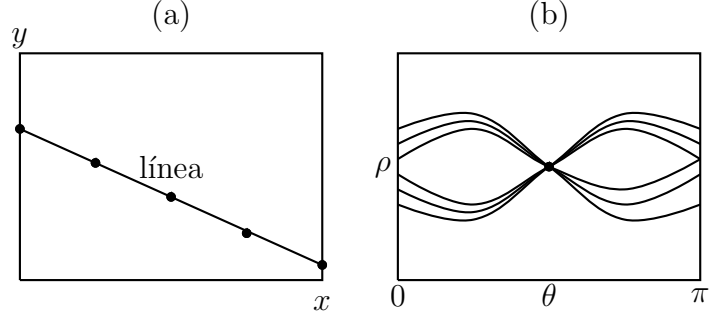


Figura 3.3: Varios puntos de una misma línea (a) generan, en el espacio de parámetros (b), un conjunto de curvas sinusoidales, llamado en la literatura mariposa, con un solo punto de intersección.

El procedimiento práctico para la detección de líneas rectas en una imagen digital por medio de la transformada de Hough, se lleva a cabo de la siguiente manera:

Paso 1:

Se subdivide el espacio de los dos parámetros θ, ρ en celdas rectangulares para la creación de la matriz acumulador Ac . Para eso, se toma en cuenta que $\theta \in [0, \pi]$ y $\rho \in [0, \rho_{\text{máx}}]$, donde $\rho_{\text{máx}}$ depende del tamaño de la imagen en la cual las líneas rectas deben ser detectadas y del problema de aplicación a resolver. Los intervalos $[0, \pi]$ y $[0, \rho_{\text{máx}}]$ son divididos en porciones de la misma longitud, es decir, para números naturales convenientes m y n , la cuantización puede ser como sigue:

$$\begin{aligned} [0, \pi] &= \left[0, \frac{1}{m}\pi\right] \cup \left[\frac{1}{m}\pi, \frac{2}{m}\pi\right] \cup \left[\frac{2}{m}\pi, \frac{3}{m}\pi\right] \cup \dots \cup \left[\frac{m-1}{m}\pi, \pi\right], \\ [0, \rho_{\text{máx}}] &= \left[0, \frac{1}{n}\rho_{\text{máx}}\right] \cup \left[\frac{1}{n}\rho_{\text{máx}}, \frac{2}{n}\rho_{\text{máx}}\right] \cup \dots \cup \left[\frac{n-1}{n}\rho_{\text{máx}}, \rho_{\text{máx}}\right]. \end{aligned}$$

Eso genera una subdivisión del espacio de parámetros:

$$[0, \pi] \times [0, \rho_{\text{máx}}] = \bigcup_{\substack{i=1, \dots, m \\ j=1, \dots, n}} \left[\frac{i-1}{m}\pi, \frac{i}{m}\pi \right] \times \left[\frac{j-1}{n}\rho_{\text{máx}}, \frac{j}{n}\rho_{\text{máx}} \right]$$

Cada celda rectangular $\left[\frac{i-1}{m}\pi, \frac{i}{m}\pi \right] \times \left[\frac{j-1}{n}\rho_{\text{máx}}, \frac{j}{n}\rho_{\text{máx}} \right]$ será denotada simplemente por (i, j) . Estas mismas coordenadas i, j , son usadas como índices en la matriz Ac de m columnas y n filas, ver figura 3.4, con valores iniciales $Ac(i, j) = 0$ para todos los $i = 1, \dots, m, j = 1, \dots, n$.

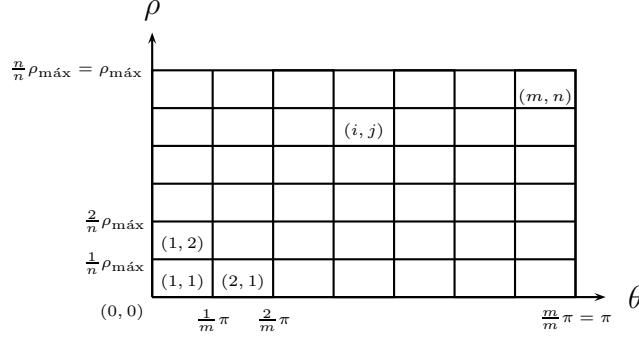


Figura 3.4: Arreglo y matriz acumulador Ac del espacio de Hough.

Paso 2

Para cada punto (x, y) en la imagen, el cual es candidato para pertenecer a alguna línea recta de interés, se genera una curva sinusoidal en base a la ecuación (4) para x, y fijos: para un valor de θ representativo de cada subintervalo de $[0, \pi]$, por ejemplo, $\theta = \frac{i}{m}\pi$, se calcula el valor correspondiente ρ . Si el valor de ρ cae dentro del intervalo $[\frac{j-1}{n}\rho_{\text{máx}}, \frac{j}{n}\rho_{\text{máx}}]$, entonces, la celda (i, j) suma un voto, es decir, $Ac(i, j)$ es aumentado por 1.

Paso 3:

Por último, se buscan los máximos locales de los valores $Ac(i, j)$. Estos máximos corresponden a las celdas más votadas dentro del arreglo, y consecuentemente, estas celdas representan a los parámetros de las líneas encontradas en la imagen.

3.3. Detección de círculos

3.3.1. Método general (Hough 3D) para círculos

La transformada de Hough para la detección de círculos utiliza la siguiente representación paramétrica, introducida en la sección 2.1.3, dada por

$$x = r \cos \theta + h, \quad y = r \sin \theta + k, \quad \theta \in [0, 2\pi).$$

Aquí, $\theta \in [0, 2\pi)$ es un parámetro libre que genera la curva, y r, h, k son los parámetros característicos del círculo, siendo r su radio y (h, k) su punto centro. En consecuencia, el espacio de Hough es de dimensión tres, y cada círculo en la imagen es mapeado al espacio de Hough hacia un solo punto (h, k, r) , vea figura 3.5.

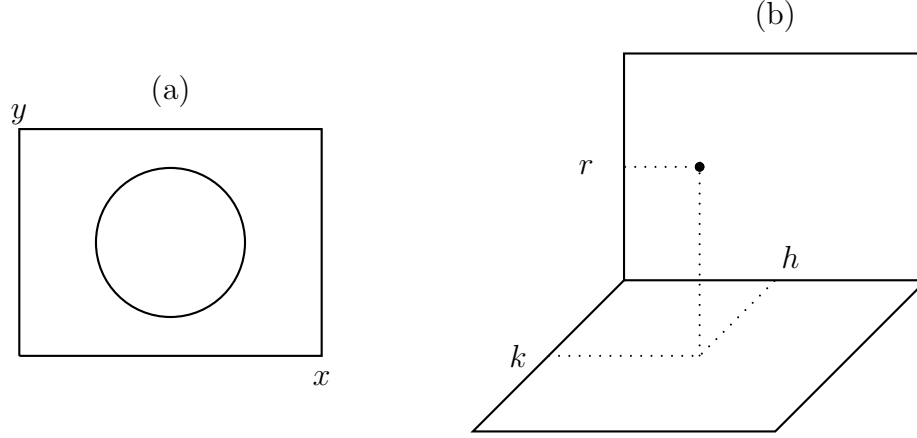


Figura 3.5: Representación de un círculo: (a) cuando sus tres parámetros (h, k, r) son conocidos, y, (b) en el espacio de Hough 3D.

El algoritmo general para la detección de círculos a través de la transformada de Hough tiene como objetivo determinar tripletas (h, k, r) de círculos presentes en la imagen, a partir de un conjunto de puntos candidatos. Cada punto candidato (x, y) de la imagen es mapeado al espacio de Hough en forma de un cono, puesto que es un conjunto de círculos con diferentes radios, ver figura 3.6. Eso se basa en considerar la ecuación paramétrica del círculo con x, y fijos (conocidos) pero r, h, k libres y θ recorriendo valores en $[0, 2\pi)$. Eso equivale a interpretar al punto $p = (x, y)$ como elemento de todas las posibles circunferencias en la imagen. Cuando consideramos un radio r fijado, es decir, observamos al plano (h, k) del espacio Hough a la altura r , el conjunto de los centros de todas las circunferencias con este mismo radio r que pueden contener a p , forma una circunferencia de radio r con p como punto centro. Esta circunferencia precisamente es el tal llamado **círculo de votos** en el plano Hough. En consecuencia, p es el centro de todos los círculos de votos, para todos los radios bajo consideración. Por eso, en el espacio 3D de Hough, p es el punto base del cono recto formado por todos los círculos de votos.

La figura 3.6 (a) muestra un punto $p = (x, y)$ en el plano imagen que pertenece a alguna circunferencia con radio y centro desconocidos, el cual es considerado como elemento de todas las circunferencias que podrían pasar por él. En la parte (b) de la figura, este conjunto de circunferencias fue mapeado al espacio 3D de Hough donde genera un cono recto con punto base p .

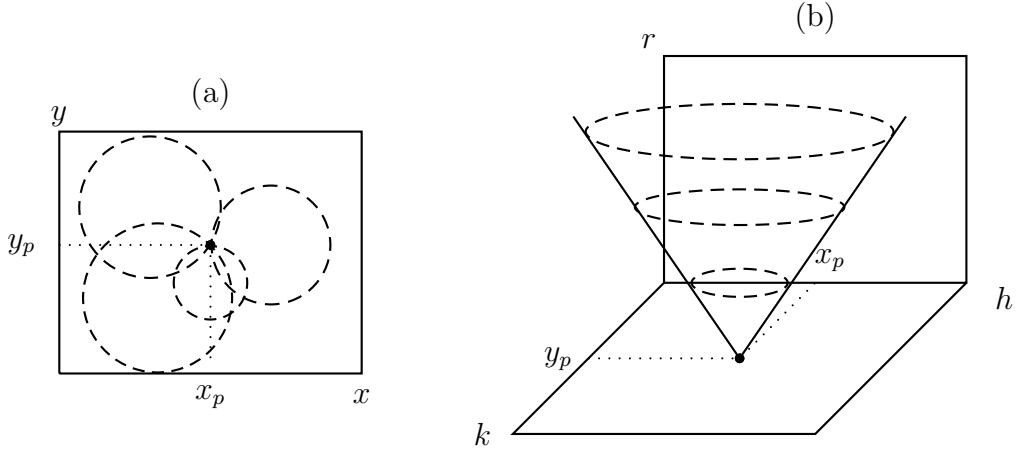


Figura 3.6: Un punto en el plano imagen y su representación en el espacio de Hough 3D, vea texto.

El procedimiento práctico para la detección de círculos por medio de la transformada de Hough en una imagen digital sigue los siguientes pasos:

Paso 1

Se subdivide el espacio de parámetros en celdas para la creación de la matriz acumulador Ac . Los tres parámetros desconocidos pueden ser supuestos como positivos y dentro de rangos acotados: existen números $h_{\text{máx}}, k_{\text{máx}}, r_{\text{máx}}$ tales que $h \in [0, h_{\text{máx}}]$, $k \in [0, k_{\text{máx}}]$ y $r \in [0, r_{\text{máx}}]$. Estos intervalos son divididos en porciones de la misma longitud: para números naturales convenientes l, m, n , la cuantización puede ser como sigue:

$$\begin{aligned}
 [0, h_{\text{máx}}] &= \left[0, \frac{1}{l}h_{\text{máx}}\right] \cup \left[\frac{1}{l}h_{\text{máx}}, \frac{2}{l}h_{\text{máx}}\right] \cup \dots \cup \left[\frac{l-1}{l}h_{\text{máx}}, h_{\text{máx}}\right], \\
 [0, k_{\text{máx}}] &= \left[0, \frac{1}{m}k_{\text{máx}}\right] \cup \left[\frac{1}{m}k_{\text{máx}}, \frac{2}{m}k_{\text{máx}}\right] \cup \dots \cup \left[\frac{m-1}{m}k_{\text{máx}}, k_{\text{máx}}\right], \\
 [0, r_{\text{máx}}] &= \left[0, \frac{1}{n}r_{\text{máx}}\right] \cup \left[\frac{1}{n}r_{\text{máx}}, \frac{2}{n}r_{\text{máx}}\right] \cup \dots \cup \left[\frac{n-1}{n}r_{\text{máx}}, r_{\text{máx}}\right].
 \end{aligned}$$

Eso genera una subdivisión del espacio de parámetros en paralelepípedos:

$$[0, h_{\text{máx}}] \times [0, k_{\text{máx}}] \times [0, r_{\text{máx}}] = \bigcup_{\substack{g=1, \dots, l \\ i=1, \dots, m \\ j=1, \dots, n}} (g, i, j),$$

donde (g, i, j) denota una celda:

$$(g, i, j) = \left[\frac{g-1}{l} h_{\text{máx}}, \frac{g}{l} h_{\text{máx}} \right] \times \left[\frac{i-1}{m} k_{\text{máx}}, \frac{i}{m} k_{\text{máx}} \right] \times \left[\frac{j-1}{n} r_{\text{máx}}, \frac{j}{n} r_{\text{máx}} \right]$$

Estas mismas coordenadas g, i, j son usadas como índices para la matriz acumulador Ac , ver figura 3.7, con valores iniciales $Ac(g, i, j) = 0$ para todos los g, i, j .

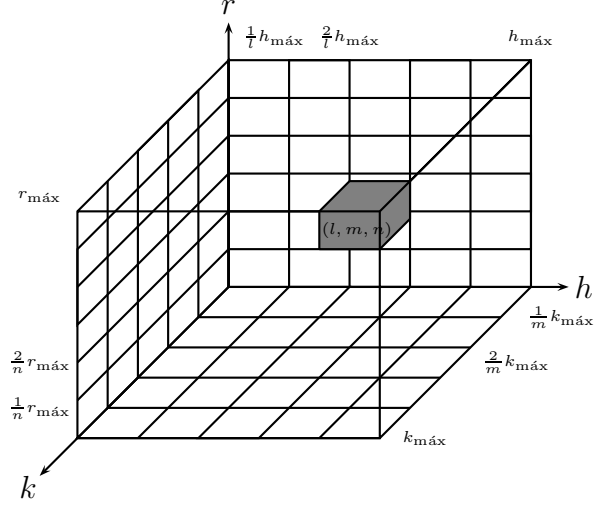


Figura 3.7: División del espacio de Hough 3D en celdas correspondientes a los elementos de la matriz acumulador Ac .

Paso 2

Para cada punto candidato (x, y) en la imagen, se generan los círculos que forman el cono en el espacio de Hough mediante la variación de los radios y, para cada radio fijado de prueba, la asunción de los parámetros (h, k) como libres. Si un punto del cono dado por h, k, r cae dentro de una celda (g, i, j) , esta celda suma un voto, es decir, $Ac(g, i, j)$ es aumentado por uno.

Paso 3

Por último, se buscan los picos de la matriz acumulador, es decir, sus valores máximos locales, los cuales sugieren los potenciales parámetros de los círculos presentes en la imagen.

El ejemplo de la figura 3.8 muestra al espacio de Hough cuando h, k, r son desconocidos y se tiene información de tres puntos u, v, w sobre la circunferencia buscada. Los puntos en el espacio de parámetros generan conos con su centro correspondiente a las coordenadas de éstos puntos, además, se pueden observar círculos con diferentes radios a través de los tres

conos. Cuando circunferencias generadas por los puntos u, v, w se intersecan, se sugiere que las coordenadas obtenidas de la intersección en (h, k, r) son los parámetros de la posible circunferencia a encontrar en la imagen original.

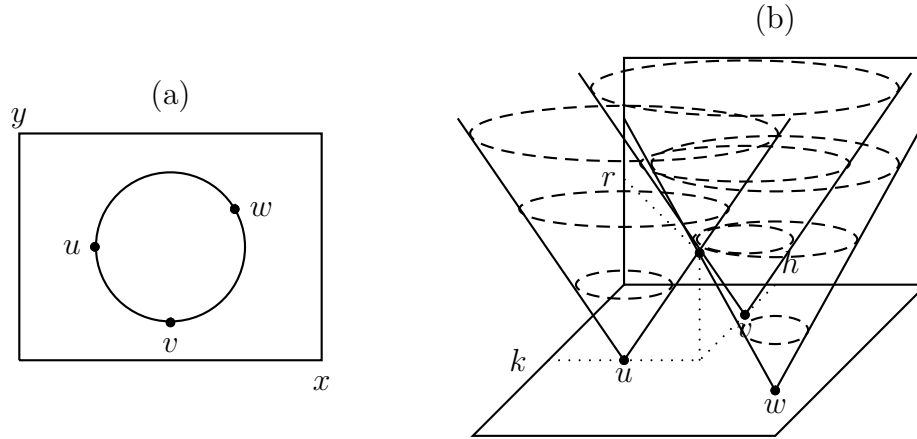


Figura 3.8: Tres puntos en la imagen sobre una circunferencia (a), y los conos correspondientes en el espacio de Hough (b), vea texto.

3.3.2. Método para radio conocido (Hough 2D)

Cuando se conoce el radio de los círculos a buscar, ya no es necesario generar a partir de cada punto candidato al cono completo en el espacio de Hough 3D, vea figura 3.6, sino solamente la intersección del cono con un plano horizontal a la altura de un radio r fijo; eso es un círculo. Es decir, el espacio de Hough es de dimensión dos, teniendo solamente las coordenadas h, k . El plano de Hough es dividido en celdas como fue explicado en la sección anterior; las celdas ahora son rectangulares y cada celda (g, i) corresponde a un elemento de la matriz acumulador Ac .

Para cada punto candidato (x, y) en la imagen, usando la representación paramétrica del círculo para x, y y r fijos, quedando solamente los parámetros h, k como libres, se usan valores $\theta \in [0, 2\pi)$ para calcular puntos (h, k) :

$$h = x - r \cos \theta, \quad k = y - r \sin \theta, \quad \theta \in [0, 2\pi).$$

Estos puntos (h, k) generan una curva en el plano de Hough, la cual definitivamente resulta ser un círculo, con radio r y centro (x, y) , vea figura 3.9. Si un punto de este círculo cae dentro de una celda (g, i) , esta celda suma un voto, es decir, $Ac(g, i)$ es aumentado por 1.

Al final, los máximos locales en la matriz acumulador son determinados, obteniendo así los parámetros h, k de las circunferencias con más evidencia en la imagen.

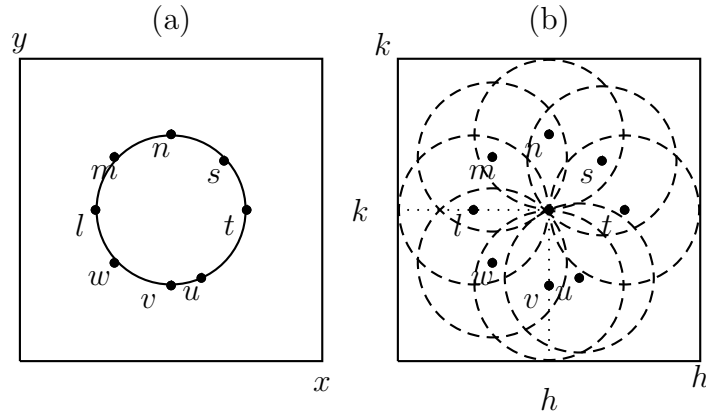


Figura 3.9: Ejemplo de varios puntos candidatos pertenecientes a la misma circunferencia en la imagen (a), y los círculos generados por ellos en el plano de Hough (b). El punto de intersección de todos estos círculos determina las coordenadas del centro de la circunferencia buscada en la imagen.

3.3.3. Método para centro conocido (Hough 1D)

Si se conoce la localización (h, k) del centro del círculo a buscar en la imagen, solamente falta detectar al radio. En el espacio 3D de Hough, los valores posibles del radio en un intervalo $[0, r_{\text{máx}}]$, forman un segmento de línea recta que inicia en el centro (h, k) y que es perpendicular al plano (h, k) , vea figura 3.10. Al ser sólo un parámetro a detectar, tanto el espacio de Hough de interés como también la matriz acumulador, tiene dimensión uno.

Para un punto candidato en la imagen, su correspondiente curva de votos en el espacio de Hough, ya solamente es un punto dentro de la recta de valores de r . Aplicando el método general, esta línea recta de Hough es dividida en celdas correspondientes a n subintervalos que descomponen al intervalo de valores $[0, r_{\text{máx}}]$. Cada celda (j) corresponde a un elemento $Ac(j)$; ahora la matriz acumulador simplemente es un vector el cual, al momento de acumular los valores r , capta la información de evidencia de ciertos radios de círculos presentes en la imagen.

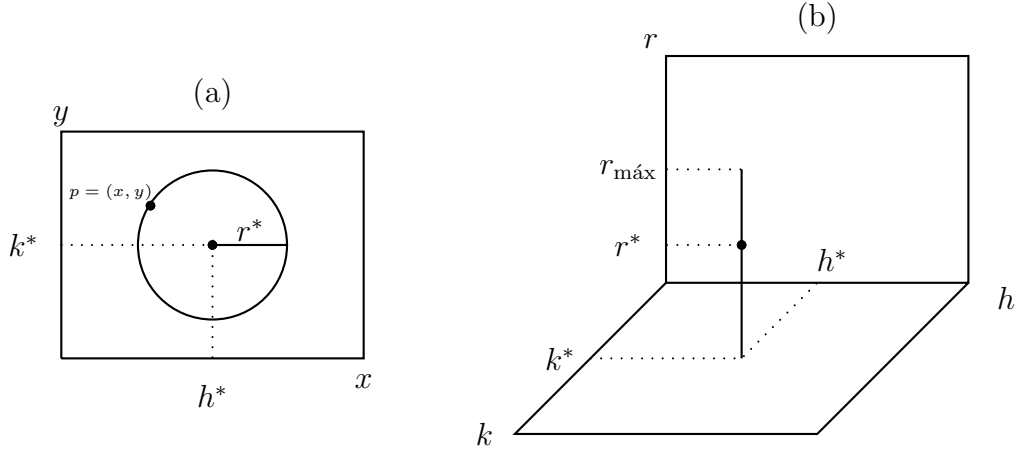


Figura 3.10: El punto p sobre la circunferencia en (a) es mapeado al espacio de Hough 3D (b), donde es representado por un punto del segmento de recta correspondiente a valores r en el intervalo $[0, r_{\text{máx}}]$.

Para cada punto candidato $p = (x, y)$ en la imagen, el radio r correspondiente a p , el cual debe ser registrado en la matriz Ac , es calculado como la distancia de p al centro conocido (h, k) , es decir, $r = d(p, (h, k))$. Si el valor calculado r cae dentro de una celda (j) , esta celda suma un voto. El vector $(Ac(1), Ac(2), \dots, Ac(n))$ representa un histograma de radios cuyos máximos locales indican cuáles radios son los de circunferencias, todas con el centro prefijado (h, k) , que aparecen con más evidencia en la imagen original. Cuando los máximos locales de este histograma de radios no son fácilmente determinables, se recomienda aplicar algún filtro que los acentúa. Por ejemplo, en [19], el filtro lineal dado por la siguiente máscara fue propuesto para este fin:

$$t = \frac{1}{4\sqrt{2}r} \begin{bmatrix} \frac{-3r}{2(r-2)} & 1 & 1 & 1 & \frac{-3r}{2(r+2)} \end{bmatrix},$$

donde r es el valor de radio leído del histograma. Este filtro es aplicado mediante convolución del histograma original $(Ac(1), Ac(2), \dots, Ac(n))$ con t , para generar un nuevo vector $(Ac'(1), Ac'(2), \dots, Ac'(n))$:

$$Ac'(i) = \frac{1}{4\sqrt{2}r} \left(\frac{-3r}{2(r-2)} Ac(i-2) + Ac(i-1) + Ac(i) + Ac(i+1) + \frac{-3r}{2(r+2)} Ac(i+2) \right).$$

El nuevo vector representa a un histograma modificado, en el cual, supuestamente, los máximos locales son más fáciles de detectar.

Esta variante simple del método basada en la transformada de Hough, donde el punto centro de la circunferencia a buscar, es conocido, es apropiada, por ejemplo, para detectar circunferencias concéntricas cuyo punto centro es conocido por conocimiento a priori o como resultado de otras técnicas de análisis aplicadas antes a la imagen.

Además, como veremos más adelante, algunos métodos logran estimar primero las coordenadas del punto centro del círculo, independientemente de su radio. Con el punto centro ya determinado, sólo falta aplicar al método de la presente sección para completar la determinación de los datos del círculo a ser detectado en la imagen.

3.3.4. Método práctico para un rango conocido de radios

Cuando se conoce o se puede estimar un rango estrecho para los radios de las circunferencias de interés presentes en la imagen, la memoria y el tiempo de cómputo necesarios para aplicar al método general de Hough se reducen esencialmente. A pesar de que se desconocen los tres parámetros (h, k, r) , la ventaja de poder limitar a los posibles radios a un intervalo pequeño, permite centralizar la búsqueda hacia circunferencias de interés que tienen un propósito de acuerdo a la aplicación, pasando por alto la existencia de otras circunferencias en la imagen que no se encuentran en el rango del valor del radio. Aunque se limita el valor del radio en un rango, se sigue utilizando una matriz de dimensión 3D. En este caso, el procedimiento útil en la práctica para encontrar a los círculos en la imagen se divide en dos partes:

Primero, un intervalo pequeño $[r_{\min}, r_{\max}]$ de radios es seleccionado, junto con una secuencia finita de números representantes:

$$r_1 = r_{\min} < r_2 < r_3 < \cdots < r_{k-1} < r_k = r_{\max}$$

Se determinan los puntos candidatos en la imagen, es decir, los puntos de borde. Para cada punto candidato $p = (x, y)$ y sólo utilizando uno de los valores r_i ($i \in \{1, 2, \dots, k\}$) a la vez, se realiza el procedimiento del método para radio conocido descrito en la sección 3.3.2.

En la segunda parte, se determinan los máximos locales de la matriz Ac , que son las celdas correspondientes a los parámetros (h, k) de los centros de las circunferencias.

La figura 3.11 (a) muestra un ejemplo de varios puntos candidatos de la misma circunferencia en la imagen, con su mapeo al espacio de Hough en (b). En este ejemplo, se usa el intervalo $[1, 3]$ para el valor del radio, es decir, las posibilidades consideradas para radios enteros r solamente son los números 1, 2, 3. En el plano de Hough, los círculos de los puntos candidatos son mostrados en color negro para el radios $r = 1$, en gris claro para $r = 2$ y en gris oscuro para $r = 3$. La posición con mayor cantidad de votaciones en la matriz acumulador se encuentra en la coordenada (h, k) , la cual determina el centro de la circunferencia.

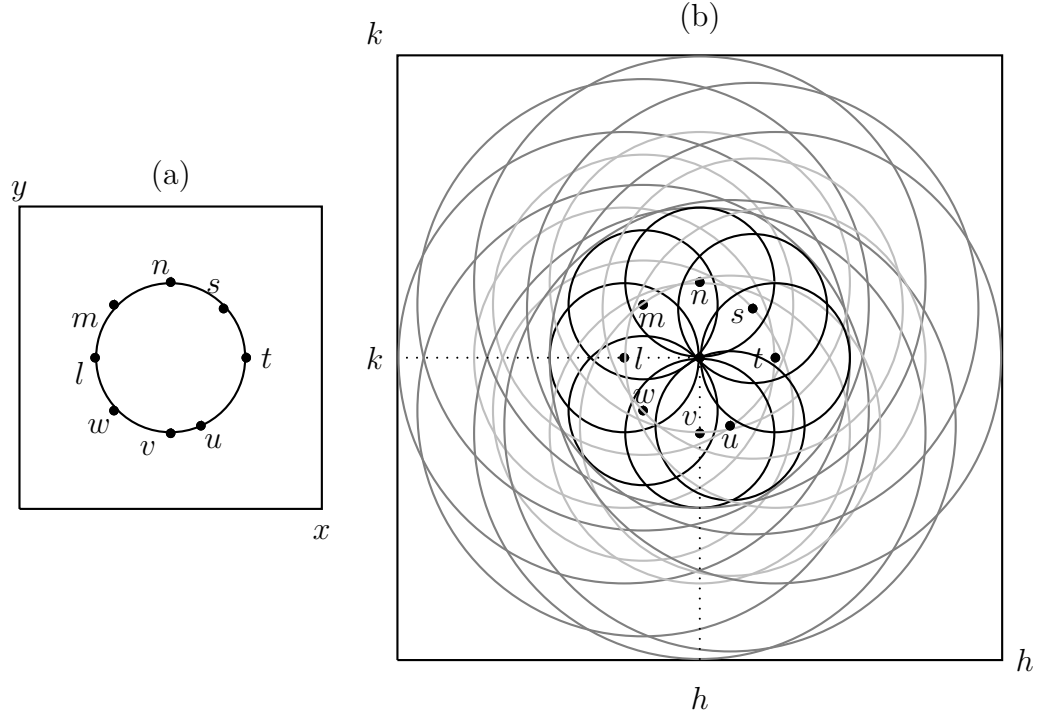


Figura 3.11: Ejemplo de varios puntos candidatos de la misma circunferencia, con su mapeo al espacio de Hough, vea texto.

Otras variantes interesantes se dan cuando se conoce el valor, o intervalos muy restrictivos, de una de las dos variables del centro de las circunferencias a detectar, como se reportará en la siguiente sección.

3.3.5. Variantes para conocimiento sobre el centro

Cuando se tiene conocimiento de intervalos bien especificados para uno de los parámetros del centro (h, k) , o para ambos, se reduce en gran medida el costo computacional para la determinación de los parámetros (h, k, r) . Consideremos tres casos:

Caso 1: se conoce un intervalo para la coordenada h del centro.

Caso 2: se conoce un intervalo para la coordenada k del centro.

Caso 3: se conocen intervalos restrictivos para ambas coordenadas h, k del centro.

En los tres casos, el espacio de Hough aparece delimitado por los intervalos, por lo que, al momento de realizar las votaciones en la matriz acumulador se despreciarían todos los puntos pertenecientes a los conos que no formen parte de los intervalos determinados, vea figura 3.12.

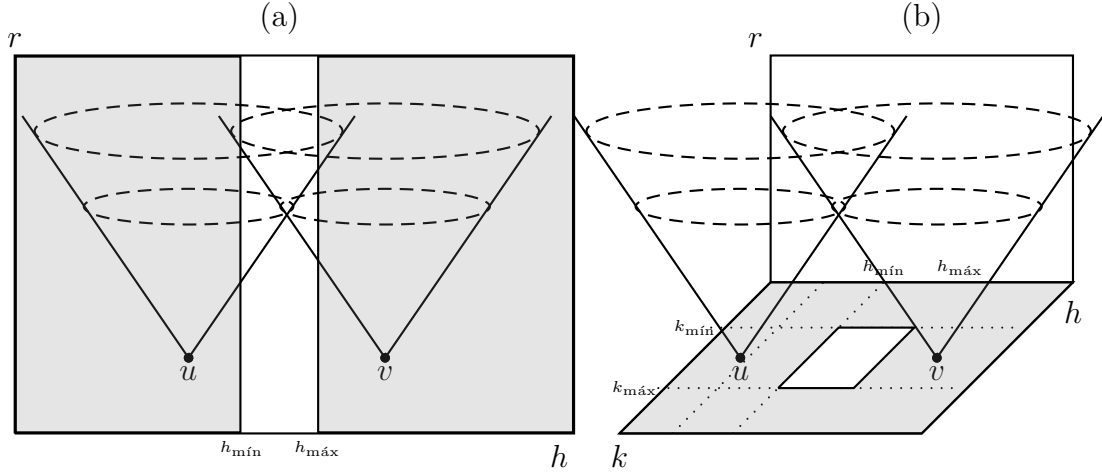


Figura 3.12: Representación de dos puntos en el espacio de Hough. La parte sombreada se desprecia y solamente se considera parte en blanco; (a) espacio de Hough 2D sólo se considera $h \in [h_{\min}, h_{\max}]$, (b) los conos generados en el espacio de Hough 3D únicamente se valoran los puntos dentro de los intervalos $[h_{\min}, h_{\max}]$ y $[k_{\min}, k_{\max}]$.

Para ilustrar las restricciones en el espacio de parámetros en los tres casos, veamos el siguiente ejemplo en la figura 3.13 para el cual $h \in [-2, 2]$ y $k \in [-2, 2]$. En (a), varios puntos candidatos se encuentran sobre la misma circunferencia en la imagen. La parte (b) muestra el Caso 1, representando la situación en el plano de Hough 2D limitando el valor de $h \in [h_{\min}, h_{\max}]$. El Caso 3 puede observarse en (c), donde el plano de Hough está limitado por ambos valores $h \in [h_{\min}, h_{\max}]$ y $k \in [k_{\min}, k_{\max}]$. La parte sombreada no se considera. El caso 2 es análogo al caso 1.

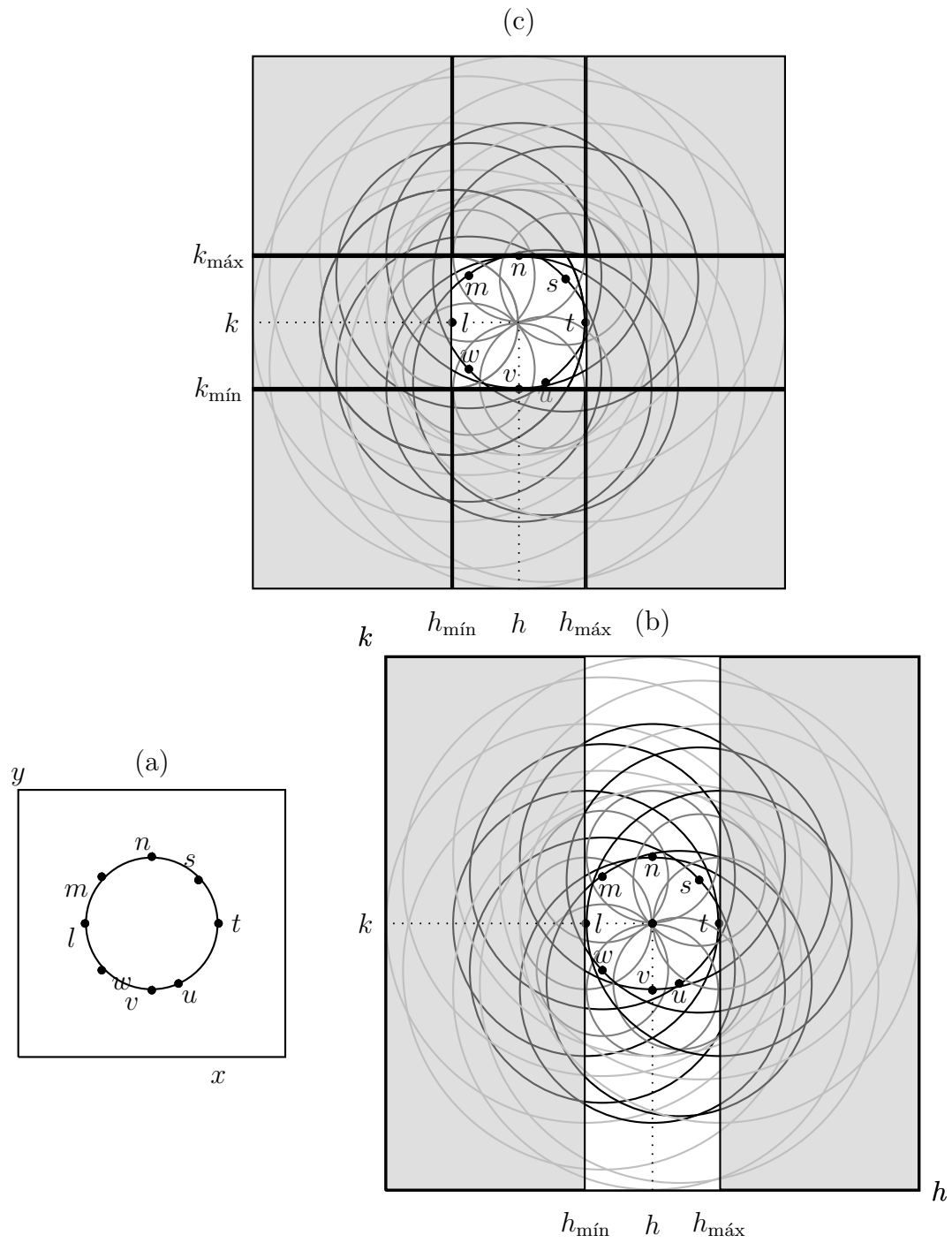


Figura 3.13: Restricción del espacio de Hough por intervalos, vea texto.

Otras variantes muy interesantes son las siguientes:

- h conocido (k, r desconocidos, Hough 2D).
- k conocido (h, r desconocidos, Hough 2D)

En estas dos variantes en las que se conoce el valor exacto de uno de los parámetros del centro, para obtener el otro parámetro del centro, se despeja el valor de r en alguna de las dos ecuaciones paramétricas del círculo y se sustituye en la otra:

$$\begin{aligned} \text{para } h \text{ conocido: } r &= \frac{x - h}{\cos \theta}, \quad k = y - r \sin \theta, \\ \text{para } k \text{ conocido: } r &= \frac{x - k}{\sin \theta}, \quad h = x - r \cos \theta. \end{aligned}$$

Es importante notar que en ambos casos, se deben despreciar valores cuando $\sin(\pi/2) = \sin(3\pi/2) = 0$ y $\cos(\pi) = \cos(2\pi) = 0$, pues la división entre cero no está definida. La representación de un punto $p = (x, y)$ en la imagen al espacio de Hough para estos casos es una curva. Haciendo uso de un programa implementado en MATLAB, se calculó un ejemplo donde $p = (22,75, 21,30)$, $r = 10$, $h = 15$ (para el caso de h conocido) y $k = 15$ (para el caso de k conocido), vea las figuras 3.14 y 3.15.

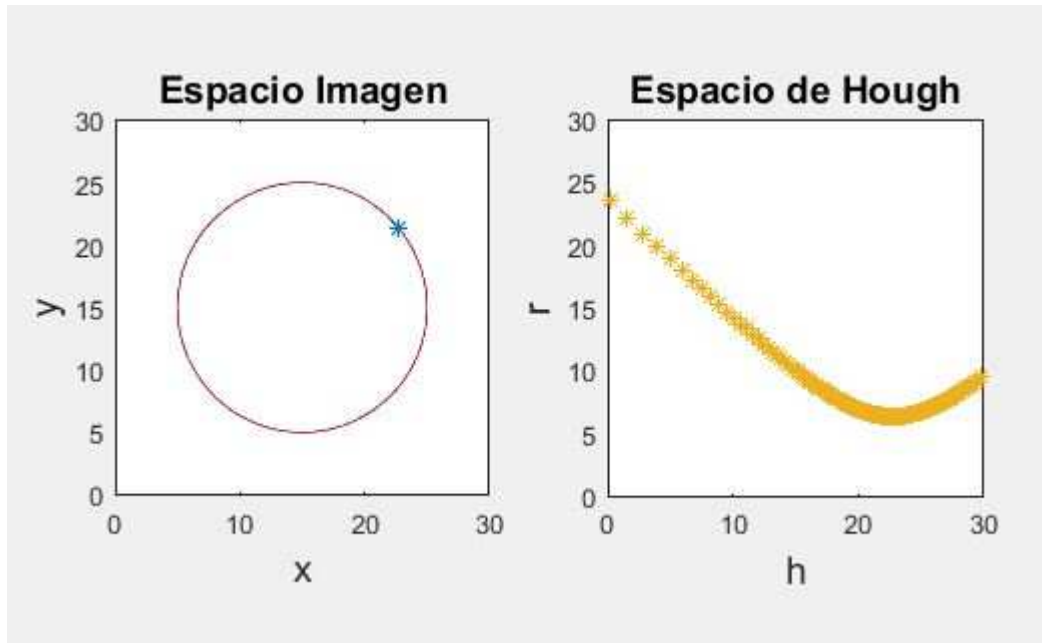


Figura 3.14: Representación de un punto del espacio imagen al plano de Hough (h, r) .

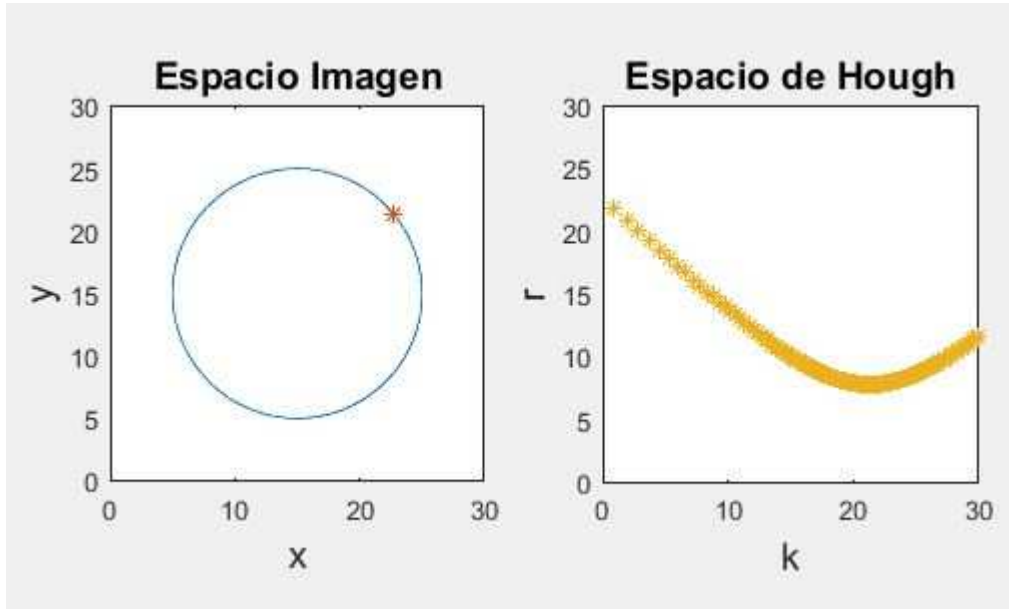


Figura 3.15: Representación de un punto del espacio imagen al plano de Hough (k, r) .

3.3.6. Métodos para la reducción de la dimensión o para la descomposición del espacio de Hough

Una solución para hacer viable el método de detección de círculos mediante la transformada de Hough, es utilizar la dirección de gradiente de los puntos candidatos. Eso permite reducir la dimensión del espacio de Hough. Recordemos de la sección 1.5 que el gradiente de la función imagen $f(x, y)$ es el vector $\nabla f = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right)$, con su magnitud dada por la norma euclidiana,

$$|\nabla f| = \sqrt{\left(\frac{\partial f}{\partial x} \right)^2 + \left(\frac{\partial f}{\partial y} \right)^2}, \text{ y su pendiente por } \tan \theta = \frac{f_y}{f_x}.$$

Para todo punto candidato que pertenece a un círculo, la dirección del gradiente en ese punto apunta hacia donde se encuentra el centro del círculo. Sólo sería necesario trazar una línea recta de longitud r desde el punto candidato en la dirección del gradiente, para encontrar el centro. El uso del gradiente disminuye de manera importante el número y tiempo de cálculos para la detección de la circunferencia. El programa MATLAB utiliza la dirección de gradiente en su método de detección de circunferencias, además, limita la búsqueda a un rango de radios, aunque aparece limitado al tener que especificar un rango de r que cumpla con $r_{\min} > 10$, $r_{\max} < 3r_{\min}$, $(r_{\max} - r_{\min}) < 100$. Para la búsqueda del radio correcto ofrece dos opciones: el método *Two Stage* y el método *Phase Coding*. Con *Two Stage*, se hace uso de un histograma de los valores del radio; mientras que la opción *Phase Coding* considera al radio que implícitamente fue utilizado para los cálculos, como el correcto.

Un método descrito en [26] y presentado a continuación, aprovecha la idea de dirección

de gradiente, con base en las derivadas de la función correspondiente a la curva del círculo, vea figura 3.16.

Sea $(x(\theta), y(\theta))$ un píxel candidato que pertenece a un círculo, cuya representación paramétrica puede ser interpretada como una función vectorial de la siguiente forma:

$$w = \begin{pmatrix} x \\ y \end{pmatrix} = w(\theta) = \begin{pmatrix} x(\theta) \\ y(\theta) \end{pmatrix} = \begin{pmatrix} h + r \cos \theta \\ k + r \sin \theta \end{pmatrix}$$

donde las coordenadas h, k corresponden al centro del círculo con radio r . Entonces

$$v(\theta) = w - \begin{pmatrix} h \\ k \end{pmatrix} = \begin{pmatrix} x(\theta) - h \\ y(\theta) - k \end{pmatrix} = \begin{pmatrix} r \cos(\theta) \\ r \sin(\theta) \end{pmatrix} .$$

Se obtienen la primera y segunda derivada de $v(\theta)$:

$$v'(\theta) = \begin{pmatrix} x'(\theta) \\ y'(\theta) \end{pmatrix} = \begin{pmatrix} -r \sin \theta \\ r \cos \theta \end{pmatrix}$$

$$v''(\theta) = \begin{pmatrix} x''(\theta) \\ y''(\theta) \end{pmatrix} = \begin{pmatrix} -r \cos \theta \\ -r \sin \theta \end{pmatrix}$$

Como conclusión, $v''(\theta) = -v(\theta)$. Tomando en cuenta que el vector $v'(\theta)$ determina a la tangente al círculo en el punto $v'(\theta)$, el ángulo entre el eje x y el vector $v(\theta)$ es dado por $\alpha = \theta + 90^\circ$. Si β denota al ángulo entre el eje x y el vector $v''(\theta)$, $v''(\theta) = -v(\theta)$ implica que $\beta = \theta + 180^\circ$, es decir, los vectores $v'(\theta)$ y $v''(\theta)$ son perpendiculares, y las pendientes son como sigue:

$$\tan \alpha = \frac{y'(\theta)}{x'(\theta)}, \quad \tan \beta = \tan(\theta + 180^\circ) = \tan \theta = \frac{y''(\theta)}{x''(\theta)} = -\frac{1}{\tan \alpha} .$$

Recordando que

$$\tan \beta = \frac{y''(\theta)}{x''(\theta)} = \frac{-r \sin \theta}{-r \cos \theta} = \frac{r \sin \theta}{r \cos \theta} = \frac{y(\theta) - k}{x(\theta) - h} ,$$

obtenemos la ecuación de una línea recta que pasa por el centro (h, k) y el punto $(x(\theta), y(\theta))$ de la circunferencia:

$$y(\theta) = \tan \beta (x(\theta) - h) + k = -\frac{x(\theta) - h}{\tan \alpha} + k .$$

Por eso, k puede ser obtenido por cualquiera de las siguientes ecuaciones, usando $\tan \theta = \tan \beta$ o $\tan \alpha$:

$$k = y(\theta) - \tan \beta (x(\theta) - h), \quad k = \frac{x(\theta) - h}{\tan \alpha} + y(\theta) .$$

Lo importante de ambas ecuaciones es que no dependen de r . En la literatura se dice que se ha *descompuesto el espacio de parámetros*: se puede usar el espacio 2D de Hough para

determinar los centros, con independencia del radio, y resolver después el problema del radio. El hecho importante para poder usar esta última ecuación, es que para cada punto candidato $(x(\theta), y(\theta))$, podemos determinar el valor de $\tan \alpha$ o $\tan \beta = -\frac{1}{\tan \alpha}$, a partir de la imagen gradiente. Ya que aún no se tiene conocimiento del valor de h , se calcula k para cada valor de $h \in [0, h_{\text{máx}}]$ en un espacio de Hough 2D con abscisa h y ordenada k . Entonces, la matriz acumulador suma un voto en las coordenadas (h, k) . Es decir, toda la línea recta de los puntos dados por cada $h \in [0, h_{\text{máx}}]$ y el k correspondiente calculado, llamada ***línea de votos***, es generada en un plano de Hough. Cada punto candidato es usado para generar una tal línea recta.

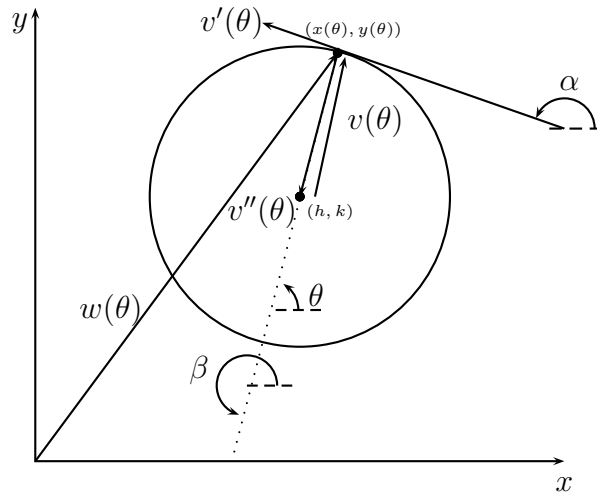


Figura 3.16: Representación gráfica del método basado en la magnitud y la dirección del gradiente

Sobre la base matemática explicada arriba, el ***método para la detección de centros de círculos basado en la magnitud y la dirección del gradiente***, consiste en los siguientes pasos:

1. Se calcula el gradiente en la imagen original en cada punto, esto permite obtener el magnitud y la dirección del gradiente que corresponde a la pendiente $\tan \beta$. Se establece un valor umbral, y todos los puntos con magnitud gradiente mayor que el valor umbral son considerados puntos candidatos; otros puntos son despreciados.
2. Se busca en la imagen los puntos candidatos, para cada punto p candidato, $p = (x, y) = (x(\theta), y(\theta))$, se efectúa el paso 3.
3. Se genera la línea de votos en el plano (h, k) de Hough, y los votos son acumulados en una matriz Ac de dimensión dos: se varía h dentro de un intervalo prefijado $[0, h_{\text{máx}}]$ o

$[h_{\min}, h_{\max}]$, y se calcula k a partir de la ecuación

$$k = y(\theta) - \tan \beta (x(\theta) - h), \quad k = \frac{x(\theta) - h}{\tan \alpha} + y(\theta),$$

donde se puede trabajar con α o con β equivalentemente, $\beta = \alpha + 90^\circ$. Si $\alpha = 0$ o $\alpha = n \cdot 180^\circ$ ($n \in \mathbb{Z}$), eso significa que $\tan \alpha = 0$ por lo cual k no está definida, este punto candidato p es ignorado y se procede con el paso 2.

4. Después de haber generado todas las líneas de votos de los puntos candidatos en el plano (h, k) de Hough y haber acumulado estos votos en la matriz Ac , se determinan los máximos locales en esta matriz. Las celdas correspondientes proporcionan las coordenadas de los puntos centros de las circunferencias más evidentes en la imagen.
5. Por último, para la detección de los radios de los círculos, se aplica el método para centro conocido expuesto en la sección 3.3.3.

Otra opción es el *método de dos puntos finales de una cuerda* [26]. Ese es muy similar al método anterior, también usa una matriz acumulador Ac de dimensión dos para encontrar los parámetros (h, k) del centro de la circunferencia, y después un histograma (de dimensión uno) en la búsqueda del radio r . Este método no usa conocimiento del punto $p = (x, y)$, tampoco de α o de β . La idea alternativa para encontrar los parámetros (h, k) consiste en el uso de una línea de votos L la cual es perpendicular a la cuerda creada por dos puntos $p_1 = (x_1, y_1)$ y $p_2 = (x_2, y_2)$ pertenecientes a la circunferencia a encontrar. Se supone que la distancia entre p_1 y p_2 es estrictamente menor al diámetro de esta circunferencia.

Tales dos puntos p_1 y p_2 de la misma circunferencia pueden ser encontrados entre los puntos candidatos de la imagen, por ejemplo, seleccionando p_1 arbitrariamente y luego buscando a p_2 dentro de una vecindad apropiada de p_1 . Observando la cuerda $\overline{p_1 p_2}$, vea figura 3.17, se considera la línea recta L que pasa por el centro y el punto p desconocido, es perpendicular a la cuerda y la bisecta. Eso significa que L pasa también por el punto medio $p_m = (x_m, y_m)$ de la cuerda $\overline{p_1 p_2}$, dado por

$$x_m = \frac{x_1 + x_2}{2}, \quad y_m = \frac{y_1 + y_2}{2}.$$

El conocimiento de los puntos p_1, p_2 es tan útil como si se conociera p , dado que la pendiente de la cuerda $\overline{p_2 p_1}$ coincide con la pendiente de la tangente en p , $\tan \alpha$, y la línea L con pendiente $\tan \beta$ es perpendicular a la cuerda. Como antes, se tiene $\alpha = \beta + 90^\circ$. Observando al triángulo de color gris en la figura 3.17, se nota que

$$\tan(180^\circ - \alpha) = \frac{y_2 - y_1}{x_2 - x_1} = -\tan \alpha = -\frac{1}{\tan \beta},$$

y tomando en cuenta que β es el ángulo de la pendiente de L , se obtiene que

$$y_m - k = \tan \beta \cdot (x_m - h).$$

Por eso, la línea recta L , al ser construida en el espacio 2D de Hough, no solamente es dada por $k = y_m - \tan \beta \cdot (x_m - h)$ sino también y alternativamente, por

$$k = y_m + \frac{(x_2 - x_1)(x_m - h)}{(y_2 - y_1)} .$$

Esta última ecuación permite la construcción de L mediante la variación de h dentro de un intervalo $[0, h_{\text{máx}}]$ y calculando los valores de k , utilizando únicamente p_1 y p_2 (y p_m el cual es obtenido a partir de p_1 y p_2), sin necesidad de conocer el ángulo β . Es importante mencionar que la ecuación anterior es completamente independiente del radio r . Nuevamente, se ha logrado una *descomposición del espacio de Hough* lo cual permite calcular primero el centro del círculo, y dejar la determinación del radio para un paso después y aparte.

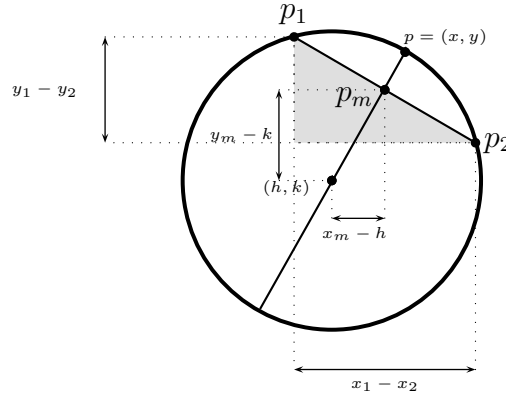


Figura 3.17: Representación gráfica del método de dos puntos finales de una cuerda.

Aprovechando lo expuesto anteriormente, y conforme a lo propuesto en [26], el **método de dos puntos finales de una cuerda**, consiste en los siguientes pasos:

1. Primero, la imagen es escaneada para encontrar un punto candidato $p_1 = (x_1, y_1)$. Luego, un segundo punto $p_2 = (x_2, y_2)$ es buscado en una vecindad cuadrada de 25x25 píxeles centrada en p_1 , pero fuera de la vecindad de 21x21 píxeles. Es decir, p_2 debe satisfacer que $10 < |x_2 - x_1| \leq 12$ y $10 < |y_2 - y_1| \leq 12$. En caso de no hallar un segundo punto, se elige otro punto p_1 y se repite el proceso anterior, hasta encontrar una pareja de puntos candidatos p_1, p_2 .
2. Se obtiene el punto medio $p_m = (x_m, y_m)$ del segmento de línea recta $\overline{p_1 p_2}$, y la pendiente de la línea de votos L :

$$x_m = \frac{x_1 + x_2}{2} , \quad y_m = \frac{y_1 + y_2}{2} , \quad a = \frac{x_2 - x_1}{y_2 - y_1} .$$

3. En el plano de Hough, se genera la línea de votos dada por la siguiente ecuación, variando h dentro de un intervalo $[0, h_{\text{máx}}]$ o $[h_{\text{mín}}, h_{\text{máx}}]$, y calculando el valor correspondiente de k para cada h :

$$k = a \cdot (x_m - h) + y_m .$$

Cada par de valores h, k obtenido proporciona un voto en una matriz acumulador Ac de dimension dos, correspondiente al plano de Hough donde la línea L es creada.

4. Se determinan los máximos locales en la matriz acumulador Ac , que corresponden a coordenadas (h, k) de posibles puntos centros de circunferencias presentes en la imagen.
5. Aprovechando cada punto centro ya determinado, se aplica el método de detección de círculos con centro conocido expuesto en la sección 3.3.3, para la detección del radio de esta circunferencia.

Una variante del método de los dos puntos de una cuerda, se presenta en [19]. Esta variante también hace uso de un par de puntos p_1, p_2 formando una cuerda de una circunferencia, y de la línea L que es perpendicular y bisecta a esta cuerda. Sin embargo, la diferencia es que ahora, los datos de entrada de la imagen deben representar no solamente puntos de borde o puntos candidatos, sino curvas digitales o contornos. Los puntos candidatos deben estar 8-conectados, para formar uno o varios círculos digitales. La búsqueda de los dos puntos p_1, p_2 finales de una cuerda, es realizada en cada componente 8-conexa del conjunto de todos los puntos candidatos. Se selecciona al punto p_1 , luego p_2 es determinado en una vecindad de cierta cercanía de p_1 , pero dentro de la curva digital dada y aprovechando su sentido de trazado. Recordemos que una curva digital es una secuencia ordenada de puntos (c_1, c_2, \dots, c_k) . Por ejemplo, si p_1 coincide con c_i , p_2 puede ser seleccionado como el punto c_{i+k} o el punto c_{i-k} .

Otro método para la detección de círculos es el mencionado en [39] y denominado por los autores como “*Fast Finding and Fitting (FFF)*”. Éste es muy similar al método descrito en [40] el cual originalmente es propuesto más generalmente para detectar elipses y será explicado en esta tesis en la siguiente sección. Adaptando estos métodos a la detección de círculos, la idea básica sería buscar dos parejas de puntos pertenecientes a una misma circunferencia, donde en cada pareja, ambos puntos tengan direcciones de gradiente opuestos o iguales (detectores de bordes paralelos). La motivación para eso es que estos cuatro puntos sean los puntos finales de dos ejes de simetría, perpendiculares entre sí, del círculo. Más especialmente, los autores de [39] consideran solamente a los ejes de simetría vertical y horizontal del círculo. Precisamente estos ejes de simetría son usados como líneas de votos. Al final, las celdas de la matriz acumulador 2D que presentan una mayor acumulación de votos, corresponden a posibles centros de círculos cuyo radio simplemente es la distancia euclidiana del centro a cualquier de los cuatro puntos sobre la circunferencia.

3.4. Detección de elipses

3.4.1. Aplicación de la transformada de Hough para la detección de elipses

Para la detección de elipses mediante la transformada de Hough se puede utilizar la representación paramétrica introducida en la sección 2.2.3, es decir,

$$x = a \cos \theta + h, \quad y = b \sin \theta + k,$$

donde $\theta \in [0, 2\pi)$ es un parámetro libre que permite generar la curva, y a, b, h, k son los parámetros de una elipse, siendo a el semi-eje mayor, b el semi-eje menor y (h, k) las coordenadas del punto centro. Podemos escribir equivalentemente

$$h = x - a \cos \theta, \quad k = y - b \sin \theta,$$

donde se expresa mejor la idea de qué queremos hallar. La idea básica de la transformación de Hough para la detección de elipses es análoga al caso de los círculos. Recordemos que la representación de arriba con cuatro parámetros supone que el eje mayor de la elipse está alineado con el eje x del sistema de coordenadas. Para este caso, el espacio de Hough tiene dimensión cuatro y cada elemento de la matriz acumulador tiene la forma $Ac(h, k, a, b)$. Sin embargo, una elipse general puede estar rotada, es decir, tener una orientación de su eje mayor con respecto al eje x . Por eso, en [26] es utilizada la siguiente representación general:

$$h = x - a_x \cos \theta - b_x \sin \theta \tag{3.1}$$

$$k = y - a_y \cos \theta - b_y \sin \theta, \tag{3.2}$$

donde, h, k son las coordenadas del centro de la elipse; (a_x, a_y, b_x, b_y) representan los semiejes de la elipse con respecto a los ejes coordenados x, y , es decir, son las componentes de los semiejes, ver figura 3.18. La orientación α del eje mayor de la elipse con respecto al eje x , y sus semiejes a (mayor) y b (menor) pueden entonces ser obtenidos como sigue:

$$\tan \alpha = \frac{a_y}{a_x}, \quad a = \sqrt{a_x^2 + a_y^2}, \quad b = \sqrt{b_x^2 + b_y^2}.$$

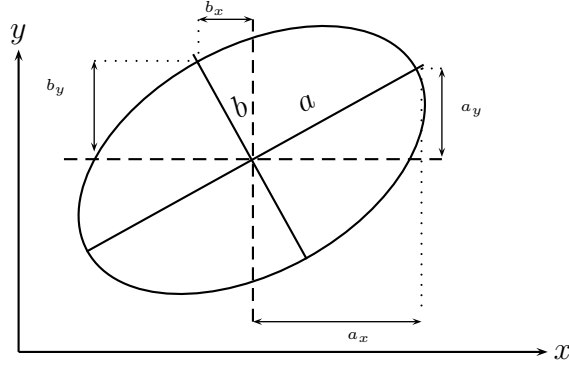


Figura 3.18: Componentes de los semiejes a, b de una elipse.

El espacio de Hough para representar elipses es de dimensión cinco: dos coordenadas para el centro, dos para sus (semi-)ejes y una para la orientación. Igualmente, cada elemento de la matriz acumulador Ac tiene cinco índices. La memorización y el tratamiento de un espacio de Hough con tan alta dimensión, en particular, el registro de los votos y luego la determinación de los máximos locales en Ac , son computacionalmente muy costosos. Por eso, el método de Hough no es usado en la práctica en toda su generalidad para detectar elipses. Sin embargo, aprovechando conocimiento sobre la especificación o la acotación severa de los parámetros o sobre los puntos candidatos, se han diseñado métodos viables para detectar elipses con ciertas restricciones. También, en caso que los puntos candidatos son puntos de borde detectados en la imagen mediante algún filtro de gradiente, conocimiento de la dirección de la primera o segunda derivada en cada de estos puntos puede ser utilizado para bajar la dimensión del espacio de Hough, de manera parecida a como fue explicado para círculos en las secciones anteriores.

El procedimiento general para la detección de elipses por medio de la transformada de Hough en una imagen digital, si consideramos que el eje mayor a es paralelo al eje x , (sin rotación) consiste en los siguientes pasos:

Paso 1

Se subdivide el espacio de cuatro parámetros (h, k, a, b) en celdas para la generación de la matriz acumulador Ac . Suponiendo que $h \in [0, h_{\text{máx}}]$, $k \in [0, k_{\text{máx}}]$, $a \in [0, a_{\text{máx}}]$, $b \in [0, b_{\text{máx}}]$, estos intervalos son divididos en porciones, por ejemplo, para números l, m, n, s convenientes,

como sigue:

$$\begin{aligned}
[0, h_{\text{máx}}] &= \left[0, \frac{1}{l}h_{\text{máx}}\right] \cup \left[\frac{1}{l}h_{\text{máx}}, \frac{2}{l}h_{\text{máx}}\right] \cup \dots \cup \left[\frac{l-1}{l}h_{\text{máx}}, h_{\text{máx}}\right], \\
[0, k_{\text{máx}}] &= \left[0, \frac{1}{m}k_{\text{máx}}\right] \cup \left[\frac{1}{m}k_{\text{máx}}, \frac{2}{m}k_{\text{máx}}\right] \cup \dots \cup \left[\frac{m-1}{m}k_{\text{máx}}, k_{\text{máx}}\right], \\
[0, a_{\text{máx}}] &= \left[0, \frac{1}{n}a_{\text{máx}}\right] \cup \left[\frac{1}{n}a_{\text{máx}}, \frac{2}{n}a_{\text{máx}}\right] \cup \dots \cup \left[\frac{n-1}{n}a_{\text{máx}}, a_{\text{máx}}\right], \\
[0, b_{\text{máx}}] &= \left[0, \frac{1}{s}b_{\text{máx}}\right] \cup \left[\frac{1}{s}b_{\text{máx}}, \frac{2}{s}b_{\text{máx}}\right] \cup \dots \cup \left[\frac{s-1}{s}b_{\text{máx}}, b_{\text{máx}}\right].
\end{aligned}$$

Eso genera una subdivisión del espacio de parámetros en paralelepípedos:

$$[0, h_{\text{máx}}] \times [0, k_{\text{máx}}] \times [0, a_{\text{máx}}] \times [0, b_{\text{máx}}] = \bigcup_{\substack{f=1,\dots,l \\ g=1,\dots,m \\ i=1,\dots,n \\ j=1,\dots,s}} (f, g, i, j),$$

donde (f, g, i, j) denota la siguiente celda de dimensión cuatro:

$$\left[\frac{f-1}{l}h_{\text{máx}}, \frac{f}{l}h_{\text{máx}}\right] \times \left[\frac{g-1}{m}k_{\text{máx}}, \frac{g}{m}k_{\text{máx}}\right] \times \left[\frac{i-1}{n}a_{\text{máx}}, \frac{i}{n}a_{\text{máx}}\right] \times \left[\frac{j-1}{s}b_{\text{máx}}, \frac{j}{s}b_{\text{máx}}\right]$$

Estas mismas coordenadas f, g, i, j son usadas como índices para la matriz acumulador Ac con valores iniciales $Ac(f, g, i, j) = 0$ para todos los f, g, i, j .

Paso 2

Para cada punto (x, y) en la imagen donde es candidato para pertenecer a alguna elipse, en el espacio de Hough se varían los valores de a, b y se calculan los parámetros h, k . Variando a, b significa que a y b son fijados dentro de sus respectivos intervalos $[0, a_{\text{máx}}]$, $[0, b_{\text{máx}}]$ o $[a_{\text{mín}}, a_{\text{máx}}]$, $[b_{\text{mín}}, b_{\text{máx}}]$. Luego, cada vez que a, b son fijados, aplicando la representación paramétrica y asumiendo valores cuantizados para $\theta \in [\theta, 2\pi)$, los parámetros libres h, k son calculados, obteniéndose parámetros h, k, a, b los cuales, cuando caen en una celda (f, g, i, j) , esta celda suma un voto, es decir, $Ac(f, g, i, j)$ es aumentado por 1.

Paso 3

Se determinan los máximos locales en la matriz Ac , correspondientes a potenciales parámetros de las elipses presentes en la imagen.

Cabe notar que fue expuesto aquí solamente la idea básica del método, la cual en la práctica sólo es realizable si los intervalos para los parámetros son muy pequeños o si t, l, m, n, s pueden ser pequeños, o si el número de píxeles candidatos es muy pequeño. Bajo condiciones generales, el uso de la transformada de Hough con cuatro o cinco parámetros no es viable. Para usos prácticos, el proceso de detección de elipses tiene que ser separado en varias partes y el espacio de Hough tiene que ser descompuesto, para poder usar una matriz acumulador de dimensión máxima dos en cada parte del método.

3.4.2. Método geométrico con dirección de gradiente

Un método para la detección de elipses que fue propuesto en [40], se basa en la transformada de Hough y en una descomposición del espacio de parámetros de tal manera que la acumulación de evidencias de elipses en la imagen se realiza en una matriz de dimension dos. Las ideas básicas de este método son las siguientes:

- Una elipse E es completamente determinada por cualquier rectángulo inscrito en E que tiene el mismo punto centro y la misma orientación que E , es decir, uno de sus lados es paralelo al eje mayor y el otro al eje menor de la elipse.
- Los vértices de un tal rectángulo pueden ser encontrados en la imagen, usando la dirección del gradiente en los puntos candidatos (puntos de borde) de la imagen.
- La búsqueda de cuartetos de puntos que corresponden a los vértices de un rectángulo inscrito en alguna elipse presente en la imagen, y con el mismo centro y orientación como ésta, tendrá muchos resultados erróneos. Por eso, es conveniente usar cada cuarteto encontrado para calcular los datos de la posible elipse (tal vez errónea), y luego someter estos datos al método de Hough de acumulación de evidencia. Los autores de [40] calculan los cuatro vértices $v_i = (x_{v_i}, y_{v_i})$, $i = 1, 2, 3, 4$, de la elipse a partir de cada cuarteto de puntos (siendo vértices de rectángulo), y registran todas las coordenadas x_{v_i}, y_{v_i} en una matriz acumulador correspondiente a un espacio de parámetros de dimensión dos. Cuando el proceso de acumulación ha terminado, una elipse presente en la imagen corresponde a un cuarteto de máximos locales en esta matriz.

A continuación, será explicada cada una de estas bases del método. Sea E una elipse con punto centro $p_m = (h, k)$ y su eje mayor alineado con un sistema cartesiano de coordenadas. Además, consideremos los vértices $p_1 = (x_1, y_1)$, $p_2 = (x_2, y_2)$, $p_3 = (x_3, y_3)$, $p_4 = (x_4, y_4)$ acomodados como en la figura 3.19, de un rectángulo inscrito en E con el mismo centro p_m , alineado con los ejes mayor y menor de E . Entonces sus diagonales $\overline{p_1 p_3}$ y $\overline{p_2 p_4}$ se intersecan en el punto p_m , es decir,

$$h = \frac{x_1 + x_3}{2} = \frac{x_2 + x_4}{2}, \quad k = \frac{y_1 + y_3}{2} = \frac{y_2 + y_4}{2}.$$

Obviamente, la distancia $d(p_1, p_3)$ coincide con $d(p_2, p_4)$, y vale que

$$d(p_m, p_1) = d(p_m, p_2) = d(p_m, p_3) = d(p_m, p_4) = \frac{d(p_1, p_3)}{2} = \frac{d(p_2, p_4)}{2} = r,$$

donde r es el radio del círculo centrado en p_m que circunscribe al rectángulo. Es importante notar que estas últimas ecuaciones permiten determinar al punto centro (h, k) de la elipse, y en caso de que la elipse fuera un círculo, al radio r de este, a partir del cuarteto de puntos (p_1, p_2, p_3, p_4) .

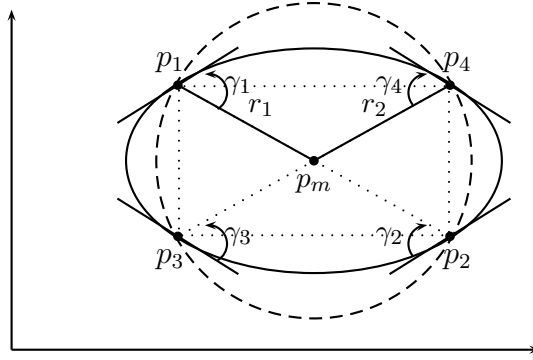


Figura 3.19: Una elipse y los elementos geométricos para el uso del método geométrico con dirección de gradiente.

La propiedad más importante para este método de detección de elipses es que las tangentes a la curva de la elipse en los puntos p_1 y p_2 son paralelas; igualmente, las tangentes en p_3 y p_4 son paralelas. De hecho, vea figura 3.20, las tangentes en p_1 y p_2 son paralelas entre sí y a la línea recta que pasa por p_3 y p_4 . Similarmente, las tangentes en p_3 y p_4 son paralelas entre sí y también a la línea recta que pasa por p_1 y p_2 .

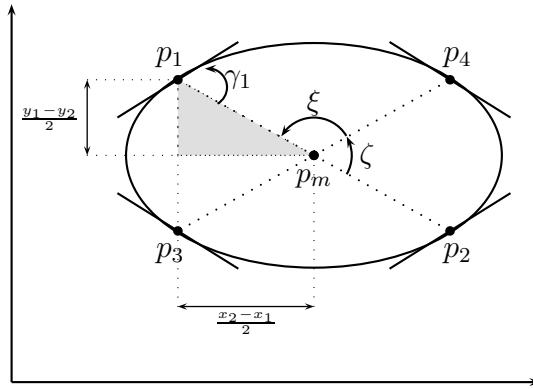


Figura 3.20: Ángulos importantes en la elipse.

Definimos, para cada punto p_i , $i = 1, 2, 3, 4$, γ_i como el ángulo menor de los ángulos formados por la tangente en p_i y el segmento de línea $\overline{p_m p_i}$, vea figuras 3.19 y 3.20. Tenemos $\gamma_i \in (0, \pi/2]$, $i = 1, 2, 3, 4$, y cuando $\gamma_i = \pi/2$, entonces la elipse es un círculo. Dadas las

simetrías del rectángulo y de la elipse,

$$\gamma_1 = \gamma_2 = \gamma_3 = \gamma_4 ,$$

lo cual justifica denotar cualquiera de estos ángulos por γ . Este ángulo puede ser determinado a partir de los puntos p_i . Observando al triángulo marcado en color gris en la figura 3.20, se tiene que

$$\tan\left(\frac{\zeta}{2}\right) = \frac{y_1 - y_2}{x_2 - x_1} .$$

Sin embargo, ζ y γ_1 son ángulos formados por líneas paralelas y la misma línea $\overline{p_1 p_2}$, por lo cual, son iguales. Tenemos entonces para todo $i = 1, 2, 3, 4$, vea figura 3.20, que

$$\gamma = \gamma_i = \zeta = 180^\circ - \xi, \quad \tan\left(\frac{\gamma}{2}\right) = \tan\left(\frac{\gamma_i}{2}\right) = \frac{y_1 - y_2}{x_2 - x_1} .$$

La obtención del ángulo γ permite determinar si trata de una elipse o de un círculo (sólo si $\gamma = \frac{\pi}{2}$).

Ahora vamos a determinar todos los parámetros de la elipse, es decir, las coordenadas del punto centro, las longitudes de los ejes mayor y menor, y la inclinación de la elipse, a partir del cuarteto de puntos (p_1, p_2, p_3, p_4) .

Las propiedades mencionadas arriba del rectángulo formado por p_1, p_2, p_3, p_4 , se mantienen cuando la elipse E se encuentra inclinada, es decir, rotada por un ángulo θ , vea figura 3.21. Consideramos la elipse rotada dentro de un sistema de coordenadas (x, y) centrado en $p_m = (h, k)$, pero también puede ser vista en un sistema de coordenadas x', y' el cual se obtiene por rotación por el ángulo θ . Es decir, el sistema (x', y') se encuentra alineado con los ejes de la elipse E .

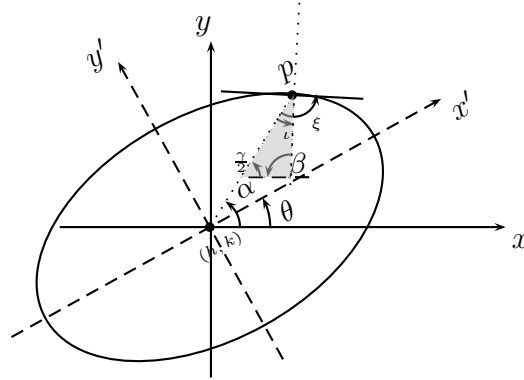


Figura 3.21: Elipse rotada por un ángulo θ .

Recordemos que el rectángulo formado por (p_1, p_2, p_3, p_4) tiene la misma inclinación que la elipse, la cual es determinada por su eje mayor. Los segmentos de línea recta $\overline{p_1 p_4}$, $\overline{p_3 p_2}$ ambos son paralelos al eje mayor de E . Por eso, para calcular la inclinación de E , tenemos, por ejemplo, las siguientes posibilidades equivalentes:

$$\theta = \tan^{-1} \left(\frac{y_4 - y_1}{x_4 - x_1} \right) = \tan^{-1} \left(\frac{y_3 - y_2}{x_3 - x_2} \right)$$

Para determinar las longitudes de los ejes mayor y menor, o equivalentemente, los parámetros a, b de la elipse, recordemos su representación algebraica: en coordenadas x', y' , cada punto $p = (x', y') \in E$, vea figura 3.21, cumple que

$$\frac{x'^2}{a^2} + \frac{y'^2}{b^2} = 1 .$$

Representamos a p por coordenadas polares, denotando por r la distancia entre p y p_m , por α al ángulo entre el eje x y el segmento de línea $\overline{p_m p}$, es decir, $\alpha = \tan^{-1} \left(\frac{y'-k}{x'-h} \right)$. Tomando en cuenta la inclinación de E por el ángulo θ ,

$$x' = r \cos(\alpha - \theta) , \quad y' = r \sin(\alpha - \theta) .$$

Entonces obtenemos la siguiente representación, llamada forma polar de la elipse en [40]:

$$\frac{r^2 \cos^2(\alpha - \theta)}{a^2} + \frac{r^2 \sin^2(\alpha - \theta)}{b^2} = 1 .$$

En coordenadas x, y , la representación de $p = (x, y)$ es dada por

$$x = r \cos \alpha , \quad y = r \sin \alpha .$$

Entonces podemos expresar a x', y' en dependencia de x, y :

$$\begin{aligned} x' &= r \cos(\alpha - \theta) = (r \cos \alpha) \cos \theta + (r \sin \alpha) \sin \theta = x \cos \theta + y \sin \theta \\ y' &= r \sin(\alpha - \theta) = (r \sin \alpha) \cos \theta - (r \cos \alpha) \sin \theta = -x \sin \theta + y \cos \theta \end{aligned}$$

Nos interesa obtener el gradiente, es decir, la pendiente de la línea recta perpendicular a la tangente, o la de la tangente misma, en el punto p . Considerando y' como función de x' , aplicamos derivación implícita $\frac{d}{dx}$ a la ecuación algebraica:

$$\frac{x'^2}{a^2} + \frac{y'^2}{b^2} = 1 \implies \frac{1}{a^2} \frac{dx'^2}{dx} + \frac{1}{b^2} \frac{dy'^2}{dx} = 0 \implies \frac{2x'}{a^2} + \frac{1}{b^2} \frac{dy'^2}{dx} = 0 .$$

Por la regla de la cadena se obtiene

$$\frac{2x'}{a^2} + \frac{2y'}{b^2} \frac{dy'}{dx} = 0 ,$$

lo cual se resuelve para $\frac{dy'}{dx'}$ que es la pendiente de la tangente:

$$\frac{dy'}{dx'} = -\frac{b^2 x'}{a^2 y'}.$$

Expresando x', y' en términos de las coordenadas x, y como se hizo arriba, se llega a lo siguiente, ya incluyendo la rotación de la elipse por el ángulo θ :

$$\tan(\beta - \theta) = -\left(\frac{b}{a}\right)^2 \frac{x'}{y'} = -\left(\frac{b}{a}\right)^2 \frac{x \cos \theta + y \sin \theta}{-x \sin \theta + y \cos \theta}.$$

Considerando la relación del punto $p = (x_p, y_p)$ con respecto al centro de la elipse $p_m = (h, k)$, se tiene que $x = x_p - h$ y $y = y_p - k$, y por tanto

$$\tan(\beta - \theta) = -\left(\frac{b}{a}\right)^2 \left[\frac{(x_p - h) \cos \theta + (y_p - k) \sin \theta}{-(x_p - h) \sin \theta + (y_p - k) \cos \theta} \right]$$

donde β es el ángulo de la dirección del gradiente en el punto p , β puede ser obtenido considerando las figura 3.21. En el triángulo de color gris se observa que

$$\xi = 90^\circ + \iota \implies \iota = \xi - 90^\circ, \quad \xi = 180^\circ - \gamma \implies \iota = 90^\circ - \gamma$$

por lo que

$$\beta + \frac{\gamma}{2} + \iota = 180^\circ \implies \beta = 180^\circ - \frac{\gamma}{2} - \iota = 90^\circ + \frac{\gamma}{2}.$$

Por último, se necesita resolver para a y b el sistema de dos ecuaciones, la primera formada para $\tan(\beta - \theta)$, y la segunda ecuación dada por la forma polar de la elipse, utilizando cualquier método de resolución de sistemas de ecuaciones.

Como resultado, se obtienen todos los parámetros (h, k, a, b, θ) de la elipse, considerando únicamente la cuarteta de puntos p_1, p_2, p_3, p_4 . Este cuarteto había sido seleccionado de los puntos candidatos de la imagen y, supuestamente, son los vértices de un rectángulo inscrito en una elipse. Sin embargo, en la práctica, muchos de tales cuartetos resultarán erróneos, en el sentido de que, por ejemplo, no pertenecen a la misma elipse, o que el rectángulo no tiene la misma inclinación que la elipse buscada. Eso implica que también los datos (h, k, a, b, θ) calculados solamente son preliminares puesto que, en muchos casos, son erróneos.

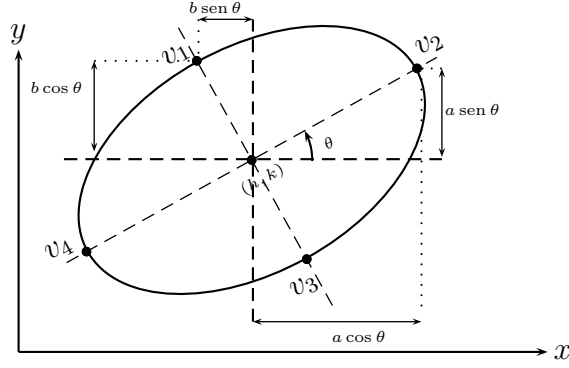


Figura 3.22: Relación entre los vértices v_1, v_2, v_3, v_4 de la elipse y sus parámetros θ, h, k, a, b .

Por eso, los autores de [40] al final aplican un proceso de acumulación de evidencias, usando una matriz acumulador de dimensión dos, con parámetros correspondientes a las coordenadas x_{vj}, y_{vj} de los cuatro vértices de la elipse calculados a partir de cada cuarteto (p_1, p_2, p_3, p_4) . Eso se basa en las siguientes relaciones entre los parámetros de la elipse y sus vértices v_1, v_2, v_3, v_4 , vea la figura 3.22:

$$\begin{aligned} v_1 &= (x_{v1}, y_{v1}) = (h - b \sen \theta, k + b \cos \theta) \\ v_2 &= (x_{v2}, y_{v2}) = (h + a \cos \theta, k + a \sen \theta) \\ v_3 &= (x_{v3}, y_{v3}) = (h + b \sen \theta, k - b \cos \theta) \\ v_4 &= (x_{v4}, y_{v4}) = (h - a \cos \theta, k - a \sen \theta). \end{aligned}$$

El proceso de acumulación de evidencias consiste en registrar cada par de coordenadas x_{vj}, y_{vj} de cualquiera de los cuatro vértices v_j calculados a partir de cada cuarteto (p_1, p_2, p_3, p_4) seleccionado en la imagen. Todos los registros se llevan a cabo en la misma única matriz acumulador Ac cuyos elementos corresponden a celdas que cuantifican un plano de parámetros. Si los valores x_{vj}, y_{vj} caen en cierta celda, entonces el correspondiente elemento de Ac suma un voto.

Aunque muchos datos erróneos son registrados en la matriz Ac , los datos correspondientes a elipses verdaderamente presentes en la imagen, deben destacar por reflejar mucha evidencia, es decir, por haber acumulado valores máximos en Ac . Posteriormente al proceso de votación, cada elipse presente en la imagen debe reflejarse en un conjunto de cuatro máximos locales en la matriz Ac , correspondientes a los cuatro vértices v_1, v_2, v_3, v_4 de esta elipse. Una vez teniendo estos vértices destacados, denotémoslos nuevamente por v_1, v_2, v_3, v_4 , resultantes del proceso de acumulación y votación, se procede a calcular los parámetros correctos (h, k, a, b, θ) de la elipse detectada en la imagen:

$$h = \frac{x_{v1} + x_{v2} + x_{v3} + x_{v4}}{4}, \quad k = \frac{y_{v1} + y_{v2} + y_{v3} + y_{v4}}{4},$$

$$a = \frac{\sqrt{(x_{v2} - x_{v4})^2 + (y_{v2} - y_{v4})^2}}{2}, \quad b = \frac{\sqrt{(x_{v1} - x_{v3})^2 + (y_{v1} - y_{v3})^2}}{2},$$

$$\theta = \tan^{-1} \left(\frac{y_{v2} - y_{v4}}{x_{v2} - x_{v4}} \right).$$

Notemos que la determinación de los parámetros correctos de la elipse es realizada en varios pasos, iniciando con la selección del cuarteto de puntos (p_1, p_2, p_3, p_4) , calculando primero parámetros preliminares y luego vértices que son sometidos a un proceso de acumulación y votación en un espacio de parámetros de dimensión dos. Al final, ciertos datos ganadores son usados para calcular los verdaderos parámetros de la(s) elipse(s). Eso muestra que el espacio de parámetros que originalmente tiene dimensión cinco, se ha descompuesto para poder desarrollar este método.

Justificado por las bases matemáticas explicadas arriba, el **método geométrico con dirección de gradiente para detectar elipses**, propuesto en [40], consiste en los siguientes pasos. Como dato de entrada se tiene una imagen en la cual, cada punto candidato tiene un magnitud de gradiente que pasa cierto umbral prefijado, y para el cual, una estimación de la dirección del gradiente ya fue calculada.

Paso 1: Se crea una matriz acumulador Ac de dimensión dos correspondiente a parámetros x_{vi}, y_{vi} cuyos valores iniciales todos son cero. Ac corresponde a un plano de parámetros $M \times N$, siendo M el intervalo de los valores posibles o esperados para x_{vi} , y N el intervalo de los valores y_{vi} . $M \times N$ es descompuesto en celdas rectangulares de la forma $m \times n$ donde m es un subintervalo de M y n es un subintervalo de N ; cada celda corresponde a un elemento de Ac . En el paso 7, vea abajo, cada vez que un par (x_{vi}, y_{vi}) calculado cae en cierta celda del plano de parámetros, el elemento correspondiente de Ac acumula un voto, es decir, es aumentado por uno.

Paso 2: Se buscan en toda la imagen dos puntos candidatos $p_1 = (x_1, y_1)$ y $p_2 = (x_1, y_2)$, con direcciones de gradiente opuestas o iguales, es decir, el gradiente en ambos puntos detecta bordes paralelos, vea sección 1.5. Se supone que p_1, p_2 pertenecen a la misma elipse, entonces las tangentes en estos dos puntos son paralelas. Se supone su punto medio $p_m = (h, k)$ como el centro de esta elipse:

$$h = \frac{x_2 + x_1}{2}, \quad k = \frac{y_2 + y_1}{2}.$$

Se calcula la distancia r_1 de (h, k) a cualquiera de los puntos p_1, p_2 , por ejemplo,

$$r_1 = \sqrt{(y_1 - k)^2 + (x_1 - h)^2}.$$

Igualmente, se busca otro par de puntos candidatos $p_3 = (x_3, y_3), p_4 = (x_4, y_4)$, con direcciones de gradiente opuestas o iguales, y se calculan el punto medio (h', k') de $\overline{p_3 p_4}$ y la distancia r_2 entre (h, k) y p_3 o p_4 :

$$h' = \frac{x_4 + x_3}{2}, \quad k' = \frac{y_4 + y_3}{2}, \quad r_2 = \sqrt{(y_3 - k)^2 + (x_3 - h)^2}.$$

Si $h' \neq h$ o $k' \neq k$ o $r_2 \neq r_1$, los puntos p_3, p_4 son despreciados y sigue la búsqueda hasta obtener p_3, p_4 tales que $h' = h$, $k' = k$, $r_2 = r_1$. Si no se pueden encontrar p_3, p_4 cumpliendo estas condiciones, también p_1, p_2 son despreciados, y se reinicia el paso 2 en la búsqueda por mejores candidatos p_1, p_2 .

El resultado del paso 2 es una cuarteta (p_1, p_2, p_3, p_4) de puntos que son interpretados como los vértices de un rectángulo inscrito en una elipse, con el mismo centro y alineado con ésta. Además, ya se obtuvieron las coordenadas (h, k) del centro de la elipse determinada por (p_1, p_2, p_3, p_4) .

Paso 3: Se calcula el ángulo γ dado por

$$\tan\left(\frac{\gamma}{2}\right) = \frac{y_1 - y_2}{x_2 - x_1}.$$

Si $\gamma = \pi/2$, se tiene el caso especial donde la elipse determinada por la cuarteta (p_1, p_2, p_3, p_4) es un círculo. Entonces el método sigue con el paso 5, vea abajo.

Si $\gamma \neq \pi/2$, se sigue con el paso 4.

Paso 4: El ángulo de inclinación del eje mayor de la elipse con respecto al eje x es calculada por

$$\theta = \tan^{-1}\left(\frac{y_4 - y_1}{x_4 - x_1}\right).$$

Paso 5: Los parámetros a, b de la elipse son determinados, a partir de un sistema de dos ecuaciones: la ecuación de la dirección del gradiente

$$\tan(\beta - \theta) = -\left(\frac{b}{a}\right)^2 \left[\frac{(x_1 - h) \cos \theta + (y_1 - k) \sin \theta}{-(x_1 - h) \sin \theta + (y_1 - k) \cos \theta} \right]$$

y la presentación polar de la elipse, donde $\alpha = \tan^{-1}\left(\frac{y_1 - k}{x_1 - h}\right)$,

$$\frac{r^2 \cos^2(\alpha - \theta)}{a^2} + \frac{r^2 \sin^2(\alpha - \theta)}{b^2} = 1.$$

Como resultado se obtuvieron todos los parámetros (h, k, a, b, θ) de la elipse determinada por (p_1, p_2, p_3, p_4) .

En el caso que $\gamma = \pi/2$, tenemos un círculo, donde obviamente $\theta = 0$, y se obtiene para la magnitud de los semiejes que $a = b$, por tanto, el radio del círculo resulta ser $r = a = b$.

Paso 6: A partir de la cuarteta (p_1, p_2, p_3, p_4) , los parámetros x_{vi}, y_{vi} son calculados. Siguiendo el caso de la elipse, cuando $\gamma \neq \pi/2$, se calculan:

$$\begin{aligned} v_1 &= (x_{v1}, y_{v1}) = (h - b \sin \theta, k + b \cos \theta) \\ v_2 &= (x_{v2}, y_{v2}) = (h + a \cos \theta, k + a \sin \theta) \\ v_3 &= (x_{v3}, y_{v3}) = (h + b \sin \theta, k - b \cos \theta) \\ v_4 &= (x_{v4}, y_{v4}) = (h - a \cos \theta, k - a \sin \theta). \end{aligned}$$

En el caso especial del círculo, cuando $\gamma = \pi/2$, los vértices v_1, v_2, v_3, v_4 son calculados más fácilmente como sigue:

$$\begin{aligned}(x_{v1}, y_{v1}) &= (h, k + r) , & (x_{v2}, y_{v2}) &= (h + r, k) , \\(x_{v3}, y_{v3}) &= (h, k - r) , & (x_{v4}, y_{v4}) &= (h - r, k) .\end{aligned}$$

Cada pareja (x_{vi}, y_{vi}) calculada, $i = 1, 2, 3, 4$, es registrada en la misma matriz acumulador Ac : después de determinar en cuál celda del plano de parámetros encaja la pareja (x_{vi}, y_{vi}) , el elemento correspondiente de Ac suma un voto.

Los **pasos 2-6 son repetidos** para otros puntos candidatos, hasta que ya no se pueden encontrar nuevas cuartetos (p_1, p_2, p_3, p_4) , o hasta que un número prefijado de cuartetos fueron usadas.

Paso 7: En la matriz acumulador Ac se detectan cuatro máximos locales para cada elipse presente en la imagen. Estos máximos pertenecen a los cuatro puntos vértices de esta elipse que sean nuevamente denotados por $v_i = (x_{vi}, y_{vi})$, $i = 1, 2, 3, 4$. Entonces, los parámetros correctos h, k, a, b, θ de esta elipse cuya presencia en la imagen ya fue evidenciada, son calculados como sigue:

$$\begin{aligned}h &= \frac{x_{v1} + x_{v2} + x_{v3} + x_{v4}}{4} , & k &= \frac{y_{v1} + y_{v2} + y_{v3} + y_{v4}}{4} , \\a &= \frac{\sqrt{(x_{v2} - x_{v4})^2 + (y_{v2} - y_{v4})^2}}{2} , & b &= \frac{\sqrt{(x_{v1} - x_{v3})^2 + (y_{v1} - y_{v3})^2}}{2} , \\ \theta &= \tan^{-1} \left(\frac{y_{v2} - y_{v4}}{x_{v2} - x_{v4}} \right)\end{aligned}$$

En el caso especial del círculo ($\gamma = \pi/2$), las fórmulas para las coordenadas definitivas h, k del centro son las mismas que para la elipse general, y el radio definitivo del círculo está dado por

$$r = \frac{x_{v2} - x_{v4} + y_{v1} - y_{v3}}{4} .$$

Con eso finaliza el algoritmo.

Comentarios sobre este método:

El método aprovecha propiedades geométricas de la elipse y la dirección de gradiente de los puntos candidatos. Eso culmina en lograr una descomposición del espacio de cinco parámetros al utilizar una matriz acumulador de dimensión dos solamente. Aun así, el método utiliza muchos recursos computacionales, pues analiza toda la imagen hasta encontrar las cuartetos (p_1, p_2, p_3, p_4) , siendo dos parejas, los puntos de cada pareja teniendo la misma o la opuesta dirección de gradiente. Otra desventaja a resaltar es que el método sirve bien para detectar una sola elipse en la imagen de entrada. Una imagen con más elipses produciría más de cuatro máximos locales en la matriz Ac , y no es claro cómo separar todos estos máximos en grupos de cuatro, cada grupo correspondiente a una elipse detectada. El artículo [40] no hace mención alguna sobre este problema, a pesar de que presenta algunos ejemplos sencillos, supuestamente de experimentación, de imágenes con varios elipses.

Capítulo 4

Circularidad digital

4.1. Circularidad digital versus detección de círculos basada en geometría euclidiana

Cada círculo o arco circular en una imagen digital es un conjunto de píxeles. Bajo un arco circular, tanto en el plano euclidiano como también en el plano discreto, entendemos un subconjunto propio conexo de un círculo, es decir, un segmento de la curva circunferencia. Como fue explicado en la sección 1.4, un modelo de un *círculo digital* es el resultado de la discretización de intersección con la retícula de una circunferencia en \mathbb{R}^2 [5],[21]. Este también es llamado *círculo de Bresenham*, su representación analítica fue introducida en la sección 1.4, y se puede demostrar que siempre es un 8-camino cerrado, aunque no siempre es 8-irreducible.

Los métodos usados en el análisis de imágenes digitales para detectar círculos o arcos circulares, o para determinar si una curva o un k -camino es “circular”, no siempre se basan en la definición matemática estricta de un círculo digital.

Por ejemplo, en el análisis de imágenes microscópicas de células biológicas, un objetivo es distinguir entre objetos redondos o circulares, objetos alargados, y objetos cuyas fronteras presentan esquinas agudas o concavidades pronunciadas. Para lograr eso, los contornos de los objetos son determinados y luego, descriptores de estos contornos (y por tanto, de los objetos) son calculados. Para decidir si un objeto representado por su contorno C , es redondo o circular, los siguientes descriptores son herramientas estándar en muchos paquetes computacionales para el tratamiento de imágenes:

* Si las longitudes de los dos lados del rectángulo que circunscribe a C , o mejor, de todos los lados del octágono que circunscribe a C , son iguales, eso puede indicar que C es aproximadamente circular. Los datos de este rectángulo u octágono pueden ser fácilmente calculados durante el proceso de segmentación de la imagen, vea [36],[37].

* Si L es un estimador de la longitud del contorno, A un estimador del área del objeto, y D un estimador de su diámetro, entonces los descriptores calculados como $L^2/(4A)$ y $(4A)/(D^2)$ miden qué tan “redondo” es el objeto: Entre más cerca de π están sus valores, el objeto es más

similar a un disco. L puede ser el número de píxeles en C , A el número de píxeles en el objeto, y D puede ser calculado a partir de los datos del octágono que circunscribe a C , vea [36].

* Debido a que muchas células en imágenes microscópicas son objetos ovalados, son aproximadas por elipses, las cuales son adaptadas al contorno (o al octágono que circunscribe al objeto) mediante minimización del error por medio de mínimos cuadrados. Si la elipse obtenida así es un círculo, entonces el objeto es considerado circular.

Los métodos de detección de círculos o elipses basados en la transformación de Hough, tratados antes en esta tesis, hacen uso de las ecuaciones algebraicas de estas curvas, pero no aprovechan o toman en cuenta de manera explícita la naturaleza digital de las curvas a detectar. Los efectos de discretización están implícitos y contenidos en los errores que al final dificultan determinar los máximos locales en la matriz de acumulación. La ventaja de estos métodos es que los datos de entrada consisten en los puntos de borde de la imagen original; no se requiere que estos puntos estén organizados, por ejemplo, que formen curvas digitales o k -caminos.

A continuación se presentan algunas ideas básicas de un método de detección de círculos y de segmentación de 8-caminos en arcos circulares, el cual sí aprovecha la definición del círculo digital. Este método fue propuesto por Bhowmick [5],[6], originalmente para generar círculos en el plano discreto, de manera más eficiente que el algoritmo de Bresenham. El nuevo método proporciona algoritmos simples y asombrosamente eficientes de detección de círculos y de segmentación de curvas, pero necesita como dato de entrada un 8-camino irreducible, y reconoce circularidad solamente en el sentido estricto de la definición.

4.2. Definición de circularidad digital

Recordemos de la sección 1.4 que un círculo digital $\mathcal{C}^{\mathbb{Z}}(p, r)$ con centro $p \in \mathbb{Z}^2$ y radio $r \in \mathbb{N}$ está completamente definido por su primer octante, y se distingue de un círculo centrado en $o = (0, 0)$ sólo por una traslación. Por eso, aquí vamos a tratar al círculo y a formular las bases y el método mismo, sólo para el octante 1, y suponiendo que $p = o = (0, 0)$.

Un círculo $\mathcal{C}^{\mathbb{Z}}(o, r)$ es trazado desde su punto inicial $(0, r)$ hasta que termina el octante 1. En el punto inicial empieza la primera corrida horizontal, siendo la corrida superior (*top run*), cuyos puntos tienen la coordenada y máxima igual a r . Durante el trazado, todas las corridas son horizontales, y el paso del último punto de una corrida al primer punto de la siguiente, siempre tiene dirección de Freeman 7, como se puede apreciar en la figura 4.1. Eso es verdad siempre en el octante 1, a pesar de que el círculo completo puede ser un 8-camino no irreducible. Observe por ejemplo que los círculos con radios 4 y 11 tienen esquinas, es decir, su código de cadena contiene el segmento 06, mostrando que no son 8-irreducibles. Sin embargo, resulta que estas esquinas de un círculo digital solamente pueden ocurrir sobre las fronteras entre dos octantes. En consecuencia, la restricción al octante 1 garantiza que el código de longitudes de corridas caracteriza al círculo.

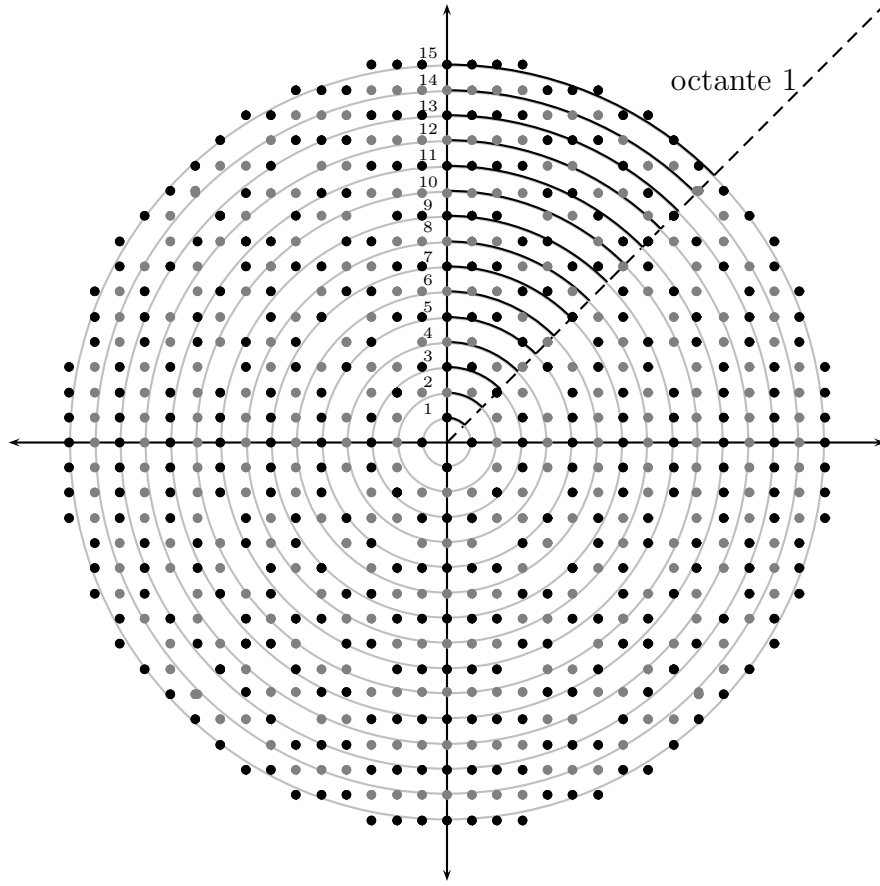


Figura 4.1: Discretización de intersección con la retícula de circunferencias euclidianas con radios $r = 1, \dots, 15$.

Ahora, sea $S = (s_0, s_1, \dots, s_n)$ un código de longitudes de corridas de un 8-camino correspondiente a una curva digital cuyo trazado avanza en corridas horizontales hacia la derecha, empezando con la primera corrida siendo la superior (coordenada y máxima), y llegando a la siguiente corrida siempre por un paso en dirección de Freeman 7. En [6], el problema de detectar si la secuencia dada S corresponde a un círculo digital o a un arco circular, es precisado en las siguientes dos versiones de definir circularidad:

Versión 1:

S tiene **circularidad digital** (S es digital circular) si existe un círculo digital $\mathcal{C}^{\mathbb{Z}}(o, r)$ cuyo código de longitudes de corridas coincide en sus primeros $n + 1$ números con S .

S tiene **circularidad digital parcial** si existen $m < n$ y un círculo digital $\mathcal{C}^{\mathbb{Z}}(o, r)$ cuyo código de longitudes de corridas coincide en sus primeros $m + 1$ números con los primeros $m + 1$ números de S .

La relación entre estas definiciones y el problema de detección de círculos es obvia: Si S

tiene circularidad digital, entonces S es un fragmento de círculo digital centrado en $o = (0, 0)$ con radio r , el cual inicia en el punto $(0, r)$ de la corrida superior. Si S tiene circularidad digital parcial, entonces para un $m < n$, la subsecuencia (s_0, s_1, \dots, s_m) es un fragmento de círculo digital dado por sus primeras $m + 1$ corridas. Sin embargo, el número s_{m+1} falla de ser la longitud de la $(m + 2)$ -ésima corrida del círculo determinado por sus primeras $m + 1$ corridas de longitudes (s_0, s_1, \dots, s_m) .

Cuando el análisis de la secuencia de entrada S revela circularidad parcial para un número m , es natural entonces investigar si la secuencia restante $(s_{m+1}, s_{m+2}, \dots, s_n)$ tiene circularidad o circularidad parcial, y así sucesivamente, para encontrar todos los círculos correspondientes a segmentos de la curva digital original. Sin embargo, para una curva dada y representada por S , en general, es demasiado restrictivo suponer que la parte del círculo digital correspondiente a la curva, o las partes de los varios círculos correspondientes a segmentos de esta curva, de casualidad siempre inicien en el primer punto de la corrida superior. Se espera más bien que los fragmentos de círculo detectados puedan iniciar en cualquier corrida, aunque recordemos que estamos tratando sólo el octante 1. Por eso, el problema general se relaciona con la siguiente definición:

Versión 2:

S tiene **circularidad digital** si existen un círculo digital $\mathcal{C}^{\mathbb{Z}}(o, r)$ y un número entero k^* con $0 \leq k^* \leq n$ tal que el código de longitudes de corridas del círculo

$(c_0, c_1, \dots, c_{k^*-1}, c_{k^*}, c_{k^*+1}, \dots, c_{k^*+n}, c_{k^*+n+1}, \dots)$ cumple que

$$c_{k^*} = s_0, c_{k^*+1} = s_1, c_{k^*+2} = s_2, \dots, c_{k^*+n} = s_n.$$

S tiene **circularidad digital parcial** si existen un círculo digital $\mathcal{C}^{\mathbb{Z}}(o, r)$ y números enteros k^*, m con $0 \leq k^* \leq m < n$ tales que el código de longitudes de corridas del círculo,

$(c_0, c_1, \dots, c_{k^*-1}, c_{k^*}, \dots, c_{k^*+m}, c_{k^*+m+1}, \dots, c_n, c_{n+1}, \dots)$, cumple que

$$c_{k^*} = s_0, c_{k^*+1} = s_1, c_{k^*+2} = s_2, \dots, c_{k^*+m} = s_m.$$

4.3. Caracterización y detección de circularidad digital

Para resolver el problema de detección de circularidad según la versión 1, en [6] fue presentado el siguiente teorema. Aparte de la función de redondeo hacia abajo que ya fue usada antes, recordemos la *función de redondeo hacia arriba*: $\lceil a \rceil = \min\{z \in \mathbb{Z} : z \geq a\}$, $a \in \mathbb{R}$.

Teorema 1. Sea $S = (s_0, s_1, \dots, s_n)$ el código de longitudes de corridas de una curva digital cuyo trazado avanza en corridas horizontales hacia la derecha, donde la primera corrida es la superior (coordenada y máxima), y se llega a la siguiente corrida siempre por un paso en

dirección de Freeman 7. Para $k = 0, 1, \dots, n$, sean

$$S_k = \sum_{j=0}^k s_j, \quad R_k = [r'_k, r''_k] \text{ (intervalo), donde}$$

$$r'_k = \left\lceil \frac{(S_k - 1)^2 + k(k+1) + 1}{2k+1} \right\rceil, \quad r''_k = \left\lfloor \frac{S_k^2 + k(k+1)}{2k+1} \right\rfloor.$$

La secuencia S coincide con los primeros $n+1$ números del código de longitudes de corridas de un círculo digital $\mathcal{C}^{\mathbb{Z}}(o, r)$ si y sólo si

$$r \in R = \bigcap_{k=0}^n R_k.$$

Si $R_0 \cap R_1 \cap \dots \cap R_n = \emptyset$, entonces no existe círculo digital cuyo código de longitudes de corridas coincide en sus primeros $n+1$ números con S .

El teorema reporta una propiedad del código de longitudes de corridas que es cumplida por cada círculo digital, pero que también es característica. Es decir, cuando una secuencia S cualquiera de números la cumple, entonces S (o cierta primera parte de S) corresponde a un círculo o a varios círculos con radios dentro de un intervalo. Eso hace clara su utilidad para detectar circularidad: dada una secuencia S , solamente hay que leer uno por uno los números s_0, s_1, \dots , efectuar los cálculos para obtener los intervalos R_k , y determinar la intersección $R = R_0 \cap R_1 \cap \dots \cap R_n$ que puede resultar vacía, o, proporciona el intervalo de radios de círculos presentes en S . En cada caso, eso da la respuesta al problema de detectar circularidad de S .

Analicemos qué significa el teorema para los primeros números de S :

$k = 0$:

$$S_0 = s_0, \quad R_0 = [r'_0, r''_0], \quad r'_0 = \left\lceil \frac{(s_0-1)^2+0+1}{1} \right\rceil = (s_0 - 1)^2 + 1, \quad r''_0 = \left\lfloor \frac{s_0^2+0}{1} \right\rfloor = s_0^2.$$

Eso significa que se requiere $r \in R_0 = [(s_0 - 1)^2 + 1, s_0^2]$ para asegurar que el número s_0 es la longitud de la primera corrida del círculo digital $\mathcal{C}^{\mathbb{Z}}(o, r)$ (Lema 2 de [6]). En consecuencia, si S tiene sólo el elemento s_0 , es decir $n = 0$, se sigue inmediatamente que S tiene circularidad digital, y el círculo detectado, en general, no es uno solo, sino es que se trata de todo un conjunto de círculos con radios dentro del intervalo $[(s_0 - 1)^2 + 1, s_0^2]$. Por ejemplo, para $s_0 = 6$ se obtiene $r \in [5^2 + 1, 6^2] = [26, 36]$. Es decir, la primera corrida de todos los círculos digitales cuyo radio entero está en el intervalo $[26, 36]$, consiste de exactamente seis puntos.

$k = 1$:

$$S_1 = s_0 + s_1, \quad R_1 = [r'_1, r''_1], \quad r'_1 = \left\lceil \frac{(S_1-1)^2+3}{3} \right\rceil, \quad r''_1 = \left\lfloor \frac{S_1^2+2}{3} \right\rfloor.$$

Si S tiene al menos los dos números s_0, s_1 , S puede tener circularidad digital, o S puede tener circularidad parcial hasta $m = 1$, solamente si $R_0 \cap R_1 \neq \emptyset$. En este caso, $R = R_0 \cap R_1$

nuevamente es un intervalo, en general más pequeño que R_0 y R_1 , el cual proporciona el rango de radios de los círculos detectados en S (Lema 3 de [6]).

Por ejemplo, para $s_0 = 6$ y $s_1 = 4$, ya vimos arriba que $R_0 = [26, 36]$. Luego, $S_1 = 10$, $r'_1 = \lceil \frac{9^2+3}{3} \rceil = 28$, $r''_1 = \lfloor \frac{10^2+2}{3} \rfloor = 34$, $R_1 = [28, 34]$, $R = R_0 \cap R_1 = [26, 36] \cap [28, 34] = [28, 34]$. Todos los círculos digitales cuyo radio entero está en el intervalo $[28, 34]$, tienen sus primeras dos corridas de longitudes 6, 4.

Para $s_0 = 6$ y $s_1 = 3$, se obtiene un resultado diferente: $S_1 = 9$, $r'_1 = \lceil \frac{8^2+3}{3} \rceil = 23$, $r''_1 = \lfloor \frac{9^2+2}{3} \rfloor = 27$, $R_1 = [23, 27]$, $R = R_0 \cap R_1 = [26, 36] \cap [23, 27] = [26, 27]$. Solamente los círculos digitales con radio 26 o 27, tienen sus primeras dos corridas de longitudes 6, 3.

Por último, supongamos $s_0 = 6$ y $s_1 = 2$. Entonces $S_1 = 8$, $r'_1 = \lceil \frac{7^2+3}{3} \rceil = 18$, $r''_1 = \lfloor \frac{8^2+2}{3} \rfloor = 22$, $R = R_0 \cap R_1 = [26, 36] \cap [18, 22] = \emptyset$. No existe círculo digital cuyas primeras dos corridas son de longitudes 6, 2.

El teorema de arriba se basa en estudios sobre la distribución de números cuadrados consecutivos realizados por los autores de [6], lo cual se relaciona directamente con la forma de un círculo digital descompuesto en sus corridas horizontales. Un hecho clave para poder deducir el teorema de arriba, es el siguiente (Lema 1 en [6]):

Lema 1. *Para todo círculo digital $\mathcal{C}^{\mathbb{Z}}(o, r)$ con código de longitudes de corridas $(s_0, s_1, s_2, \dots, s_n)$, y para todo $k = 0, 1, 2, \dots, n$, el intervalo $I_k = [u_k, v_k]$ dado por $u_k = \max\{0, (2k-1)r - k(k-1)\}$, $v_k = (2k+1)r - k(k+1) - 1$, contiene todos los números cuadrados (y solamente éstos) x^2 de las abscisas x de todos los píxeles (x, y) que pertenecen a la $k+1$ -ésima corrida (la que tiene longitud s_k) de $\mathcal{C}^{\mathbb{Z}}(o, r)$ en su octante 1.*

Por ejemplo, para el octante 1 del círculo $\mathcal{C}^{\mathbb{Z}}(o, r)$ con radio $r = 10$, vea la figura 4.1, tenemos $S = (4, 2, 1, 1)$. La primera corrida ($k = 0$) contiene los puntos $(x, 10)$ con $x \in \{0, 1, 2, 3\}$, y el intervalo $I_0 = [u_0, v_0] = [0, 9]$ contiene los cuatro números cuadrados consecutivos $0 = 0^2, 1 = 1^2, 4 = 2^2, 9 = 3^2$. La segunda corrida ($k = 1$) contiene $(x, 9)$ con $x \in \{4, 5\}$, y el intervalo $I_1 = [u_1, v_1] = [10, 3 \cdot 10 - 2 - 1] = [10, 27]$ contiene los siguientes dos números cuadrados consecutivos $16 = 4^2, 25 = 5^2$. La tercera corrida ($k = 2$) sólo contiene al punto $(6, 8)$, $I_2 = [u_2, v_2] = [3 \cdot 10 - 2, 5 \cdot 10 - 6 - 1] = [28, 43]$ contiene sólo al siguiente número cuadrado $36 = 6^2$. La última corrida ($k = 3$) tiene sólo el punto $(7, 7)$, $I_3 = [u_3, v_3] = [5 \cdot 10 - 6, 7 \cdot 10 - 12 - 1] = [44, 57]$ contiene sólo al cuadrado $49 = 7^2$. Eso completa el círculo en el octante 1, pues el siguiente punto debería tener coordenadas $x = 8$, $y = 6$ lo cual ya no cumple $x \leq y$.

Consideremos un ejemplo de aplicar el teorema para determinar si una secuencia dada de números S tiene circularidad digital.

Ejemplo: consideremos una curva digital dada por el siguiente código de longitudes de corridas horizontales, vea la figura 4.2:

$$S = (s_0 = 5, s_1 = 4, s_2 = 2, s_3 = 2, s_4 = 1).$$

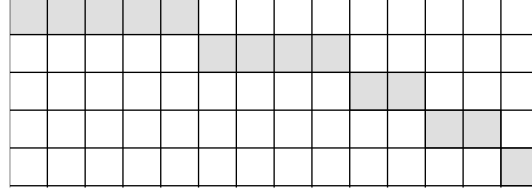


Figura 4.2: Ejemplo de curva digital.

$k = 0$: $S_0 = s_0 = 5$,

$$r'_0 = 4^2 + 1 = 17, r''_0 = 5^2 = 25, R = R_0 = [17, 25]$$

Todos los círculos digitales con radios en $[17, 25]$ tienen su primera corrida de la longitud s_0 .

$k = 1$: $S_1 = S_0 + s_1 = 5 + 4 = 9$,

$$r'_1 = \lceil \frac{8^2+3}{3} \rceil = 23, r''_1 = \lfloor \frac{9^2+2}{3} \rfloor = 27,$$

$$R_1 = [23, 27], R = R_0 \cap R_1 = [17, 25] \cap [23, 27] = [23, 25]$$

La subsecuencia (s_0, s_1) de S tiene circularidad digital. Todos los círculos digitales con radios en $[23, 25]$ tienen sus primeras dos corridas de las longitudes s_0, s_1 .

$k = 2$: $S_2 = S_1 + s_2 = 9 + 2 = 11$,

$$r'_2 = \lceil \frac{10^2+6+1}{5} \rceil = 22, r''_2 = \lfloor \frac{11^2+6}{5} \rfloor = 25,$$

$$R_2 = [22, 25], R = R_0 \cap R_1 \cap R_2 = [23, 25] \cap [22, 25] = [23, 25]$$

La subsecuencia (s_0, s_1, s_2) de S tiene circularidad digital. Todos los círculos digitales con radios en $[23, 25]$ tienen sus primeras tres corridas de las longitudes s_0, s_1, s_2 .

$k = 3$: $S_3 = S_2 + s_3 = 11 + 2 = 13$,

$$r'_3 = \lceil \frac{12^2+12+1}{7} \rceil = 23, r''_3 = \lfloor \frac{13^2+12}{7} \rfloor = 25,$$

$$R_3 = [23, 25], R = R_0 \cap R_1 \cap R_2 \cap R_3 = [23, 25] \cap [23, 25] = [23, 25]$$

La subsecuencia (s_0, s_1, s_2, s_3) de S tiene circularidad digital. Todos los círculos digitales con radios en $[23, 25]$ tienen sus primeras cuatro corridas de las longitudes s_0, s_1, s_2, s_3 .

$k = 4$: $S_4 = S_3 + s_4 = 13 + 1 = 14$,

$$r'_4 = \lceil \frac{13^2+20+1}{9} \rceil = 22, r''_4 = \lfloor \frac{14^2+20}{9} \rfloor = 22,$$

$$R_4 = [22, 24], R = R_0 \cap R_1 \cap R_2 \cap R_3 \cap R_4 = [23, 25] \cap [22, 24] = [23, 24]$$

Resultado: La secuencia completa $S = (s_0, s_1, s_2, s_3, s_4)$ tiene circularidad digital. Para todos los círculos digitales con radio 23 o 24, sus primeras cinco corridas tienen las longitudes iguales a los números s_0, s_1, s_2, s_3, s_4 .

Veamos otro ejemplo para mostrar cómo el teorema puede ser aplicado también para detectar circularidad parcial, y para segmentar una curva digital en segmentos que tienen circularidad, es decir, en partes correspondientes a fragmentos iniciales de círculos digitales.

Ejemplo: Consideremos una curva digital dada por el siguiente código de longitudes de corridas horizontales, vea la figura 4.3,

$$S = (s_0 = 8, s_1 = 5, s_2 = 4, s_3 = 2, s_4 = 4, s_5 = 3, s_6 = 2).$$

$$k = 0: S_0 = s_0 = 8,$$

$$r'_0 = 7^2 + 1 = 50, r''_0 = 8^2 = 64, R = R_0 = [50, 64]$$

$$k = 1: S_1 = S_0 + s_1 = 8 + 5 = 13,$$

$$r'_1 = \lceil \frac{12^2+2+1}{3} \rceil = 49, r''_1 = \lfloor \frac{13^2+2}{3} \rfloor = 57,$$

$$R_1 = [49, 57], R = R_0 \cap R_1 = [50, 64] \cap [49, 57] = [50, 57]$$

$$k = 2: S_2 = S_1 + s_2 = 13 + 4 = 17,$$

$$r'_2 = \lceil \frac{16^2+6+1}{5} \rceil = 53, r''_2 = \lfloor \frac{17^2+6}{5} \rfloor = 59,$$

$$R_2 = [53, 59], R = R_0 \cap R_1 \cap R_2 = [50, 57] \cap [53, 59] = [53, 57]$$

$$k = 3: S_3 = S_2 + s_3 = 17 + 2 = 19,$$

$$r'_3 = \lceil \frac{18^2+12+1}{7} \rceil = 49, r''_3 = \lfloor \frac{19^2+12}{7} \rfloor = 53,$$

$$R_3 = [49, 53], R = R_0 \cap R_1 \cap R_2 \cap R_3 = [53, 57] \cap [49, 53] = [53, 53]$$

Queda comprobado que S tiene circularidad parcial con $m = 3$. La subsecuencia (s_0, s_1, s_2, s_3) de S tiene circularidad digital, con una solución única: El círculo digital con radio 53, y sólo éste, tiene sus primeras cuatro corridas de longitudes iguales a s_0, s_1, s_2, s_3 .

$$k = 4: S_4 = S_3 + s_4 = 19 + 4 = 23,$$

$$r'_4 = \lceil \frac{22^2+20+1}{9} \rceil = 57, r''_4 = \lfloor \frac{23^2+20}{9} \rfloor = 61,$$

$$R_4 = [57, 61], R = R_0 \cap R_1 \cap R_2 \cap R_3 \cap R_4 = [53, 53] \cap [57, 61] = \emptyset.$$

Eso indica que la secuencia $(s_0, s_1, s_2, s_3, s_4)$ no tiene circularidad digital. No existe círculo digital tal que sus primeras cinco corridas tengan las longitudes s_0, s_1, s_2, s_3, s_4 . Entonces es conveniente separar S en la subsecuencia (s_0, s_1, s_2, s_3) cuya circularidad ya fue comprobada, y la secuencia restante (s_4, s_5, s_6) , para reiniciar el análisis según el teorema, con la secuencia de entrada $S' = (s_0 := s_4 = 4, s_1 := s_5 = 3, s_2 := s_6 = 2)$

$$k = 0: S_0 = s_0 = 4,$$

$$r'_0 = 3^2 + 1 = 10, r''_0 = 4^2 = 16, R = R_0 = [10, 16]$$

$$k = 1: S_1 = S_0 + s_1 = 4 + 3 = 7,$$

$$r'_1 = \lceil \frac{6^2+2+1}{3} \rceil = 13, r''_1 = \lfloor \frac{7^2+2}{3} \rfloor = 17,$$

$$R_1 = [13, 17], R = R_0 \cap R_1 = [16, 17]$$

$$k = 2: S_2 = S_1 + s_2 = 7 + 2 = 9,$$

$$r'_2 = \lceil \frac{8^2+6+1}{5} \rceil = 15, r''_2 = \lfloor \frac{9^2+6}{5} \rfloor = 17,$$

$$R_2 = [15, 17], R = R_0 \cap R_1 \cap R_2 = [16, 17]$$

Como resultado, la subsecuencia $S' = (s_4, s_5, s_6)$ de S tiene circularidad digital, con dos soluciones: Los círculos digitales con radio 16 y 17 tienen sus primeras tres corridas de longitudes iguales a s_4, s_5, s_6 . Los resultados de los cálculos son presentados en la Tabla 4.4.

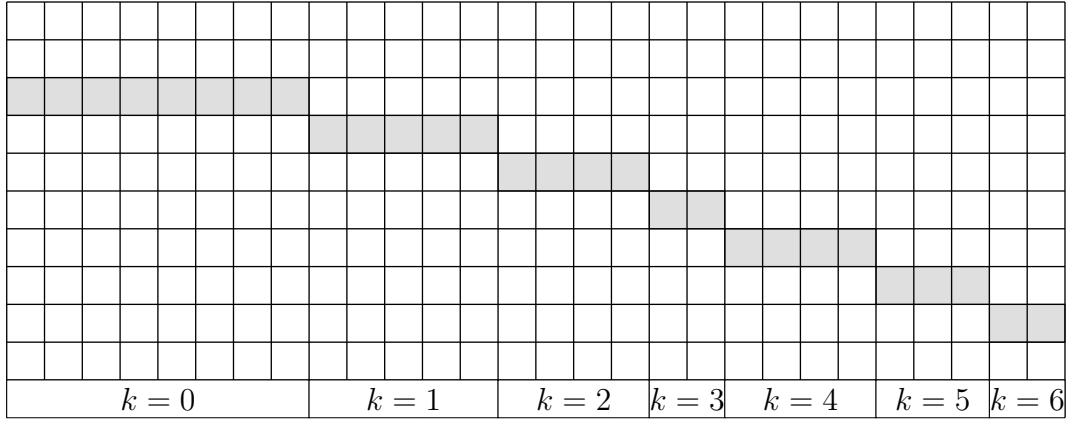


Figura 4.3: Curva digital con siete corridas ($k = 0, \dots, 6$) horizontales.

En resumen, la secuencia S fue segmentada en dos partes digitalmente circulares, la primera siendo fragmento de un círculo digital únicamente determinado con radio 53, la segunda parte correspondiente a dos posibles círculos digitales, con radios 16 o 17.

Corrida	Longitud	Intervalo R_k	$R_0 \cap R_1 \cap \dots \cap R_k$
$k = 0$	$s_0 = 8$	$[50, 64]$	$[50, 64]$
$k = 1$	$s_1 = 5$	$[49, 57]$	$[50, 57]$
$k = 2$	$s_2 = 4$	$[53, 59]$	$[53, 57]$
$k = 3$	$s_3 = 2$	$[49, 53]$	$[53, 53]$
$k = 4$	$s_4 = 4$	$[57, 61]$	\emptyset
$k = 0$	$s_0 := s_4 = 4$	$[10, 16]$	$[10, 16]$
$k = 1$	$s_1 := s_5 = 3$	$[13, 17]$	$[16, 17]$
$k = 2$	$s_2 := s_6 = 2$	$[15, 17]$	$[16, 17]$

Figura 4.4: Tabla de radios del segmento $S = \{8, 5, 4, 2, 4, 3, 2\}$.

Los ejemplos anteriores muestran que el tratamiento de cada caso $k = 0, k = 1, \dots, k = n$, es un paso del algoritmo basado en el teorema. Para $k = 0$, el paso consiste en determinar $R := R_0$. Para cada $k \geq 1$, el nuevo intervalo R_k es determinado, y el intervalo R es actualizado como $R := R \cap R_k$. Si denotamos $R = [r', r'']$ y $R_k = [s', s'']$, la intersección de estos dos intervalos puede ser calculada por simples comparaciones:

Si $s'' < r'$ o $r'' < s'$ entonces $R \cap R_k = \emptyset$.

En otro caso,

$$R \cap R_k = [r', r''] \cap [s', s''] = [\max\{s', r'\}, \min\{s'', r''\}],$$

es decir, la actualización en el algoritmo debe ser $r' := \max\{s', r'\}$, $r'' := \min\{s'', r''\}$. En consecuencia, el algoritmo basado en el teorema es como sigue:

Algoritmo de detección de circularidad digital y de segmentación de una curva digital en arcos circulares (versión 1: iniciando con la corrida superior) [6]:

Datos de entrada: Secuencia de entrada $S = (s_0, s_1, s_2, \dots, s_n)$ - código de longitudes de corridas horizontales de una curva digital.

Datos de salida: Rango de radios de círculos digitales presentes en S si S tiene circularidad digital. En otro caso, segmentación de S en subsecuencias digitalmente circulares, y para cada parte, el rango de radios de círculos es reportado.

1. $\Sigma := s_0$, $r' := (s_0 - 1)^2 + 1$, $r'' = s_0^2$,

$N :=$ el número de elementos en la secuencia de entrada.

2. Para todo $k = 1, 2, \dots, N$:

3. $\Sigma := \Sigma + s_k$, $s' := \left\lceil \frac{(\Sigma-1)^2 + k(k+1) + 1}{2k+1} \right\rceil$, $s'' = \left\lfloor \frac{\Sigma^2 + k(k+1)}{2k+1} \right\rfloor$

4. Si $s'' < r'$ o $r'' < s'$,

reportar resultado: S tiene circularidad parcial hasta la $(k-1)$ -ésima corrida, la subsecuencia $(s_0, s_1, \dots, s_{k-1})$ tiene circularidad con el rango de radios $[r', r'']$. Reiniciación (ir a paso 1) con la secuencia restante ($s_0 := s_k, s_1 := s_{k+1}, \dots, s_{n-k} := s_n$) como dato de entrada.

En otro caso: seguir con paso 5.

5. $[r', r''] := [\max\{s', r'\}, \min\{s'', r''\}]$ (actualización del intervalo de radios), seguir con paso 2 (aumentando k).

6. Reportar resultado: la secuencia de entrada (S o la última subsecuencia tratada) tiene circularidad digital con el rango de radios $[r', r'']$.

El algoritmo de arriba para solucionar la versión 1 del problema de detección de circularidad resulta ser extremadamente eficiente, puesto que solamente realiza cálculos muy sencillos.

La versión 2 de la definición de circularidad digital es mucho más realista en relacion a problemas a resolver en la práctica del tratado de imágenes digitales. En [6], un algoritmo muy eficiente correspondiente a la versión 2 es presentado. Sin embargo, sus bases matemáticas y computacionales son aún más elaboradas que el algoritmo de la versión 1, y se salen de los objetivos de esta tesis.

Cabe notar también que aquí presentamos el algoritmo de la versión 1 solamente para el octante 1.

Además, para una implementación de este método, aun siendo sólo de la versión 1, se tendría que agregar el análisis de la curva digital dada, para detectar en cuál octante se encuentra, y luego poder aplicar una versión adaptada del algoritmo para este octante, aprovechando las simetrías que existen entre todos los octantes.

Capítulo 5

Ajuste de círculos a conjuntos de puntos

El ajuste de círculos o elipses, o más generalmente, de curvas, a conjuntos de puntos que constituyen los datos de entrada, corresponde a una gran clase de métodos basados en la optimización matemática y la estadística, en la literatura en idioma inglés llamado *circle (resp., ellipse, curve) fitting*. La implementación y aplicación de estos métodos son laboriosos y por lo general, involucran métodos numéricos, en su mayoría, iterativos. Sin embargo, como en el caso de los métodos basados en la transformada de Hough, una gran ventaja es que no se requiere tener algún orden o estructura de organización de los datos de entrada. Eso define una amplia variedad de áreas donde son aplicables y apropiados. En particular, para analizar nubes de puntos que son conjuntos de puntos desorganizados y con falta de conectividad, el ajuste de curvas puede contribuir a detectar estas curvas que, de manera extremadamente interrumpida, estén presentes en los datos. Este capítulo reporta algunas ideas básicas de estos métodos.

5.1. Ajuste de una curva a un conjunto de puntos

Dados m puntos en el plano, $p_i = (x_i, y_i) \in \mathbb{R}^2$, $i = 1, 2, \dots, m$, la tarea es ajustar una curva, por ejemplo, un círculo o una elipse, a estos puntos. El término *ajustar*, se refiere a que la curva resultante sea óptima de alguna manera. También se habla de una *adaptación* de la curva a los datos. Eso significa que la curva tenga una distancia mínima de todos los puntos p_i , o que todos los puntos p_i satisfagan lo mejor posible, cierta representación de la curva. En consecuencia, el ajuste de una curva depende de cómo medir distancia, y de cómo representar la curva. La restricción fuerte aquí es el tipo de curva, la cual, por ejemplo, tiene que ser un círculo exacto; la tarea entonces consiste en determinar los parámetros de dicho círculo.

Un ajuste basado en una función de distancia d , definida en el plano euclidiano \mathbb{R}^2 , para poder medir la distancia entre cada punto p_i y la curva buscada C , es llamado en la

literatura un **mejor ajuste** (*best fit*) o **ajuste geométrico** (*geometric fit*). Este ajuste, en general, es modelado como un problema de minimización de los errores cuadráticos (vea [10],[15],[25],[34]):

$$\sum_{i=1}^m (d(p_i, C))^2 \longrightarrow \text{mín}$$

Eso es justificado porque en el caso ideal de que todos los puntos p_i se encuentren sobre la curva buscada C , las distancias $d(p_i, C)$ tienen valor cero; pero en el caso general, $d(p_i, C)$ es considerado como un error causado por el punto p_i .

La curva buscada C es determinada por n parámetros u_1, u_2, \dots, u_n los cuales, o pueden describir características geométricas de la curva, o bien, pueden ser parámetros abstractos en una ecuación algebraica. En el capítulo 2 vimos para los casos de círculos y elipses, que cualquier punto p de la curva C , satisface una ecuación algebraica

$$F(p) = F(p, u_1, u_2, \dots, u_n) = 0 .$$

Para ajustar los parámetros u_1, u_2, \dots, u_n a los puntos dados p_1, p_2, \dots, p_m , una idea es imponer el requerimiento que para todo $i = 1, 2, \dots, m$,

$$F(p_i, u_1, u_2, \dots, u_n) = 0 ,$$

siendo eso un sistema de m ecuaciones para n incógnitas. En general, m es mucho más grande que n , por lo cual, no se puede esperar obtener una solución exacta de este sistema. Por eso, las m expresiones obtenidas para $F(p_i, u_1, u_2, \dots, u_n)$, siendo en general no iguales a cero, nuevamente son consideradas como errores. El ajuste modelado por el problema de minimización

$$\sum_{i=1}^m (F(p_i, u_1, u_2, \dots, u_n))^2 = \sum_{i=1}^m |F(p_i, u_1, u_2, \dots, u_n)|^2 \longrightarrow \text{mín}$$

es llamado **ajuste algebraico** (*algebraic fit*) de la curva [15] **mediante la minimización de los errores cuadráticos** (*Least Squares Problem*). La función F en general no es lineal, entonces el problema es no lineal, y para su solución, son usados métodos numéricos iterativos, por ejemplo, el método de Gauss-Newton o variantes de éste [10],[15],[25],[34].

La minimización de los errores cuadráticos constituye el método más usado y tratado en la literatura para el ajuste de curvas a conjuntos de puntos. Sin embargo, el problema de minimización puede ser formulado usando otras representaciones del error como las siguientes:

$$\text{Error máximo:} \quad \text{máx}\{|F(p_i, u_1, u_2, \dots, u_n)|, i = 1, \dots, m\} \longrightarrow \text{mín}$$

$$\text{Error promedio:} \quad \frac{1}{m} \sum_{i=1}^m |F(p_i, u_1, u_2, \dots, u_n)| \longrightarrow \text{mín}$$

$$\text{Error cuadrático medio:} \quad \frac{1}{m} \sum_{i=1}^m |F(p_i, u_1, u_2, \dots, u_n)|^2 \longrightarrow \text{mín}$$

Para representaciones de errores distintas, los problemas de minimización tienen soluciones distintas.

5.2. Ajuste algebraico del círculo

Un círculo es determinado por tres parámetros geométricos: las dos coordenadas del centro y el radio. En su representación algebraica general, la cual es usada por ejemplo en [15], estos parámetros geométricos no son reconocibles inmediatamente, pero pueden ser recuperados como veremos a continuación. Los autores de [15] emplean una representación algebraica muy general la cual simplificamos aquí a la siguiente, denotando cualquier punto del círculo como $p = (x, y) \in \mathbb{R}^2$ y sus parámetros algebraicos por $a, b, t \in \mathbb{R}$:

$$F(p, a, b, t) = F(x, y, a, b, t) = (x, y) \begin{pmatrix} x \\ y \end{pmatrix} + (a, b) \begin{pmatrix} x \\ y \end{pmatrix} + t = x^2 + y^2 + ax + by + t = 0 .$$

Esta ecuación puede ser transformada en otra equivalente para reconocer los parámetros geométricos:

$$\begin{aligned} F(x, y, a, b, t) &= x^2 + y^2 + ax + by + t = 0 \iff \\ F(x, y, a, b, t) &= x^2 + ax + y^2 + by + t \\ &= \left(x^2 + ax + \frac{a^2}{4} \right) + \left(y^2 + by + \frac{b^2}{4} \right) + t - \frac{a^2}{4} - \frac{b^2}{4} \\ &= \left(x + \frac{a}{2} \right)^2 + \left(y + \frac{b}{2} \right)^2 + t - \frac{1}{4}(a^2 + b^2) = 0 \\ \iff & (x - c_1)^2 + (y - c_2)^2 - r^2 = 0 , \\ & \text{para } c_1 = -\frac{a}{2}, \quad c_2 = -\frac{b}{2}, \quad r^2 = \frac{a^2 + b^2}{4} - t \end{aligned}$$

La ecuación $(x - c_1)^2 + (y - c_2)^2 - r^2 = 0$ es la representación algebraica del círculo que conocimos en el capítulo 2, y las últimas formulas de arriba establecen las relaciones entre los parámetros geométricos, siendo c_1, c_2 las coordenadas del punto centro y el radio r , y los parámetros algebraicos a, b, t .

El ajuste algebraico pretende determinar los parámetros algebraicos a, b, t como solución del sistema $F(x, y, a, b, t) = 0$. Tomando en cuenta los datos de entrada que consisten de los puntos $p_i = (x_i, y_i)$, $i = 1, 2, \dots, m$, tenemos m ecuaciones de la forma

$$F(x_i, y_i, a, b, t) = x_i^2 + y_i^2 + ax_i + by_i + t = 0 .$$

Los datos de entrada proporcionan los números reales x_i, y_i , $i = 1, 2, \dots, m$. La función F es lineal en sus variables (incógnitas) a, b, t . Debido a que este sistema para las tres incógnitas a, b, t , es sobre determinado, es decir, m es mucho mayor a 3, la solución no es esperada como exacta, sino, es buscada como solución del siguiente problema de minimización:

$$\sum_{i=1}^m (F(x_i, y_i, a, b, t))^2 = \sum_{i=1}^m (x_i^2 + y_i^2 + ax_i + by_i + t)^2 \longrightarrow \text{mín}$$

5.3. Ajuste geométrico del círculo

Recordando de nuevo la ecuación algebraica $(x - c_1)^2 + (y - c_2)^2 = r^2$ de un círculo C , en términos de sus parámetros geométricos, de la sección 2.1.2, es evidente que todo punto $p = (x, y) \in C$ tiene distancia euclidiana

$$d((x, y), (c_1, c_2)) = r = \sqrt{(x - c_1)^2 + (y - c_2)^2}$$

del punto centro (c_1, c_2) . Un punto $p_i = (x_i, y_i)$ del conjunto de los datos de entrada, en general, no satisface exactamente esta ecuación. Es decir, p_i tiene una distancia, probablemente distinta de cero, al círculo C , la cual coincide con la diferencia entre $d((x_i, y_i), (c_1, c_2))$ y r :

$$d(p_i, C) = \inf\{d(p_i, q) : q \in C\} = \left| r - \sqrt{(x_i - c_1)^2 + (y_i - c_2)^2} \right|$$

El modelo más usado en la literatura del ajuste geométrico del círculo es dado por el problema de minimización de los errores cuadráticos [10],[15],[25],[34]:

$$\begin{aligned} \sum_{i=1}^m (d(p_i, C))^2 &= \sum_{i=1}^m \left| r - \sqrt{(x_i - c_1)^2 + (y_i - c_2)^2} \right|^2 \\ &= \sum_{i=1}^m \left(r - \sqrt{(x_i - c_1)^2 + (y_i - c_2)^2} \right)^2 \longrightarrow \text{mín} \end{aligned}$$

Eso puede ser interpretado también como un ajuste general (algebraico) para la representación del círculo C dada por

$$F(x, y, c_1, c_2, r) = r - \sqrt{(x - c_1)^2 + (y - c_2)^2} = 0, \quad (x, y) \in C.$$

Para todo punto de entrada $p_i = (x_i, y_i)$, si $p_i \in C$ entonces $F(x_i, y_i, c_1, c_2, r) = 0$, en otro caso, el número $F(x_i, y_i, c_1, c_2, r)$ es un error causado por p_i . Notemos que ahora, la función F no es lineal en sus incógnitas c_1, c_2, r ; se trata de un problema no lineal de minimización:

$$S(c_1, c_2, r) := \sum_{i=1}^m (F(x_i, y_i, c_1, c_2, r))^2 \longrightarrow \text{mín}$$

Siguiendo el método estándar para determinar los parámetros c_1, c_2, r tales que F adquiriera valores minimales, el problema se convierte en la búsqueda de raíces de las derivadas parciales:

$$\begin{aligned} \frac{\partial S(c_1, c_2, r)}{\partial r} &= 2 \sum_{i=1}^m \left(r - \sqrt{(x_i - c_1)^2 + (y_i - c_2)^2} \right) \cdot (1 - 0) \\ &= -2 \left(\sum_{i=1}^m \sqrt{(x_i - c_1)^2 + (y_i - c_2)^2} \right) + 2mr = 0, \end{aligned}$$

lo cual tiene la siguiente solución:

$$r = \frac{1}{m} \sum_1^m \sqrt{(x_i - c_1)^2 + (y_i - c_2)^2} .$$

En conclusión, si ya tuviéramos conocimiento del punto centro del círculo que se ajuste al conjunto de puntos dados (x_i, y_i) , $i = 1, \dots, m$, sería sencillo calcular su radio.

Analicemos ahora las derivadas con respecto a los parámetros c_1 y c_2 .

$$\begin{aligned} \frac{\partial S(c_1, c_2, r)}{\partial c_1} &= \sum_1^m 2 \left(r - \sqrt{(x_i - c_1)^2 + (y_i - c_2)^2} \right) \cdot \\ &\quad \cdot \left(0 - \frac{1}{2} \cdot [(x_i - c_1)^2 + (y_i - c_2)^2]^{-\frac{1}{2}} \cdot [2(x_i - c_1)(-1) + 0] \right) \\ &= 2 \sum_1^m \left(r - \sqrt{(x_i - c_1)^2 + (y_i - c_2)^2} \right) \cdot [(x_i - c_1)^2 + (y_i - c_2)^2]^{-\frac{1}{2}} \cdot (x_i - c_1) \\ &= 2 \sum_1^m [(x_i - c_1)^2 + (y_i - c_2)^2]^{-\frac{1}{2}} \cdot (x_i - c_1) \cdot r - 2 \sum_1^m (x_i - c_1) \\ &= 2r \sum_1^m \frac{x_i - c_1}{\sqrt{(x_i - c_1)^2 + (y_i - c_2)^2}} - 2 \sum_1^m x_i + 2 \sum_1^m c_1 \\ &= 2r \sum_1^m \frac{x_i - c_1}{\sqrt{(x_i - c_1)^2 + (y_i - c_2)^2}} - 2 \sum_1^m x_i + 2mc_1 = 0 \end{aligned}$$

Análogamente se obtiene

$$\frac{\partial S(c_1, c_2, r)}{\partial c_2} = 2r \sum_1^m \frac{y_i - c_2}{\sqrt{(x_i - c_1)^2 + (y_i - c_2)^2}} - 2 \sum_1^m y_i + 2mc_2 = 0$$

Denotando los valores medios de las coordenadas x , y de las coordenadas y de todos los puntos p_i por \bar{x} y \bar{y} , respectivamente,

$$\bar{x} := \frac{1}{m} \sum_1^m x_i , \quad \bar{y} := \frac{1}{m} \sum_1^m y_i ,$$

podemos escribir como en [34]:

$$\begin{aligned} \frac{\partial S(c_1, c_2, r)}{\partial c_1} &= 2r \sum_1^m \frac{x_i - c_1}{\sqrt{(x_i - c_1)^2 + (y_i - c_2)^2}} - 2m\bar{x} + 2mc_1 = 0 \\ \frac{\partial S(c_1, c_2, r)}{\partial c_2} &= 2r \sum_1^m \frac{y_i - c_2}{\sqrt{(x_i - c_1)^2 + (y_i - c_2)^2}} - 2m\bar{y} + 2mc_2 = 0 \end{aligned}$$

Para resolver estas dos ecuaciones, conjunto con la ecuación para r , todas simultáneamente, no existe fórmula para una solución (c_1, c_2, r) [34]. Este tal llamado *problema pleno de los mínimos cuadrados* (*Full Least Squares Method*) tiene que ser resuelto mediante métodos numéricos. Según experiencias reportadas en la literatura [15],[34], eso es laborioso y computacionalmente pesado para conjuntos grandes de datos de entrada. Otro problema importante en la determinación del centro (c_1, c_2) es que este método resulta tener gran sensibilidad con respecto a puntos dados que se encuentran muy lejos del círculo buscado. Tales puntos “no esperados”, llamados en inglés “*outliers*”, pueden estar presentes en los datos de entrada, aunque son puntos erróneos. Cada uno de estos puntos produce un gran error que, evaluado al cuadrado, entra al problema de optimización y puede causar un desvío substancial del resultado [34]. Para mejorar el desempeño del método, muchas variantes han sido propuestas en la literatura, donde propiedades geométricas son aprovechadas para desarrollar otras medidas del error, o donde los datos de entrada son usados selectivamente. Por ejemplo, una de las variantes en [34] emplea pesos en las ecuaciones para suprimir pares de puntos p_i, p_j cuando se encuentran demasiado cerca uno del otro.

Una alternativa propuesta en [34] es usar algún método distinto a la minimización de errores, para determinar al punto centro (c_1, c_2) , y después, calcular el radio mediante la fórmula sencilla que fue obtenida arriba, $r = \frac{1}{m} \sum_{i=1}^m \sqrt{(x_i - c_1)^2 + (y_i - c_2)^2}$. Este método será tratado en la siguiente sección.

5.4. Determinación del centro del círculo a partir de tripletas de puntos

Para determinar el centro de un círculo que se ajusta al conjunto dado de puntos $p_i = (x_i, y_i)$, $i = 1, \dots, m$, en [34] y [25], fue propuesto un método basado en cálculos efectuados para tripletas de puntos seleccionados.

La base de este método es la propiedad geométrica de que cualesquiera tres puntos que pertenecen que no son colineales, determinan de manera única al centro de C , proporcionando luego como resultado obvio también al radio.

Eso es verdad debido a que para toda secante, la línea recta que bisecta la secante y es perpendicular a esta, pasa por el centro del círculo. En consecuencia, para cada triplete de puntos no colineales $p_1 = (x_1, y_1)$, $p_2 = (x_2, y_2)$, $p_3 = (x_3, y_3)$ que pertenecen a una circunferencia C , las secantes $\overline{p_1 p_2}$ y $\overline{p_2 p_3}$ no son paralelas y se intersecan en el punto centro (c_1, c_2) de C , vea la figura 5.1. En otras palabras, los bisectores perpendiculares de los lados del triángulo no degenerado formado por sus vértices p_1, p_2, p_3 , se intersecan en el punto circuncentro, lo cual es el contenido del Teorema del Circuncentro conocido de la geometría del plano.

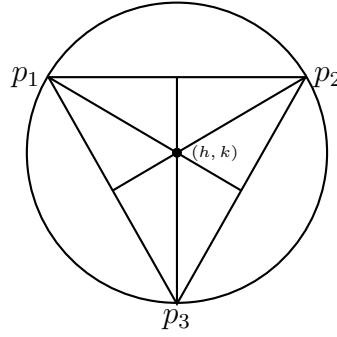


Figura 5.1: Tres puntos sobre una circunferencia forman un triángulo cuyo circuncentro coincide con el centro del círculo.

Las coordenadas del centro (c_1, c_2) determinadas por la tripleta $p_1 = (x_1, y_1)$, $p_2 = (x_2, y_2)$, $p_3 = (x_3, y_3)$ pueden ser calculadas como sigue (vea [34]):

$$c_1 = \frac{1}{2} \left(\frac{x_1^2(y_2 - y_3) + x_2^2(y_3 - y_1) + x_3^2(y_1 - y_2) - (y_1 - y_2)(y_2 - y_3)(y_3 - y_1)}{x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2)} \right)$$

$$c_2 = \frac{1}{2} \left(\frac{y_1^2(x_2 - x_3) + y_2^2(x_3 - x_1) + y_3^2(x_1 - x_2) - (x_1 - x_2)(x_2 - x_3)(x_3 - x_1)}{y_1(x_2 - x_3) + y_2(x_3 - x_1) + y_3(x_1 - x_2)} \right)$$

Luego, a partir de estos tres puntos, el radio del círculo puede ser determinado como

$$r = \frac{1}{3} \sum_{j=1}^3 \sqrt{(x_j - c_1)^2 + (y_j - c_2)^2} \quad .$$

Usando estas fórmulas, el método propuesto en [34],[25] efectúa el cálculo del centro para un número (grande) k de tripletas de puntos seleccionadas del conjunto de puntos $p_i = (x_i, y_i)$, $i = 1, \dots, m$ que constituyen los datos de entrada. Así son obtenidas k estimaciones

$$(c_{1,1}, c_{2,1}), (c_{1,2}, c_{2,2}), (c_{1,3}, c_{2,3}), \dots, (c_{1,k}, c_{2,k})$$

del punto centro del círculo buscado. También podrían ser determinadas k estimaciones del radio $r_1, r_2, r_3, \dots, r_k$ por la fórmula de arriba para r . Al final, las coordenadas del centro del círculo buscado son calculadas promediando las k estimaciones:

$$c_1 = \frac{1}{k} \sum_{i=1}^k c_{1,i} \quad , \quad c_2 = \frac{1}{k} \sum_{i=1}^k c_{2,i}$$

Similarmente, el radio podría ser determinado como

$$r = \frac{1}{k} \sum_{i=1}^k r_i \quad ,$$

pero los autores de [34] prefieren usar las coordenadas finales del centro ya determinadas c_1, c_2 , y los datos de entrada originales, como ya fue mencionado arriba:

$$r = \frac{1}{m} \sum_{i=1}^m \sqrt{(x_i - c_1)^2 + (y_i - c_2)^2} \quad .$$

Dentro del trabajo de esta tesis, para fines educativos, fue elaborado un programa en MATLAB que determina y visualiza el circuncentro de toda tripleta de puntos, dados por el usuario. De eso se aprendió que, para cualquier implementación del método propuesto en [34], toda tripleta dada de puntos tiene que ser evaluada para asegurar que los puntos no sean colineales, antes de someterla a los cálculos. Otra observación es que, un conjunto de puntos siendo los datos relacionados con algún problema de aplicación del análisis de imágenes digitales, puede ser grande, es decir, m probablemente es un número grande. Los autores de [34] proponen calcular las k estimaciones del centro para todas las tripletas posibles que se puedan formar usando los datos de entrada. Suponiendo que todas estas tripletas aprueban ser no colineales y por lo tanto, son usadas, su número está dado por

$$k = \binom{m}{3} = \frac{m!(m-3)!}{3!} \quad .$$

A primera vista no es obvio que este número crece rapidamente conforme crece m . Veamos unos ejemplos:

$$\begin{aligned} \binom{3}{3} &= 1, \quad \binom{4}{3} = \frac{4!1!}{3!} = 4, \quad \binom{5}{3} = \frac{5!2!}{3!} = \frac{5 \cdot 4 \cdot 3! \cdot 2!}{3!} = 40 \\ \binom{m}{3} &= \frac{m!(m-3)!}{3!} = m \cdot (m-1) \cdot \dots \cdot 4 \cdot (m-3)! \quad \text{para } m \geq 4 \\ \binom{10}{3} &= 10 \cdot 9 \cdot 8 \cdot 7 \cdot \dots \cdot 4 \cdot (7!) = 604800 \cdot (7!) \end{aligned}$$

Eso deja ver que para solamente 10 puntos de entrada, el número de tripletas posibles ya es enorme. Para datos que consisten de cientos o miles de puntos, la consideración de todas las tripletas está fuera de lo posible en la práctica.

Capítulo 6

Experimentos

En este capítulo, se describen implementaciones de algunos métodos de detección de circunferencias que fueron elaboradas como parte de esta tesis. Asimismo se mostrarán resultados de experimentación obtenidos al usar estos programas, los cuales fueron programados en MATLAB como GUIs (*graphical user interface*) y se pueden dividir en dos tipos de métodos:

Los programas del primer tipo de métodos usan como dato de entrada una imagen digital. Fueron implementados los siguientes cuatro métodos, seleccionables por el usuario mediante una interfaz gráfica:

H3D: el **método general Hough 3D**, descrito en la sección 3.3.1;

DG: el **método para la detección de círculos basado en valor y dirección de gradiente**, que fue presentado en la sección 3.3.6;

2P: el **método de dos puntos finales de una cuerda**, explicado en la sección 3.3.6; y

4P: el **método geométrico con dirección de gradiente** para detectar círculos, el cual fue detallado en la sección 3.4.2.

El programa del segundo tipo de métodos usa como dato de entrada una secuencia de números correspondiente a un código de longitudes de corridas horizontales que describen los avances de una curva digital. Este programa aplica la **versión 1 del Algoritmo de Bhowmick** que fue descrito en la sección 4.3), para determinar si la curva digital tiene circularidad digital o circularidad digital parcial.

6.1. Programas de cuatro métodos para la detección de circunferencias en imágenes digitales

Se desarrollaron cuatro programas, como implementación de los cuatro métodos, H3D, DG, 2P y 4P, mencionados arriba. Con estos programas, fueron efectuados experimentos con diversas imágenes: imágenes generadas en computadora en escala de grises, empezando con una imagen de una sola circunferencia, luego, se analizaron imágenes con más circunferencias, incluso con círculos concéntricos. Después, una imagen obtenida de internet conteniendo círculos pero también muchas otras curvas, fue analizada.

Para que los programas funcionen de manera correcta y eficiente, las siguientes condiciones y restricciones deben ser satisfechas:

- Los formatos de la imagen de entrada permitidos son los siguientes: .jpg, .png.; las imágenes deben ser en escala de grises, binarias o de tipo RGB.
- La matriz que representa la imagen debe tener valores de datos reales y del tipo single, double, uint8 o uint16.
- La imagen de entrada no debe tener mucho ruido. En caso de tenerlo, es recomendable efectuar un preprocesamiento apropiado antes de someter la imagen a cualquier de los cuatro programas.
- El tamaño de la imagen a tratar no debe exceder las dimensiones de 1000x1000 píxeles.
- La detección de circunferencias se limita a un rango de radios preestablecido por defecto como el intervalo $[rmin, rmax]$, donde

$$rmin = 10 \text{ píxeles} , rmax = \lfloor \max\{\frac{filas}{2}, \frac{columnas}{2}\} \rfloor \text{ píxeles}$$

donde *filas* y *columnas* hace referencia al tamaño de la matriz que describe la imagen.

- Los programas no detectan círculos con centros fuera del dominio de la imagen.

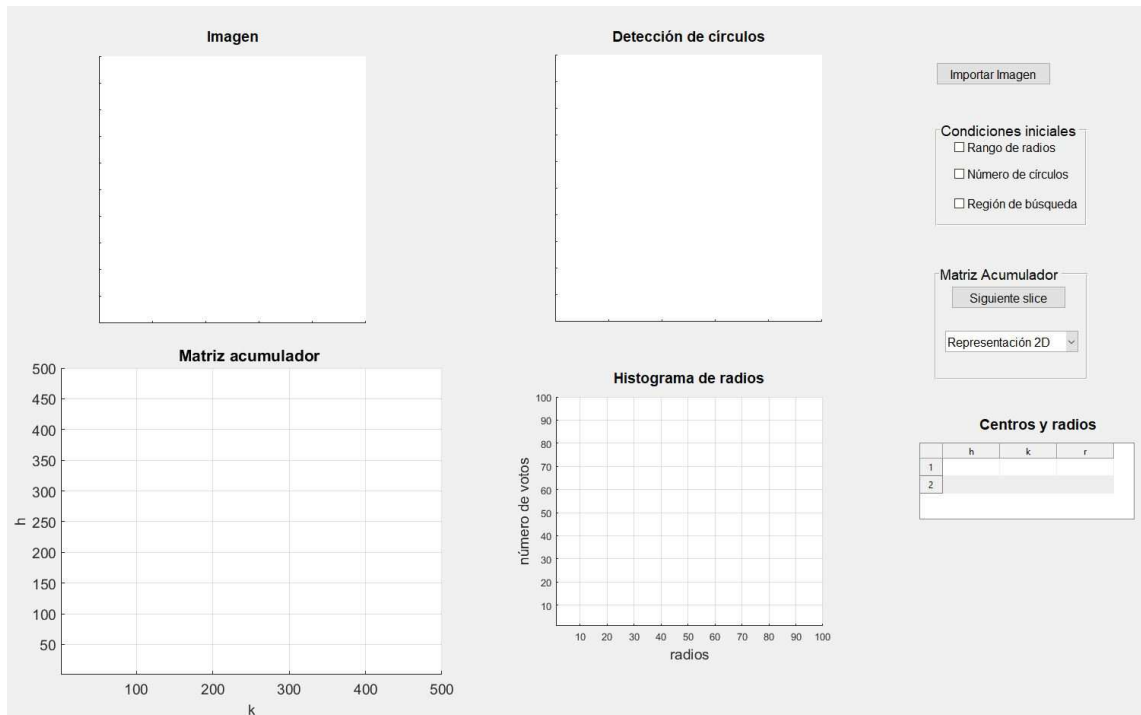


Figura 6.1: Interfaz gráfica.

La figura 6.1 muestra la interfaz gráfica general, en la cual, se puede notar que hay cuatro cuadros:

- * *Imagen* muestra la imagen original que se ha cargado para ser analizada.
- * *Detección de círculos* exhibe los círculos encontrados, superpuestos en la imagen original.
- * *Matriz acumulador* presenta las votaciones dentro de la matriz acumulador para los parámetros h, k , siendo las coordenadas del punto centro del círculo.
- * *Histograma de radios* ofrece los radios encontrados en la imagen.

También aparecen en la interfaz gráfica un botón, dos paneles de control y una tabla.

- * El botón *Importar imagen* permite cargar la imagen que se analizará.
- * El panel *Representación de Acc* da la opción de mostrar la matriz acumulador de los parámetros h, k en vista 3D o 2D.
- * *Condiciones iniciales* sirve para particularizar el resultado esperado, más adelante se mencionará el funcionamiento de las tres condiciones iniciales: *rango de radios*, *número de círculos* y *región de búsqueda*.
- * La tabla *Centros y radios* proporciona información acerca de los centros y radios de los círculos encontrados.

La metodología general para la detección de círculos, se divide en cuatro pasos:

1. Mediante funciones estándar disponibles en MATLAB, se carga la imagen de entrada, se convierte la imagen a escala de grises y posteriormente se binariza.
2. Se aplica el método correspondiente de acuerdo al programa que el usuario seleccionó (H3D, 2P, DG, 4P).
3. Se determinan los máximos locales en la matriz acumulador. Para eso, los tres programas 2P, DG, 4P, ofrecen la selección de dos métodos: por localidad o por umbral. Buscar *máximos locales por localidad* significa que el máximo es determinado en una región circular con un radio de 30 píxeles. Para determinar *máximos locales por umbral*, se analiza un área rectangular de la imagen de dimensión de $[0,1 \cdot \text{columnas}] \times [0,1 \cdot \text{filas}]$ píxeles, en la cual, se localizan las celdas con mayor número de votaciones, siempre y cuando este número sobrepasa cierto umbral preestablecido.
4. El programa le permite al usuario visualizar la matriz acumulador, la tabla con las coordenadas de centros y los radios de las circunferencias encontradas, el histograma de radios y los círculos encontrados graficados sobre la imagen original.

Los programas proporcionan, además, tres condiciones iniciales que especifican la búsqueda de circunferencias y de esta manera, se obtienen resultados más deseables. La primera condición inicial se refiere al *rango de radios*, con esto se delimitan las circunferencias a encontrar y reduce el tiempo de cálculo. El radio mínimo admitido es fijado en 10 píxeles, y el radio máximo considerado está dado por $r_{max} = \lfloor \max\{\text{filas}/2, \text{columnas}/2\} \rfloor$ píxeles. La segunda condición, el *número de círculos*, es especificado por el usuario, y define el número de círculos que se quieren obtener. Esta condición es muy útil cuando se conoce a priori la cantidad de circunferencias a detectar. Sin embargo, cuando los máximos locales no son muy evidentes en la matriz acumulador, el tiempo de cálculo para poder encontrar muchos círculos, puede aumentar considerablemente. La tercera condición inicial, llamada *región de*

búsqueda, sirve para que el usuario pueda especificar el radio de la localidad (cuando se busca por localidad) o las dimensiones del área cuadrada (cuando es búsqueda por umbral) para la búsqueda de los máximos locales.

6.2. Detección de circunferencias continuas

Para probar los programas de los cuatro métodos H3D, 2P, DG, 4P, se utilizaron las imágenes mostradas en la figura 6.2:

La imagen (a) de tamaño de 93x91 píxeles, presenta un solo círculo, pero también un poco de ruido.

La imagen (b) de 350x178 píxeles, contiene varios círculos de diferentes radios, con intersección entre ellos, y algunos no completamente contenidos en la imagen.

Lo interesante de la imagen (c) de 282x261 píxeles, es que no solamente contiene círculos de dos diferentes radios, sino que también, dos de ellos son concéntricos.

La imagen (d) de 640x320 píxeles, es la versión binarizada de una imagen gradiente de una fotografía. El reto para esta imagen es detectar los círculos presentes en las ruedas, aunque la imagen presenta muchas otras líneas y curvas.

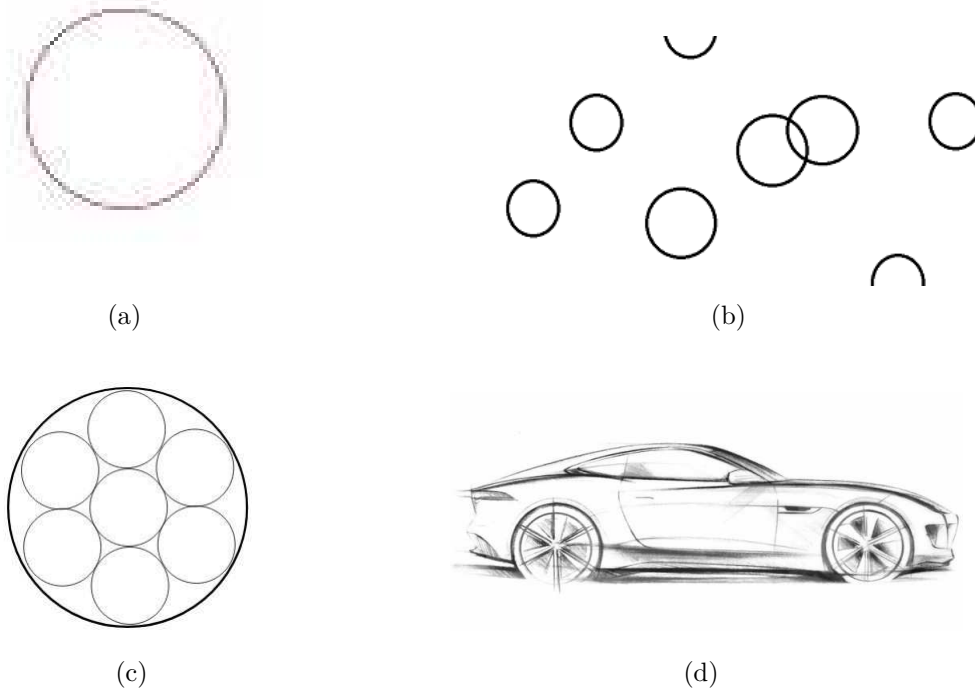


Figura 6.2: Imágenes de entrada, a ser analizadas por los cuatro métodos H3D, 2P, DG, 4P.

La imagen (a), al contener un único círculo, para cada uno de los cuatro métodos, su

matriz acumuladora exhibe claramente la acumulación de votos, vea figura 6.3. Las matrices acumuladoras para esta imagen son visualizadas en 3D, aunque corresponden a la representación gráfica de una matriz de dimensión dos de los parámetros h, k ; la base de la gráfica corresponde a dichos parámetros. Cada celda (h, k) tiene un valor entero correspondiente al número de votos, por lo que la representación gráfica del número de votos corresponde a la altura. Se utilizó un código de colores con el fin de diferenciar los máximos locales; los colores hacia el azul representan menor cantidad de votos, mientras que, colores hacia el rojo describen un mayor número de votos.

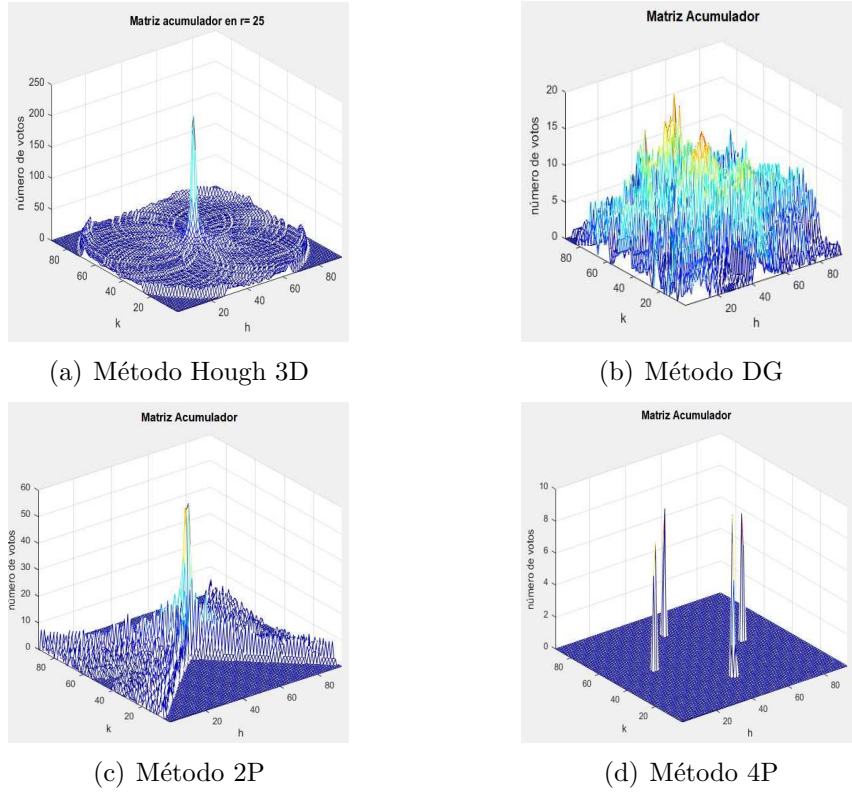


Figura 6.3: Resultados de las matrices acumuladoras correspondientes a los métodos H3D, DG, 2P y 4P, para la imagen (a) de la figura 6.2.

La figura 6.3(a) muestra el pico de acumulación de la matriz del método Hough 3D, que es dado por la intersección de las curvas generadas por cada punto candidato. Dado que es difícil representar gráficamente una matriz de dimensión tres, se representa una sección 2D de la matriz acumuladora, situada donde se encuentra el máximo global. Es decir, se graficó h, k y la sección donde el radio es igual a 25. Eso implica que cada punto candidato genera una circunferencia con $r = 25$. La mayor acumulación se encuentra en los valores de celdas $h = 46$ y $k = 47$. Esta matriz acumuladora es la que cuenta con el mayor número de votos, a comparación de los otros métodos. Esto es debido a que todos los puntos candidatos de

la imagen generan curvas de votos. También, este método es el que presenta mayor gasto computacional debido a la indexación en una matriz de dimensión tres.

La figura 6.3(b) muestra la matriz acumulador del método DG. Esta matriz es de dimensión dos, pues fue reducida por el uso de la dirección de gradiente. Este método es muy eficiente comparado con los otros tres. Sin embargo, el problema se suscita al encontrar el máximo, pues a pesar de ser una imagen simple, la matriz acumulador tiene muchos picos. En teoría, de acuerdo a [20], la dirección de gradiente de los puntos candidatos debiera apuntar hacia al centro. Entonces, todas las líneas generadas para las votaciones se encontrarían en un punto en común, que sería el máximo. Pero dada la imprecisión del valor de la dirección del gradiente, el cual fue obtenido al aplicar operadores gradiente, como Sobel que utiliza máscaras muy pequeñas, no se logra el resultado deseado. Eso ilustra el gran problema que presenta este método, en la incapacidad de obtener de manera exacta la dirección del gradiente en cada punto candidato.

En la figura 6.3(c) se notan claramente las líneas rectas de votos que fueron generadas en el espacio de Hough. El máximo global resulta ser más evidente que en la subfigura (b). Además, el gasto computacional es mucho menor que para H3D, por trabajar en una matriz acumulador de dimensión dos, y también, por considerar un menor número de puntos candidatos. Como se puede apreciar en esta matriz acumulador, las líneas de votos generadas corresponden a aquellas con una pendiente entre $(-1, 1)$, lo que de igual forma conduce a un menor almacenamiento de votos.

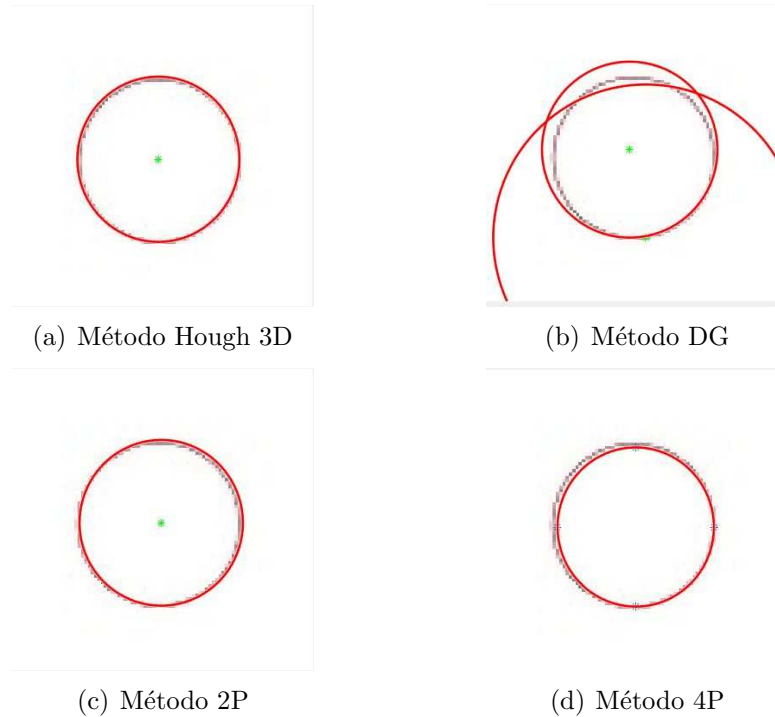
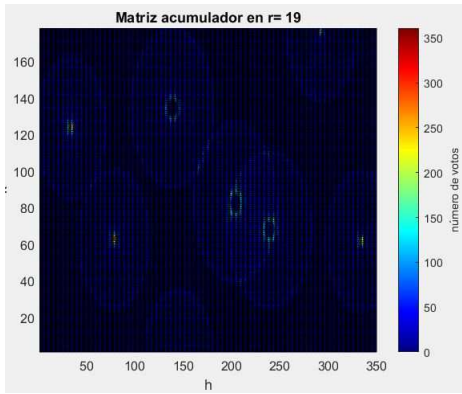


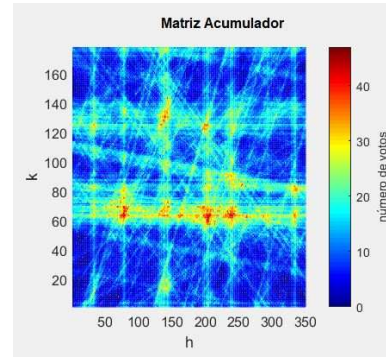
Figura 6.4: Círculos detectados en la imagen (a) de la Fig. 6.2.

La matriz acumulador con menor número de votos de estos cuatro métodos resulta ser la representada en la figura 6.3 (d). La razón de la menor cantidad de votos, es que únicamente se consideran grupos de cuatro puntos candidatos, que entre ellos tengan la misma dirección gradiente. Además, en esta gráfica fácilmente se reconocen los cuatro picos, los cuales representan los puntos finales de dos ejes de simetría que son perpendiculares entre ellos.

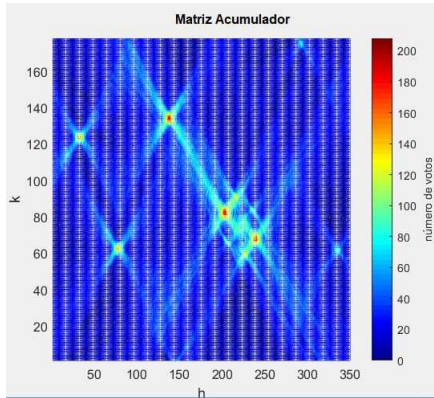
Los círculos detectados en la imagen original de la figura 6.2(a) son mostrados en la figura 6.4. Como se ve, los métodos H3D y 2P son los que ofrecen una mejor detección de circunferencias. Como resultado del método 4P, el círculo encontrado está un poco desfasado con respecto al de la imagen original. El método que presenta resultados con errores muy notorios, es el DG. En la matriz acumulador del método DG, se encontraron dos picos muy pronunciados. El pico con mayor número de votos no corresponde al círculo presente en la imagen; el segundo pico que tiene pocos votos menos que el máximo global, se ajusta mejor al círculo presente en la imagen original.



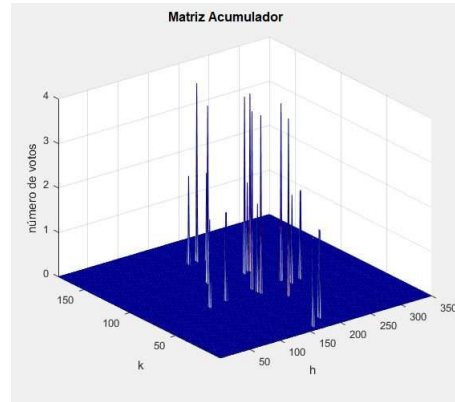
(a) Método Hough 3D



(b) Método DG



(c) Método 2P



(d) Método 4P

Figura 6.5: Resultados de las matrices acumuladoras correspondientes a los métodos H3D, DG, 2P y 4P para la imagen (b) de la figura 6.2.

La figura 6.5 muestra las matrices acumuladores de la imagen (b) de la figura 6.3. Para

el primer método, en (a), se presenta la matriz 2D obtenida de la matriz acumulador 3D con $r = 19$. En esta matriz existen tres máximos locales que corresponden a tres circunferencias con $r = 19$. Asimismo, se pueden apreciar formas similares a anillos, correspondientes a votos de otras circunferencias de radios diferentes a $r = 19$. Con respecto al segundo método, vea la subfigura (b), se muestra una matriz acumulador con muchos máximos locales, más que la cantidad de circunferencias presentes en la imagen. Este tipo de problemas promueven la detección de circunferencias falsas. La errónea acumulación de votos en celdas donde no hay centros de circunferencias, es resultado del impreciso valor de la dirección de gradiente. Por otro lado, el método 2P, vea subfigura (c), muestra la gran eficiencia del método, pues a simple vista se notan los máximos locales. Por último, el método 4P, donde su matriz acumulador está en la subfigura (c), se observan picos desordenados con lo que difícilmente se pueden determinar los cuatro puntos finales de los ejes de simetría del círculo buscado. Cabe recordar que este método, de igual manera que el DG, depende del valor de la dirección del gradiente, por lo cual, la detección de las circunferencias se ve limitada.

La siguiente figura 6.6, presenta la detección de las circunferencias por los métodos H3D y 2P. No se muestran aquí las circunferencias halladas por los métodos DG y 4P debido a sus pésimos resultados. Con el método Hough 3D, la detección de circunferencias fue muy exacta; las ocho circunferencias encontradas están bien ajustadas a la imagen original. El método 2P, a pesar de encontrar siete circunferencias muy bien emparejadas a los círculos de la imagen, detecta una circunferencia falsa. También, en este caso, el método no es capaz de localizar una media circunferencia. Para determinar los máximos locales, se utilizó la búsqueda de picos por umbral. Por tanto, la falta de votos hizo al programa omitir el máximo local de la media circunferencia, y en cambio, encontró un pico del centro de una circunferencia falsa. Si se utiliza el método de búsqueda de picos por región, se puede encontrar la media circunferencia, pero como se puede apreciar en la matriz acumulador asociada a este método, se encontrarían otras circunferencias falsas.

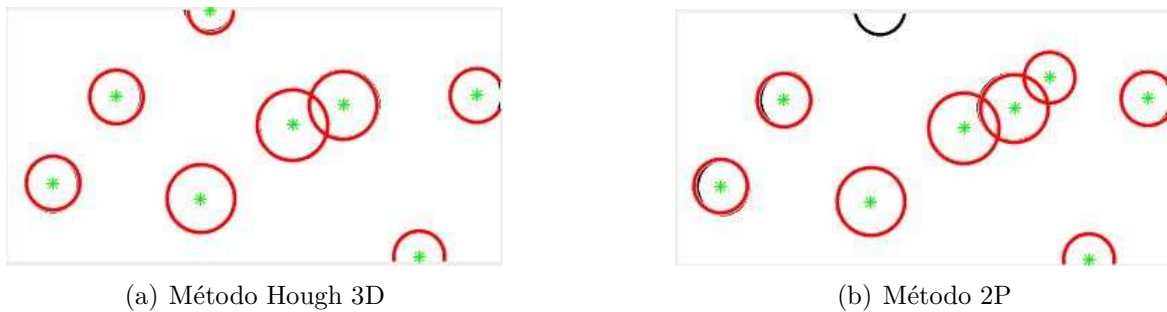


Figura 6.6: Círculos detectados en la imagen (b) de la figura 6.2.

En el siguiente ejemplo de la imagen (c) de la figura 6.2, se prueba el caso particular de circunferencias dentro de un círculo y dos circunferencias concéntricas. Para este ejemplo, se utilizaron únicamente los métodos H3D y 2P. Observando la figura 6.7), el método Hough 3D evidencia precisión: en la matriz acumulador, fácilmente se pueden determinar los máximos

locales. Las circunferencias encontradas son visualizadas en la figura 6.8(a). En cuanto al método 2P, se aprecia una peculiaridad cuando se trata con este tipo de circunferencias, la cual es la aparición de un máximo global muy evidente. Este máximo global se debe a que siempre se consideran las mismas posiciones de puntos candidatos de las circunferencias, y por ende, la generación de líneas de votos siempre tienen pendientes entre $(-1, 1)$ que se intersecan en un punto común y forman este máximo global. Esto resulta ser un problema para la localización de circunferencias concéntricas y resulta ser la gran desventaja de este método. El resultado de la aplicación de este método es mostrado en la figura 6.8(b).

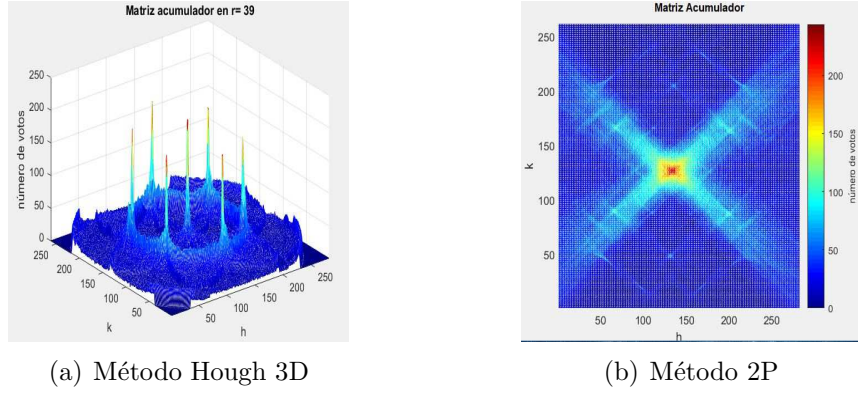


Figura 6.7: Resultados de las matrices acumuladoras correspondientes a los métodos H3D, DG, 2P y 4P para la imagen (c) de figura 6.2.

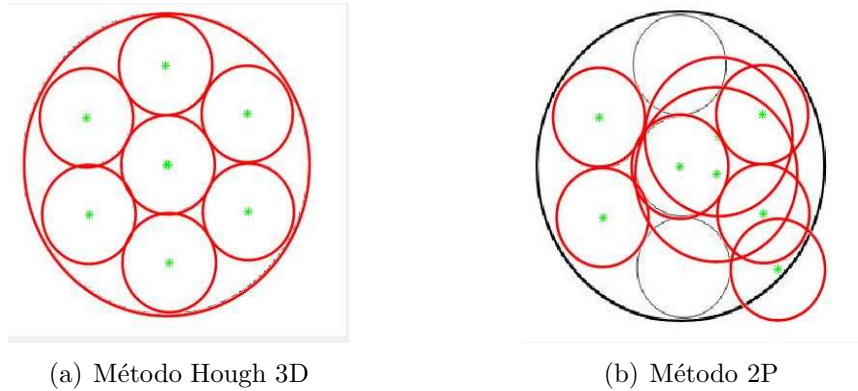


Figura 6.8: Resultados de la imagen (c).

Por último, consideremos la imagen (d) de la figura 6.2. Los resultados después de aplicar los cuatro métodos y con conocimientos a priori como el *rango de radios* y cierto *número de circunferencias*, fueron los siguientes: El método de Hough 3D (ver (a) en figura 6.9) proporciona resultados muy favorables, incluso en presencia de muchos puntos candidatos.

El método DG, a pesar de presentar limitaciones debido al valor de la dirección de gradiente, obtiene un resultado más o menos aceptable. El resultado del método 2P fue muy bueno e incluso mejor que el del método Hough 3D. No se consideró el resultado de aplicar el método 4P debido a que en la matriz acumulador difícilmente se detectaban los ocho máximos locales que forman las dos circunferencias en la imagen.

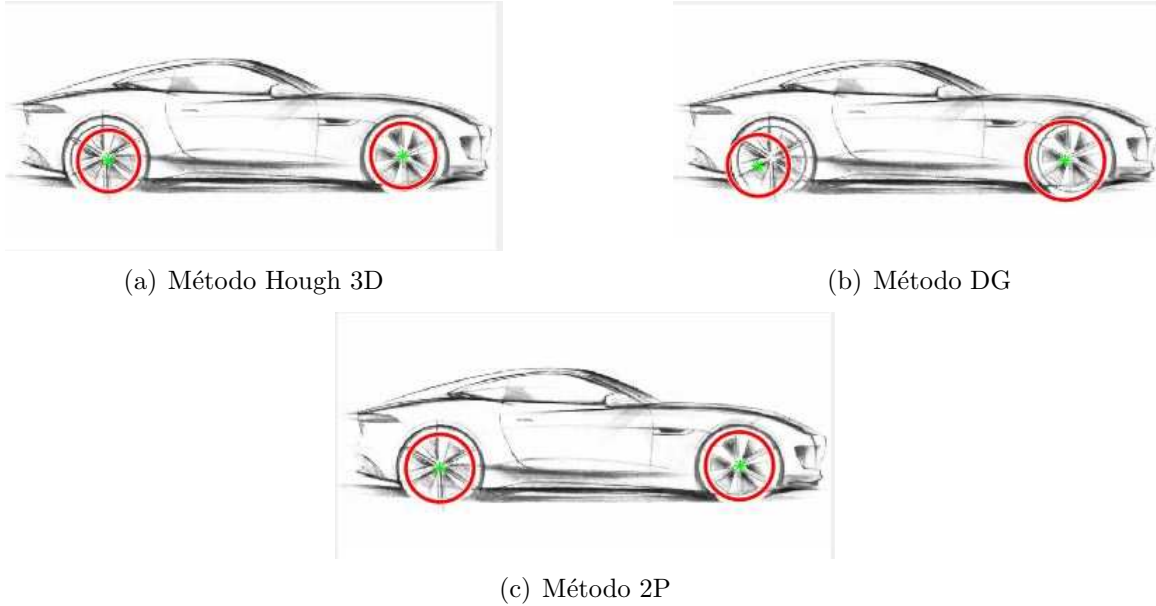


Figura 6.9: Resultados de círculos detectados en la imagen (d) de figura 6.2.

El método de dos puntos finales de una cuerda (2P) resulta ser muy preciso y la ejecución de cálculos es bastante rápida. Sin embargo, tiene una deficiencia para localizar círculos concéntricos. No obstante, este problema se puede solucionar utilizando una matriz acumulador 3D en vez de la 2D. Para utilizar una matriz 3D es necesario hacer una modificación al método 2P: El cambio consiste en determinar la distancia euclidiana de todos los puntos de la línea de votos, a cualquiera de los puntos finales de la cuerda. Este cálculo permite asociar los parámetros h, k con el radio. Con este cambio, denominaremos al método modificado por **2P3D**, haciendo referencia al uso de una matriz acumulador 3D.

Para realizar una comparación, se seleccionó el ejemplo (c) de la figura 6.2. La matriz acumulador obtenida y el resultado de las circunferencias encontradas pueden ser vistos en la figura 6.10. Como se puede apreciar en las subfiguras, la matriz acumulador tiene menor cantidad de votos a comparación de los métodos 2P y H3D. La localización de las circunferencias es correcta. Una característica importante de este método, comparado con los anteriores, es el tiempo de cálculo. Para este ejemplo, el tiempo de cómputo para 2P3D fue en promedio de 1.057630 segundos, en cambio, para H3D el tiempo fue de 19.785148 segundos. Como se nota claramente, una de las ventajas de 2P3D es el poco tiempo de ejecución para obtener las circunferencias correctas.

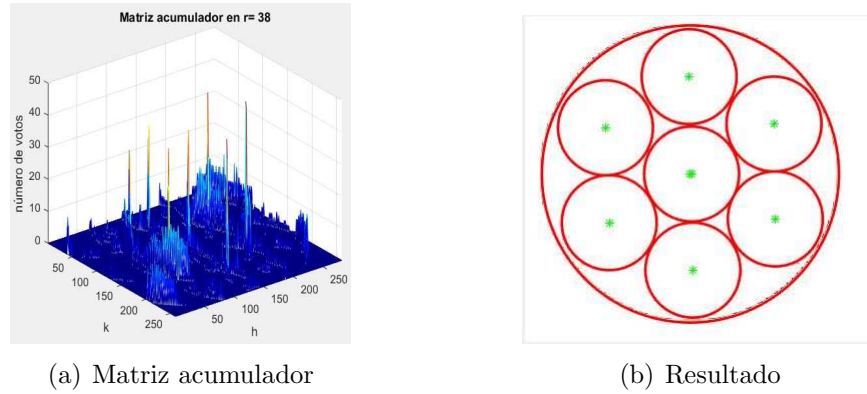


Figura 6.10: Matriz acumulador y resultado de circunferencias encontradas con el método 2P3D.

6.3. Comparación con otras implementaciones del método de Hough en MATLAB

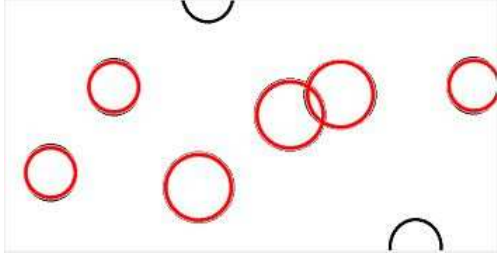
En la literatura se pueden encontrar muchos métodos para la detección de circunferencias. En cuanto a programas para el entorno MATLAB, se puede encontrar la función predeterminada que viene incluida en la biblioteca de procesamiento de imágenes denominada *imfindcircles*, la cual se menciona brevemente en la sección 3.3.6. Por otra parte, en el internet, esencialmente en los foros de *mathworks*, se mencionan aplicaciones y funciones; la mayoría de las cuales, se centra en la utilización del método Hough 3D debido a su alta precisión.

Nos parece importante hacer mención de tres funciones escritas por David Young [41]: *circle_hough*, *circle_houghpeaks* y *circlepoints*. Se destacan su alta eficiencia y su reducido tiempo de cálculo en comparación con otros. Los aspectos a resaltar de estas funciones son:

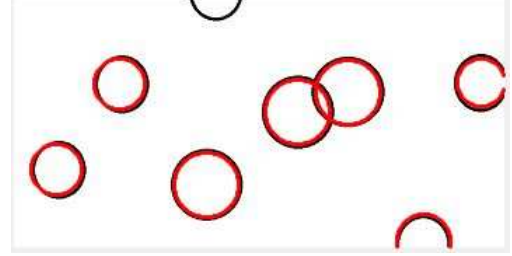
- 1) Aumentan la resolución de la matriz acumulador para obtener resultados más precisos.
- 2) Se incluye una opción denominada *normalizar*, donde todos los elementos de cada capa de la matriz acumulador 3D, es decir, los valores $Acc(:, :, k)$, son multiplicados por $1/R(k)$, donde R es el radio. Esto es útil porque los círculos más grandes necesitan más votos para confirmar su evidencia en la imagen, aproximadamente en proporción a su radio.
- 3) Se utiliza un filtro gaussiano para la matriz acumulador, eso permite suavizar los picos y posteriormente analizar los máximos locales.
- 4) Para la generación de circunferencias en el espacio de Hough y para la acumulación de votos, se crean curvas irreducibles y en un sólo cuadrante, correspondientes a un sólo octante, posteriormente se combinan los signos de las coordenadas para completar la circunferencia.

Para comparar la función estándar de MATLAB, las funciones de David Young y el método H3D, se utilizaron las imágenes de los ejemplos (b) y (c) de la figura 6.2. En el ejemplo (b) se delimitó el rango de radios a $[16, 25]$. Se pudo comprobar que la función *imfindcircles* es capaz de detectar seis circunferencias, además el tiempo de calculo fue de 1.601719 segundos.

Para los algoritmos de David Young, se estableció una región de búsqueda de 15 píxeles, también, se determinó el número de circunferencias a encontrar. Sin embargo, las funciones de Young sólo pudieron encontrar siete circunferencias y en un tiempo de 0.377178 segundos. Por último, el método H3D determinó las ocho circunferencias (ver figura 6.6), especificando un número de circunferencias, y utilizó un tiempo de cálculo de 1.954293 segundos.



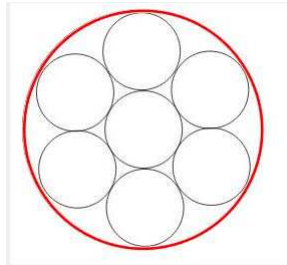
(a) Resultado de la función *imfindcircles*.



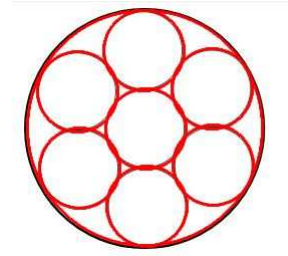
(b) Resultado de las funciones de David Young.

Figura 6.11: Comparación de resultados del ejemplo (b) con otros programas en MATLAB.

Experimentando con el ejemplo (c), se estableció un rango de radios en el intervalo [38, 119]. La función *imfindcircles* no pudo detectar todas las circunferencias debido a la condición de $3r_{min} > r_{max}$. Por el contrario, los códigos escritos por Young, fueron muy satisfactorios encontrando las ocho circunferencias, ocupando el tiempo de 2.637626 segundos, lo cual fue un tiempo incomparable con relación al método H3D que lo realizó en 19.785148 segundos.



(a) Resultado de la función *imfindcircles*.



(b) Resultado de las funciones de David Young.

Figura 6.12: Comparación de resultados del ejemplo (c) con otros programas en MATLAB.

Como se puede notar el tiempo de cálculo del ejemplo (c) aplicando el método 2P3D, fue menor que el de David Young. El tiempo de cómputo de cada método depende de la cantidad de puntos candidatos, por esta razón se ha decidido incluir un ejemplo más para notar la diferencia entre tiempos de los algoritmos de David Young y 2P3D: se consideró la siguiente imagen de la figura 6.13).

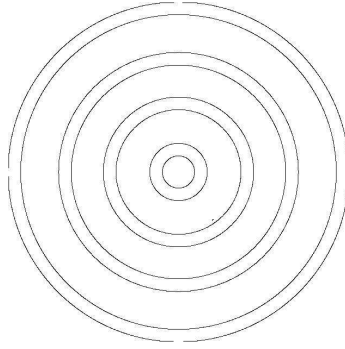
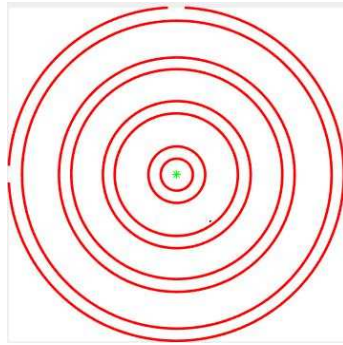
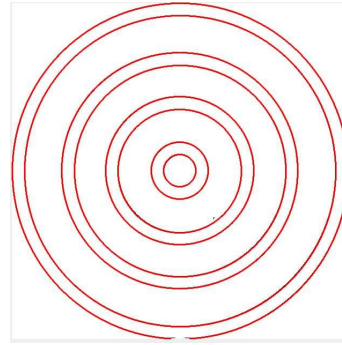


Figura 6.13: Imagen con circunferencias concéntricas.

Los resultados de este último ejemplo son mostrados en la siguiente figura 6.14. La búsqueda se realizó en un rango de radios del intervalo $[33, 313]$ y especificando ocho circunferencias a encontrar. Ambos métodos son capaces de hallar las ocho circunferencias. Sin embargo, el tiempo de cómputo varía: el método 2P3D tardó 6.752501 segundos, en cambio, los algoritmos de Young necesitaron 93.002261 segundos.



(a) Resultado del método 2P3D.



(b) Resultado de las funciones de David Young.

Figura 6.14: Resultados de detección de círculos concéntricos.

A partir de esta comparación con métodos encontrados en internet y la función incluida en la librería de procesamiento de imágenes de MATLAB, se puede concluir que *imfindcircles* se encuentra muy delimitado por el rango de búsqueda de circunferencias, además de tener problemas para detectar circunferencias parciales. En cuanto a los códigos desarrollados por David Young, la función utilizada para determinar los máximos locales está optimizada y permite obtener tiempos de cálculo muy buenos, aunque fue incapaz de detectar una media circunferencia; en general, ocupa menos tiempo de cálculo el método 2P3D.

6.4. Imágenes artificiales de nubes de puntos

En este apartado se consideran ejemplos de imágenes con nubes de puntos, pues resulta interesante cuando los puntos candidatos no se encuentran conectados y existe mucho ruido en la imagen. Primero, se analiza la imagen de la figura 6.15(a) de 300x300 píxeles, valorando un rango de radios del intervalo $[70, 150]$. Se aplicaron los métodos H3D (subfigura (b)), 2P (subfigura (c)) y 2P3D (subfigura(d)). Se aprecia que el método 2P3D es bastante afectado por el ruido y su resultado es equívoco. En cambio, los otros dos métodos presentan un resultado aceptable.

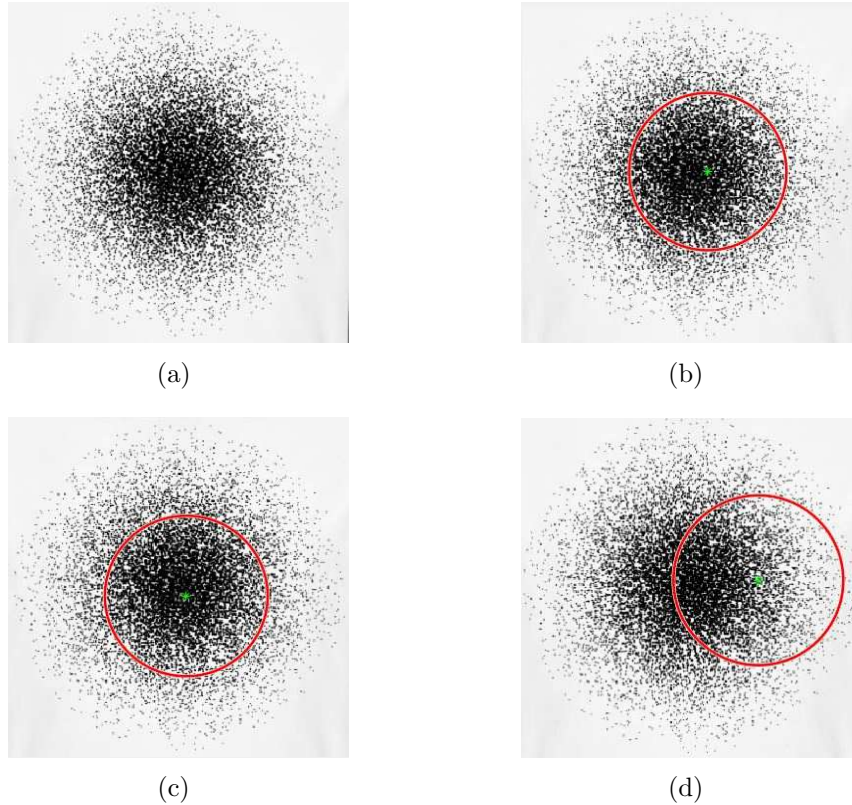


Figura 6.15: Nubes de puntos.

Después se consideraron las siguientes imágenes a analizar, donde claramente hay más de una circunferencia, vea la figura 6.16. La imagen (a) es de 716x426 píxeles; las imágenes (b) y (c) ambas son de 512x512 píxeles; y (d) de 450x392 píxeles.

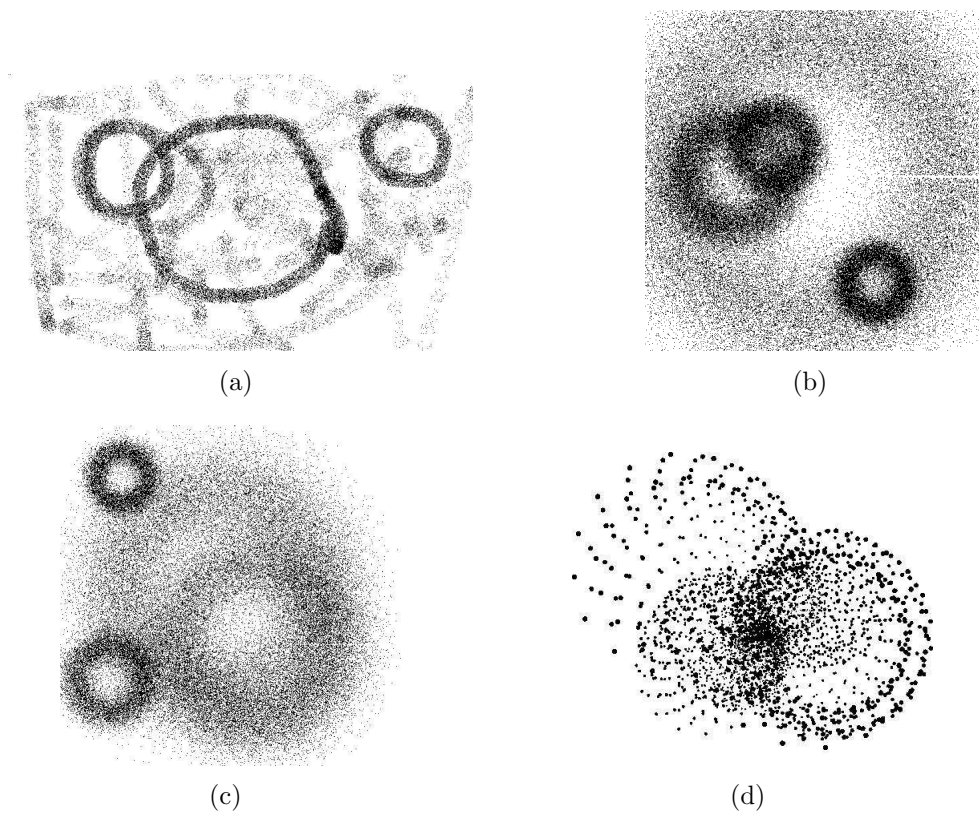


Figura 6.16: Imágenes con nubes de puntos.

Analizando la imagen (a) con los métodos H3D y 2P, se detectan los círculos mostrados como superpuestos en la imagen real, en la figura 6.17. Como se aprecia, el método 2P resulta ser incapaz de obtener las circunferencias correctas; en cambio, el método H3D localiza sin problema las circunferencias.

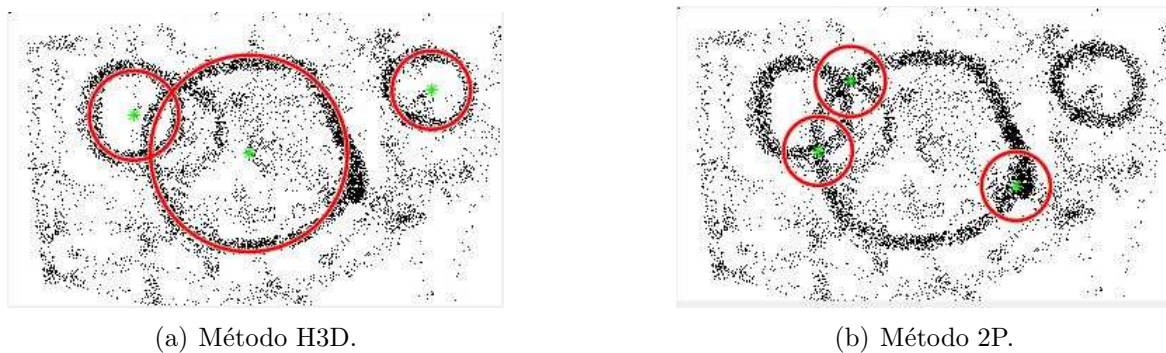


Figura 6.17: Ejemplo (a).

Dado que H3D es el único de los cuatro métodos que ha presentado buenos resultados para este tipo de imágenes, de aquí en adelante será el único método que se utilizará para el análisis de nubes de puntos. Para la imagen (b) de figura 6.16, vea figura 6.18. Se consideró en primera instancia el análisis en un rango de radios de $r \in [30, 200]$, y con un valor umbral de 100. El resultado de este análisis se muestra en la subfigura (a), y se necesitó un tiempo de 9359.2982322 segundos. Por otra parte, si se considera buscar los cuatro máximos locales dentro del mismo rango de radios, el resultado es presentado en la subfigura (b), el tiempo de cálculo fue entonces de 668.029473 segundos.

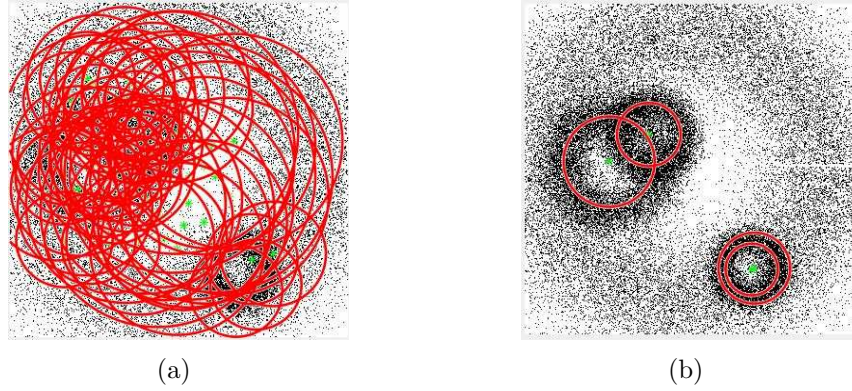


Figura 6.18: Ejemplo (b) con resultados no idóneos.

Como se notó en las imágenes anteriores, el tiempo de cálculo y la complejidad para detectar los picos correctos en la matriz acumulador, presentan retos difíciles. Por eso, se ha optado por detectar las circunferencias en dos pasos, como se muestra en la figura 6.19. En el primer paso (a), se consideró el rango de radios de $r \in [40, 70]$; y en el segundo paso (b), se tomó el rango $r \in [170, 200]$. De este modo se obtienen circunferencias aceptables y en un tiempo total de 284.778764 segundos. Por lo que, el análisis en diferentes pasos resulta más eficaz.

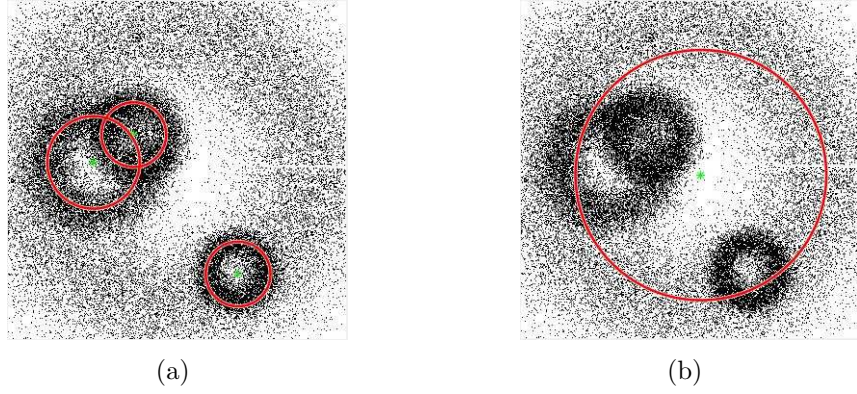


Figura 6.19: Ejemplo (b).

Para el ejemplo (c), tomando en cuenta que se conocen aproximadamente las dimensiones de las circunferencias, de igual forma se ha dividido el método en dos pasos y los resultados son presentados en la figura 6.20. En el primer paso (a) se tomó el rango de radios $r \in [30, 50]$, para el segundo, $r \in [100, 200]$. Se ocupó un tiempo de cómputo total de 413.984 segundos.

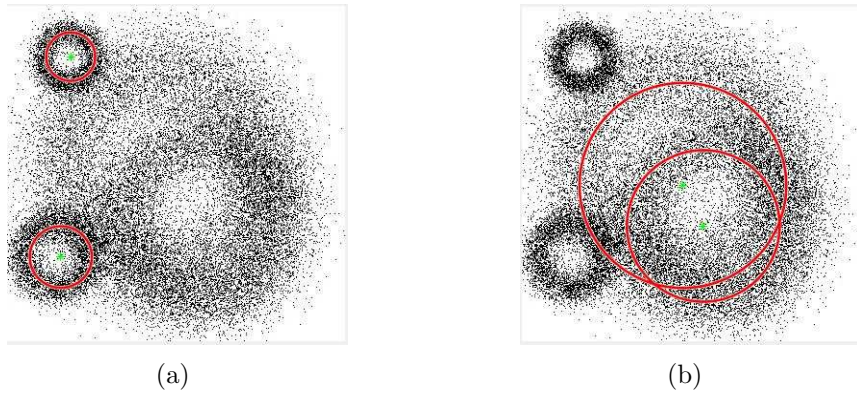


Figura 6.20: Ejemplo (c).

Por último, para el ejemplo (d) (vea resultados en la figura 6.21), en el primer paso (a) se intentaron calcular las circunferencias más pequeñas en un rango de radios $r \in [80, 100]$. Después, en (b) para $r \in [101, 150]$ y luego en (c) para $r \in [130, 200]$. El tiempo de cómputo total fue de 125.286633 segundos. El resultado de la subimagen (d) se analizó en un rango de $r \in [70, 200]$, y se determinaron las seis mejores circunferencias, es decir, aquellas que tuvieran mayor cantidad de votos. El resultado refleja una tendencia hacia un radio similar, esto muestra que cuando existen circunferencias con diferentes radios en la imagen es mejor analizarla parte por parte, pues así no se pierde la diversidad de radios. Comúnmente los radios más pequeños son los que más se repiten en estos casos.

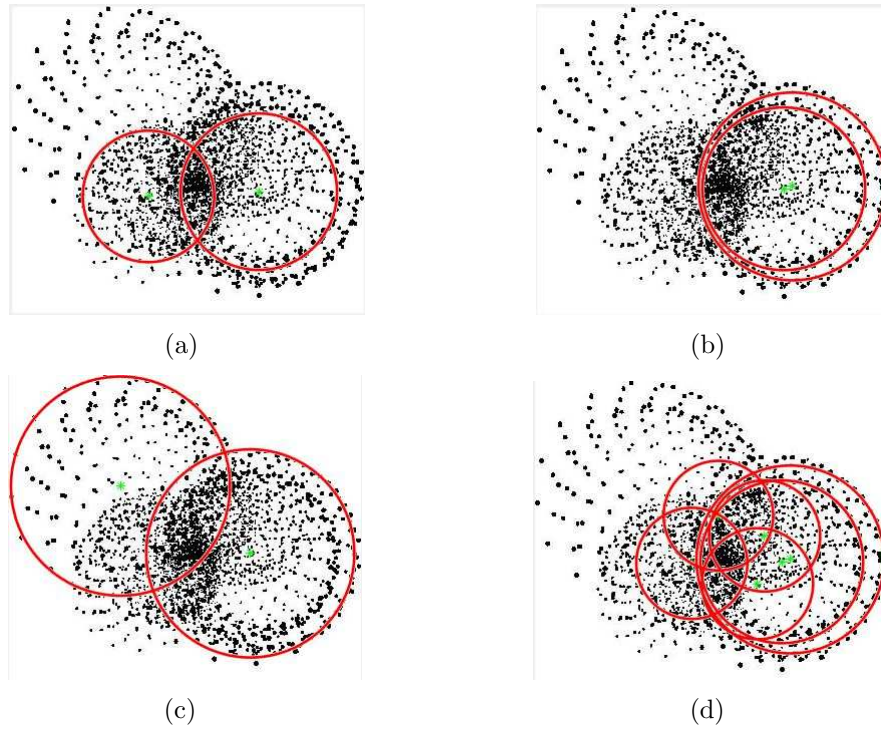


Figura 6.21: Ejemplo (d).

Las nubes de puntos contienen muchos puntos candidatos y ruido, por lo que se espera que el tiempo de ejecución sea mayor a diferencia de circunferencias continuas. Ante esta situación, en algunos casos se puede considerar realizar un preprocesamiento que permita resaltar las circunferencias existentes en la imagen, así también, reduciendo el número de votantes.

Para el ejemplo (a) de la figura 6.16 se utilizaron operaciones morfológicas aplicadas a la imagen binaria, en el siguiente orden:

- 1) se dilató la imagen con un elemento estructurante circular de radio 2;
- 2) se erosionó la imagen con un elemento estructurante circular de radio 8;
- 3) se utilizó una operación de adelgazamiento (función *thin* de esqueletización de MATLAB), la cual convierte objetos a líneas sin romper conexidad; y por último,
- 4) se aplicó una operación de MATLAB denominada *areaopen*. Esta función elimina objetos de un determinado número de píxeles.

Para la imagen (b) se realizó un preprocesamiento similar, pero obviamente, las operaciones que permitan un mejor resultado dependen de cada imagen. En este caso, los resultados del preprocesamiento y resultados de aplicar del método H3D son mostrados en la siguiente figura 6.22. Aquí se muestra claramente que el preprocesamiento de una imagen no garantiza obtener mejores resultados para la obtención de circunferencias, al menos para nubes de puntos. Por lo que es más recomendable analizar las nubes de puntos como tales.

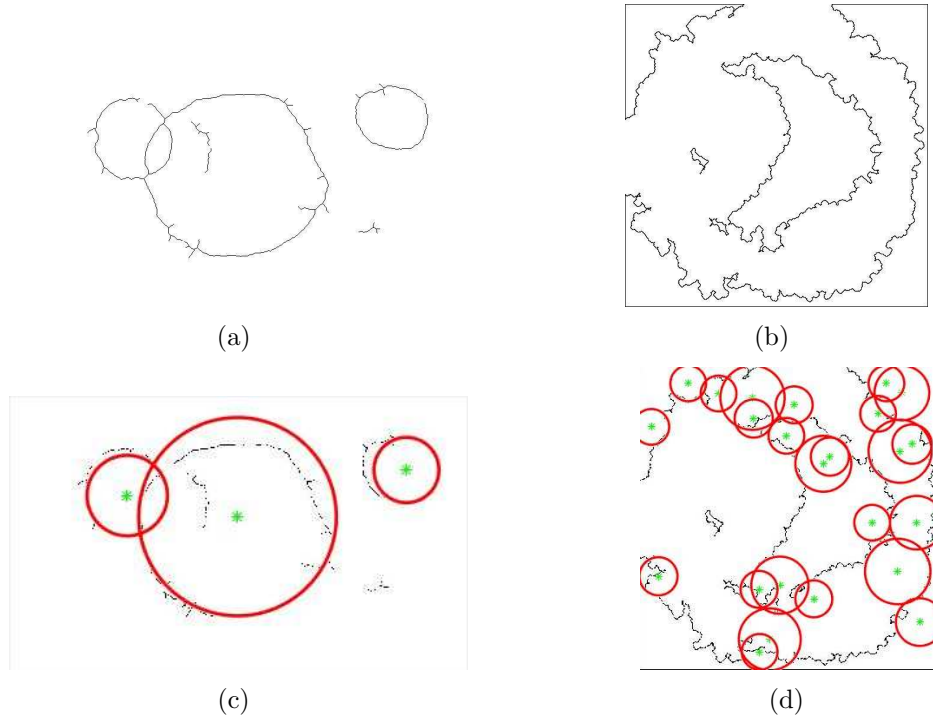


Figura 6.22: Imágenes preprocesadas y resultados.

6.5. Tratamiento de una imagen astronómica

La figura 6.23 fue tomada del artículo [4], y muestra una simulación de la imagen de una porción del universo donde cada punto obscuro representa una galaxia. Podemos observar que estos puntos no pertenecen a estructuras conexas sino más bien, forman una nube. Sin embargo, es evidente que muchos puntos parecen estar sobre circunferencias, sobre las cuales, la densidad de la nube de puntos es considerablemente más alta que en el resto de la imagen. Eso refleja una propiedad del proceso de formación de estructuras durante la historia del universo que en la física teórica se ha determinado y predicho, dentro de algunos modelos de la materia oscura. Según estas teorías, después de 350,000 años a partir de la Gran Explosión (*Bing Bang*), las galaxias formadas se agrupan sobre esferas de un radio de aproximadamente 150 millones de parsecs (cada parsec equivale a 3.26 años luz), las cuales quedan congeladas en su tamaño. Eso significa que, mientras que la historia del universo sigue en su proceso de inflación, estas esferas ya no crecen, aunque se alejan una de la otra. En imágenes astronómicas, estas esferas comúnmente aparecen como circunferencias, como en la figura 6.23. Para los astrónomos es de gran interés determinar los centros y radios de estas circunferencias. Cabe notar que actualmente todavía no se cuenta con imágenes reales astronómicas de suficiente resolución para facilitar la determinación de dichos círculos o esferas a través de técnicas del procesamiento de imágenes. El censo actual de galaxias de nuestro

universo abarca aproximadamente 2 millones de galaxias. Sin embargo, el telescopio DESI (Dark Energy Spectroscopy Instrument) que se encuentra en construcción en los Estados Unidos de América, entrará en operación en 2019. Con este telescopio, se espera aumentar el registro de datos del universo y llegar a un censo de entre 20 y 30 millones de galaxias. Investigadores mexicanos del CINVESTAV-IPN, de la UNAM, de la Universidad de Guanajuato y del Instituto Nacional de Investigaciones Nucleares, participan activamente en el proyecto DESI, por lo cual, tendrán acceso a estos nuevos datos, y entonces, la importancia del análisis de imágenes astronómicas vivirá un aumento sin precedentes.

Por eso, en esta tesis, se hace un primer intento de tratar una imagen del tipo de imágenes que se espera obtener en el futuro.

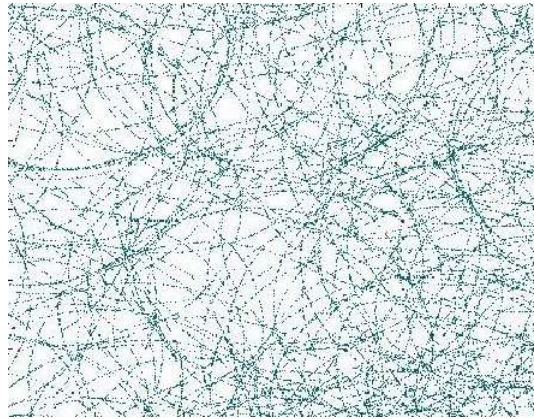


Figura 6.23: Imagen simulada astronómica (vea texto).

Dada la complejidad de la imagen se decidió utilizar, el método de Hough 3D (H3D) que provee los resultados más precisos. Además, se ha modificado el algoritmo para obtener mejores resultados, como se describe a continuación:

1. Se carga la imagen a la aplicación, se convierte la imagen a escala de grises y en seguida se binariza.
2. Se especifica un rango de radios. Posteriormente, se aplica el método H3D.
3. Se localizan los máximos locales por umbral, y así, se determinan los parámetros de *posibles circunferencias* en la imagen.
4. Se crean máscaras de las circunferencias determinadas en el paso anterior. Las máscaras son usadas en la imagen original para extraer y contar los puntos de la imagen original que realmente pertenecen a cada una de estas posibles circunferencias.
5. Para cada posible circunferencia, teniendo sus parámetros especificados, se determina el número de píxeles de su versión digitalizada. Este número (=100 por ciento) es comparado con la cantidad (menor) de píxeles extraídos de la imagen original que pertenecen a esta circunferencia. El porcentaje obtenido se usa para decidir si la posible circunferencia es considerada como idónea, en particular, cuando este porcentaje sobrepasa un umbral preestablecido.

6. Por último, se visualizan las circunferencias formadas por los puntos de la imagen original que resultaron idóneas.

En los experimentos, se consideraron tres umbrales para los porcentajes determinados en el paso 5 de arriba: 75 %, 60 % y 50 % de la cantidad ideal de píxeles de cada circunferencia digital. Para el caso de al menos 75 % de píxeles de la imagen original verdaderamente presentes en la circunferencia digital, se probaron varios intervalos como rangos de radios: $[220, 250]$, $[200, 219]$, \dots , $[20, 39]$, $[10, 19]$. Como se puede observar en la figura 6.24, solo se obtuvieron circunferencias con al menos 75 % de los puntos en el rango de radios $[10, 39]$.



Figura 6.24: Resultados cuando se exige un 75 % de puntos sobre las circunferencias.

En cuanto al caso de 60 %, se probó con los mismos rangos que el anterior. Sin embargo, por razones obvias, se obtuvieron más circunferencias que cumplían la condición, los resultados se muestran en la figura 6.25.

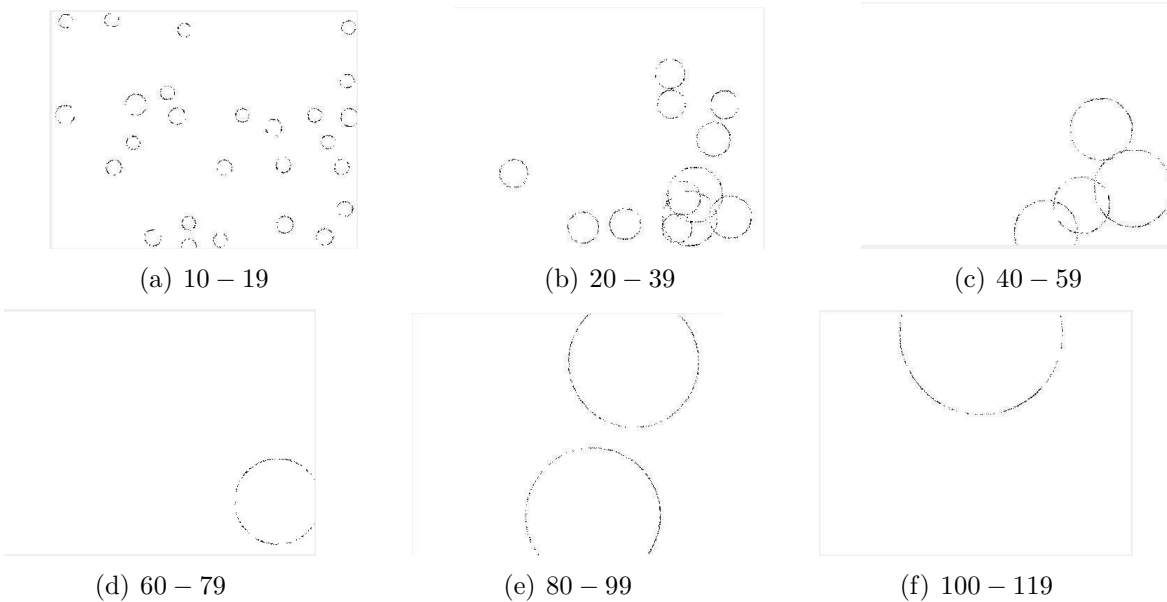


Figura 6.25: Resultados cuando se considera un 60 % de puntos sobre las circunferencias.

Utilizando el umbral de 50 %, y considerando los mismos rangos de radios, el número de circunferencias encontradas fue aún mayor, como se ve en la figura 6.26. A menor umbral para el porcentaje logrado, claro que se pueden encontrar mayor cantidad de circunferencias, ya que se necesitan menos puntos votantes para aceptar la existencia de tales circunferencias. Entonces, el punto importante es determinar qué porcentaje de puntos es suficiente para considerar confirmada la presencia de una posible circunferencia en la imagen.

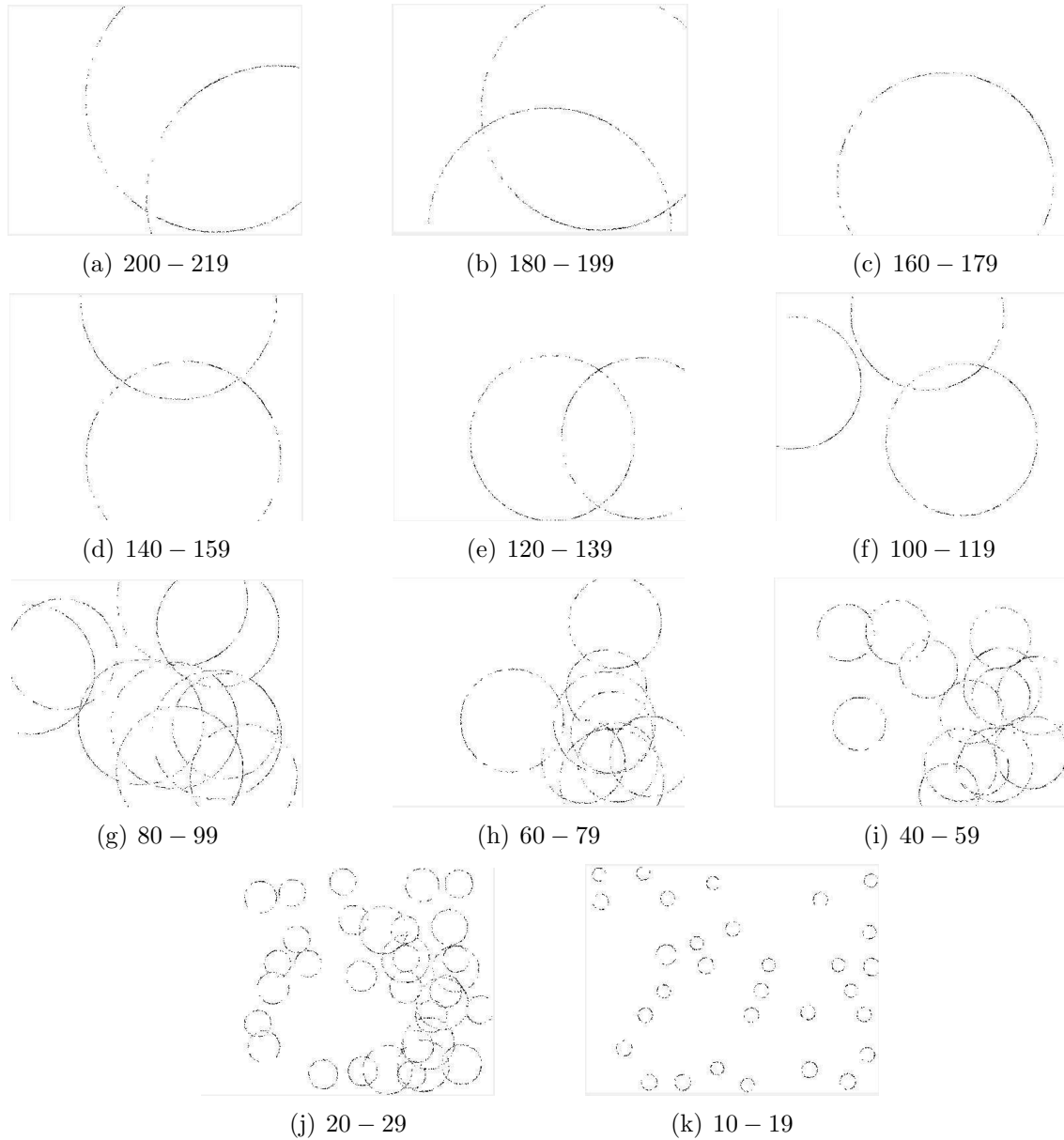


Figura 6.26: Resultados cuando se considera un 50 % de puntos sobre las circunferencias.

6.6. Programa para ilustrar la Versión 1 del Algoritmo de Bhowmick

Se elaboró un programa en MATLAB para ilustrar la eficiencia de la Versión 1 del Algoritmo de Bhowmick, descrito en la sección 4.2 y propuesto en [6]. Este programa tiene como finalidad determinar si una curva digital tiene circularidad digital completa o parcial. En el caso de circularidad parcial, el programa sigue analizando el resto de la curva dada, para lograr, en caso necesario sucesivamente, la segmentación de la curva dada en todos sus arcos circulares.

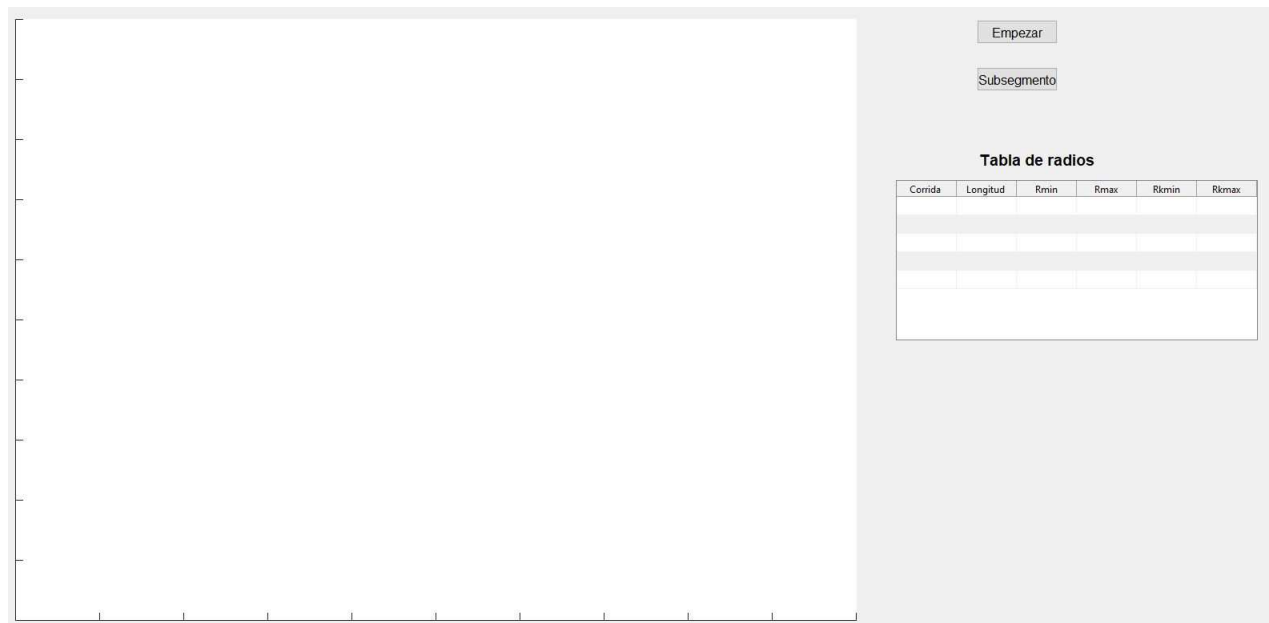


Figura 6.27: Interfaz gráfica para la Version 1 del Algoritmo de Bhowmick.

El programa cuenta con una interfaz gráfica como se muestra en la figura 6.27. Está compuesta por un recuadro, dos botones y una tabla. El recuadro visualiza la curva digital a analizar y las circunferencias de distintos radios que pasan por esta curva y son determinadas durante el algoritmo. El botón denominado *Empezar*, cuando es presionado, se abre una ventana emergente donde solicita al usuario ingresar la curva digital en formato de una secuencia de números naturales, correspondientes al código de longitudes de corridas. Cabe notar que el programa supone que las corridas son horizontales. El otro botón llamado *sub-segmento*, debe ser presionado en caso de que existan segmentos en la curva digital dada, esto permitirá visualizar cada segmento por separado. Y por último, la *Tabla de radios*, la cual está compuesta por seis columnas, contiene los siguientes datos:

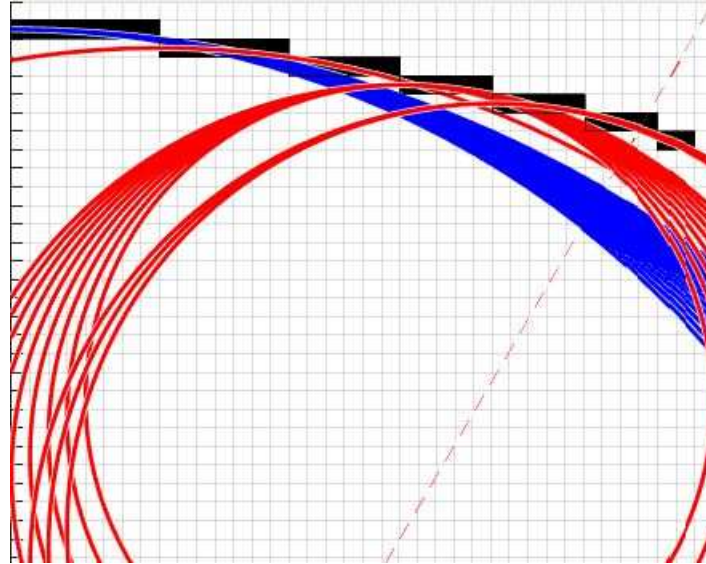
* *corrida* corresponde al número actual de corrida, empezando desde el cero el cual coincide con la corrida superior que inicia con su punto extremo izquierdo, considerando siempre

solamente el primer octante de un círculo digital;

* *longitud* determina el tamaño de la corrida;

* *Rmin* y *Rmax* proporcionan el rango de los radios de las circunferencias de acuerdo a su longitud y número de corrida; y

* *Rkmin* y *Rkmax* determinan el rango de los radios de las circunferencias posibles entre la corrida actual y la anterior.



(a)

Corrida	Longitud	Rmin	Rmax	Rkmin	Rkmax
0	8	50	64	50	64
1	7	67	75	37	49
2	6	49	57	49	49
3	5	60	66	17	25
4	5	28	34	17	25
5	4	23	27	23	25
6	2	22	25	23	25

(b)

Figura 6.28: Segmento de curva digital dado por su código de longitudes de corridas $S = \{8765542\}$. (a) Representación de la curva digital S y de los círculos digitales correspondientes a las partes circulares de S . (b) Tabla de radios asociados a las corridas.

Consideremos como ejemplo la curva digital dada por su código de longitudes de corridas horizontales $S = \{8765542\}$. Al ser ingresado S al programa (ver figura 6.28), la curva digital y las circunferencias que pasan por esta, son visualizadas, como se muestra en (a). Las circunferencias de color azul representan hasta donde se mantiene circularidad completa en esta curva, las circunferencias de color rojo pertenecen a los segmentos existentes. En la subfigura (b) se muestran los radios asociados a las corridas y longitudes. En este caso se

puede notar que la curva S no presenta circularidad digital completa, por lo que el programa la ha dividido en diferentes segmentos, cada uno, de los cuales, presenta circularidad digital.

Los segmentos en los que se dividió la curva digital S , se muestran en la figura 6.29. La subfigura (a) corresponde a la primera corrida de longitud $s_1 = \{8\}$, la cual por sí sola forma un segmento y pertenece a varios círculos digitales con radios en el intervalo $[50, 64]$. El segmento $s_2 = \{76\}$, vea subfigura (b), tiene circularidad digital para dos corridas consecutivas para radios $r \in [37, 49]$. Cabe recordar que cada segmento reinicia el número de corrida, es decir, ahora tenemos $s_0 = 7$ y $s_1 = 6$. En la subfigura (c) observamos al segmento $s_3 = \{5\}$, el cual tiene circularidad digital para radios $r \in [17, 25]$. Por último, el segmento $s_4 = \{542\}$ tiene circularidad digital para radios $r \in [23, 25]$. Como resultado, la curva completa S no posee circularidad digital, sino solamente parcial y hasta la primera corrida. En cambio, cada segmento tiene circularidad digital.

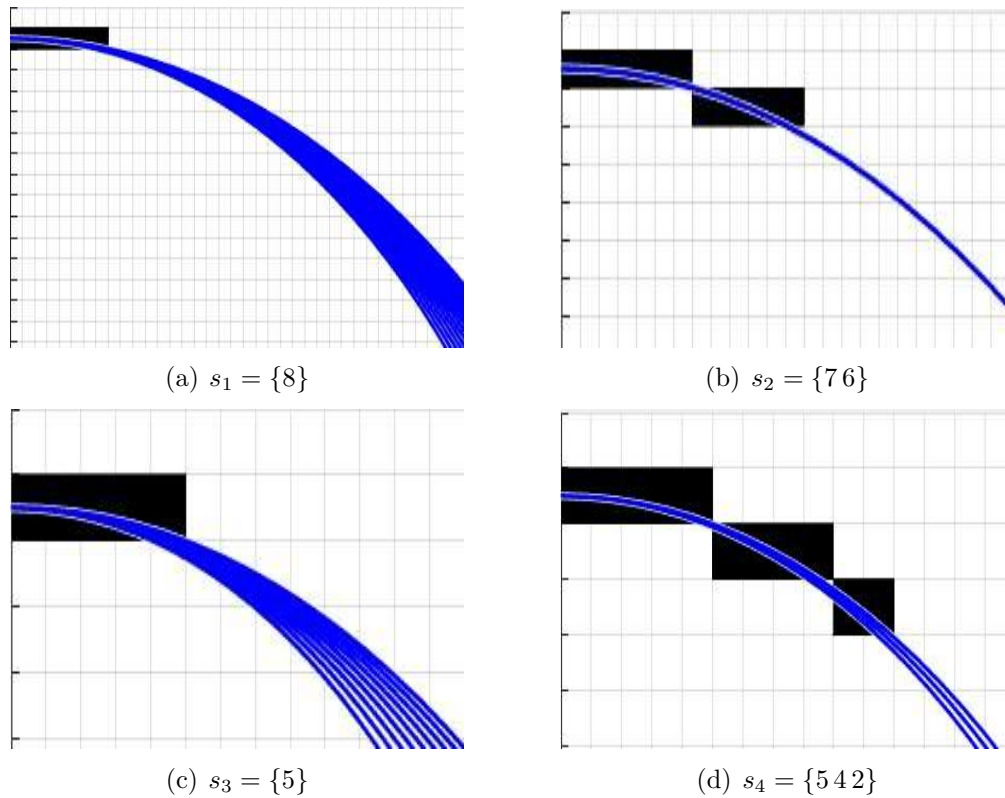


Figura 6.29: Segmentos de la curva de $S = \{8765542\}$, correspondientes a sus arcos circulares.

Conclusiones

En esta tesis fue reportado el uso de la transformada de Hough y otros métodos que utilizan la acumulación de evidencia para la detección de circunferencias y elipses en imágenes digitales. Además, se describieron varios métodos que permiten reducir la dimensión de la matriz acumulador o permiten descomponer el espacio de parámetros. Esto se hace con el afán de disminuir el gasto computacional producido por el método general de Hough. Para lograr eso, estos métodos aprovechan la imagen gradiente y propiedades geométricas.

Asimismo, se describió el concepto de circularidad digital y la utilidad que puede tener para la detección de circunferencias en imágenes. Se presentaron las ideas básicas de un algoritmo que de manera altamente eficiente, dada una curva digital, puede determinar si la curva es un círculo digital, y en caso de que no, segmentarla en sus segmentos circulares. Las bases teóricas de este algoritmo están relacionadas con la teoría de números.

También, se trataron brevemente las ideas básicas de métodos de ajuste de curvas basado en la optimización matemática para la detección de circunferencias en conjuntos desorganizados de puntos.

Se programaron en MATLAB cuatro de los métodos de acumulación de evidencia, el *método general Hough 3D (H3D)*, el *método para la detección de círculos basado en valor y dirección de gradiente (DG)*, el *método de dos puntos finales de una cuerda (2P)*, y el *método geométrico con dirección de gradiente (4P)*.

Se realizaron experimentos con imágenes de diversos tipos, por ejemplo, imágenes que contienen circunferencias de estilo de dibujo, otras que tenían que ser primero filtradas para detectar puntos de bordes, y luego binarizadas. Especialmente, se experimentó con nubes bidimensionales de puntos. Como resultado de los estudios comparativos, se concluye que el método general de Hough (H3D) ofrece los mejores resultados, puesto que mostró ser robusto al ruido y tener alta exactitud para encontrar circunferencias, aunque sean ocluidas o incompletas. Sin embargo, una de sus desventajas es el gasto computacional, ya que el tiempo de ejecución suele ser extenso en comparación a los otros métodos, además, utiliza mucha más memoria que los demás. El método 2P, ofrece menor fatiga computacional, máximos locales muy bien definidos en la matriz acumulador, pero es sensible al ruido. Cuando hay circunferencias pequeñas ocluidas o incompletas, debido al bajo número de votantes, el método 2P omite estas circunferencias, y por último, es incapaz de detectar circunferencias concéntricas. Los métodos DG y 4P, obtuvieron los peores resultados en esta serie de experimentos. Una de las razones es el uso de la dirección de gradiente la cual es calculada a partir de la imagen digital mediante filtros lineales con máscaras de dimensiones muy reducidas, por ejemplo

los filtros de Sobel o Prewitt que usan máscaras de 3x3 píxeles. Dada la inexactitud de la dirección calculada, los resultados de detección de círculos fueron pésimos. Por eso mismo, estos métodos no fueron valorados para las imágenes con nubes de puntos.

Se trabajó con una modificación del método 2P, usando una matriz acumulador 3D. El método modificado 2P3D resultó ser el más rápido de todos, presentando una alta precisión cuando no hay ruido y se trabaja con circunferencias continuas. Sin embargo, este método presenta la misma gran desventaja que la que tiene el método 2P, la falta de puntos votantes cuando hay circunferencias incompletas u ocluidas en la imagen, causando su incapacidad de detectar dichos círculos.

Los métodos implementados fueron comparados respecto a sus resultados y sus tiempo de cómputo, con funciones disponibles en paquetes de MATLAB, y con algunas funciones desarrolladas para el entorno MATLAB por expertos en programación y accesibles en internet. Una conclusión es que la calidad de los resultados de los métodos de acumulación de evidencia dependen en gran medida de la calidad de detección de máximos locales en la matriz acumulador.

Es importante resaltar que se describió una técnica para la detección de circunferencias en una simulación de imagen astronómica, proveniente de un artículo de investigación en astronomía, conteniendo una nube de puntos con muchísimo ruido. En este caso, la técnica propuesta utiliza el método general de Hough (H3D), seguido por una evaluación adicional de la presencia de los círculos detectados, para confirmar o no, su verdadera presencia en la imagen.

Por último, se implementó la versión más sencilla del Algoritmo de Bhowmick para determinar si una curva digital dada es un círculo digital, y en caso de que no, segmentarla en sus segmentos circulares, eso con el fin de entender y de ilustrar el funcionamiento de este interesante método.

Trabajo a futuro:

Las siguientes ideas pueden servir para continuar trabajando sobre los temas descritos en esta tesis:

- Utilizar algoritmos de ajuste de circunferencias a las imágenes de nubes de puntos.
- Probar algún método de optimización o estadístico para imágenes como la imagen astronómica.
- Hacer uso de la técnica usada en la imagen astronómica para mejorar los resultados para imágenes con circunferencias continuas.
- Experimentar los métodos descritos en esta tesis para imágenes más grandes, en vista de que imágenes astronómicas reales serían de mucho mayores dimensiones.
- Mejorar la determinación de los máximos locales.

Bibliografía

- [1] Aschoff, T., and Spiecker, H. (2004). Algorithms for the automatic detection of trees in laser scanner data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(Part 8), W2.
- [2] Ballard, D.H., and Brown, C.M. (1982). *Computer Vision* (Vol. 2). Prentice-Hall, Englewood Cliffs, NJ, USA.
- [3] Ballard, D.H. (1981). Generalizing the Hough transform to detect arbitrary shapes. *Pattern recognition*, 13(2), 111-122. NY, USA.
- [4] Basset, B.A., and Hlozek, R., Baryon Acoustic Oscillations, arXiv:0910.5224v1 [astro-ph.CO] 27 Oct 2009, 42 pages.
- [5] Bhowmick, P., Bera, S., and Bhattacharya, B.B. (2009). Digital circularity and its applications, *Proceedings of the 13th International Workshop on Combinatorial image Analysis (IWCIA'2009, Playa del Carmen, Mexico, Nov. 2009)*, LNCS Vol. 5852, 1-15, Springer Verlag, Berlin Heidelberg.
- [6] Bhowmick, P. and Bhattacharya, B.B. (2012). Digital straightness, circularity, and their applications to image analysis, In: V. Brimkov and R.P. Barneva (Eds.). *Digital Geometry Algorithms - Theoretical Foundations and Applications to Computational Imaging*, *Lecture Notes in Computational Vision and Biomechanics* Vol. 2, 247- 299, Springer Science+Business Media Dordrecht.
- [7] Bukowska, D.M., et al. (2015). Semi-automated identification of cones in the human retina using circle Hough transform. *Biomedical Optics Express*, 6(12), 4676-4693.
- [8] Camurri, M., Vezzani, R., and Cucchiara, R. (2014). 3D Hough transform for sphere recognition on point clouds. *Machine Vision and Applications*, 25(7), 1877-1891.
- [9] Chauris, H., Karoui, I., Garreau, P., Wackernagel, H., Craneguy, P., and Bertino, L. (2011). The circllet transform: A robust tool for detecting features with circular shapes. *Computers and geosciences*, 37(3), 331-342.
- [10] Coope, I.D. (1993). Circle fitting by linear and nonlinear least squares. *Journal of Optimization Theory and Applications*, 76(2), 381-388.

- [11] Costa, L.d.F., Cesar, R.M.(2001). Shape Analysis and Classification, CRC Press, USA.
- [12] Damaschke, P. (1995). The linear time recognition of digital arcs. Pattern Recognition Letters, 16(5), 543-548.
- [13] Equis, S., Flandrin, P., and Jacquot, P. (2011). Phase extraction in speckle interferometry by a circle fitting procedure in the complex plane. Optics Letters, 36(23), 4617-4619.
- [14] Freeman, H. (1961). On the encoding of arbitrary geometric configurations. IRE Transactions on Electronic Computers, (2), 260-268.
- [15] Gander, W., Golub, G.H., and Strebel, R. (1994). Least-squares fitting of circles and ellipses, BIT, Vol. 34, 558-578.
- [16] Gonzalez, R. C., and Woods, R. E. (2001). Digital Image Processing, Prentice Hall Inc., New Jersey, USA.
- [17] Gonzalez, R.C., Woods, R.E., and Eddins, S.L. (2004). Digital image using Matlab processing. Person Prentice Hall, Lexington.
- [18] Hough, P.V.C. (1962). Method and means for recognizing complex patterns, U.S. Patent 3,069,654, Dec. 18, 1962.
- [19] Ioannou, D., Huda, W., and Laine, A.F. (1999). Circle recognition through a 2D Hough transform and radius histogramming. Image and vision computing, 17(1), 15-26.
- [20] Kimme, C., Ballard, D., and Sklansky, J. (1975). Finding circles by an array of accumulators. Communications of the ACM, 18(2), 120-122.
- [21] Klette, R., and Rosenfeld, A. (2004). Digital Geometry: Geometric Methods for Digital Picture Analysis. Morgan Kaufmann, San Francisco, USA.
- [22] Lee, B.R., and Nguyen, H.C. (2015). Development of laser-vision system for three-dimensional circle detection and radius measurement. Optik-International Journal for Light and Electron Optics, 126(24), 5412-5419.
- [23] Liu, T., Zhao, W., Wang, Z., and Jia, S. (2008). An image-based circle extraction method of three-dimensional point cloud data. Proceedings of the Second International IEEE Symposium on Intelligent Information Technology Application (IITA'08), Vol. 2, 447-451.
- [24] Magruder, L.A., Leigh, H.W., Soderlund, A., Clymer, B., Baer, J., and Neuenschwander, A.L. (2016). Automated feature extraction for 3-dimensional point clouds. In SPIE (International Society for Optics and Photonics) Defense + Security, 98320F-98320F.
- [25] Maisonobe, L. (2007). Finding the circle that best fits a set of points. October 25th, 2007, <https://www.spaceroots.org/documents/circle/circle-fitting.pdf> (accesado septiembre 2017).

- [26] Nixon, M.S., and Aguado, A.S. (2008). Feature extraction and image processing for computer vision. 2nd Edition, Academic Press, Elsevier.
- [27] Ogundana, O.O., Coggrave, C.R., Burguete, R.L., and Huntley, J.M. (2011). Automated detection of planes in 3-D point clouds using fast Hough transforms. *Optical Engineering*, 50(5), 053609-053609. and, Ogundana, O.O., Coggrave, C.R., Burguete, R.L., and Huntley, J.M. (2007). Fast Hough transform for automated detection of spheres in three-dimensional point clouds. *Optical Engineering*, 46(5), 051002-051002.
- [28] Rabbani, T., and Hough, F.V.D.H.E. (2005). Transform for automatic detection of cylinders in point clouds, *Laser Scanning 2005. Proceedings of the ISPRS Workshop (Enschede, The Netherlands)*, September 12-14, 2005.
- [29] Shapiro, L., and Stockman, G.C. (2001). *Computer Vision*. Prentice Hall, New Jersey, USA.
- [30] Smítka, V., and Štroner, M. (2013). 3D scanner point cloud denoising by near points surface fitting. *SPIE (International Society for Optics and Photonics) Optical Metrology*, 87910C-87910C.
- [31] Sossa Azuela, J.H. (2013). *Visión Artificial - Rasgos descriptores para el reconocimiento de objetos*. Editorial Ra-Ma, Madrid, España, ISBN 978-84-9964-142-3.
- [32] Teng, Z., Kim, J. H., and Kang, D. J. (2011). Ellipse detection using an improved randomized Hough transformation. *ISandT/SPIE (International Society for Optics and Photonics) Electronic Imaging*, 78700U-78700U.
- [33] Torii, A., and Imiya, A. (2007). The randomized-Hough-transform-based method for great-circle detection on sphere. *Pattern Recognition Letters*, 28(10), 1186-1192.
- [34] Umbach, D., and Jones, K. N. (2003). A few methods for fitting circles to data. *IEEE Transactions on Instrumentation and Measurement*, 52(6), 1881-1885.
- [35] Vosselman, G., Gorte, B.G., Sithole, G., and Rabbani, T. (2004). Recognising structure in laser scanner point clouds. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 46(8), 33-38.
- [36] Wiederhold, P. (2016). Digital Geometry, 13 pages, In: P. Laplante (Ed.), *Encyclopedia of Computer Science and Technology*, Second Edition, 1500 pages, CRC Press, Taylor and Francis Group, DOI: 10.1081/E-ECST2-120054031.
- [37] Wiederhold, P. (2016). Geometría digital, curso para la maestría (60 horas, Enero-Abril 2016), Departamento de Control Automático, CINVESTAV-IPN, Zacatenco, CDMX.
- [38] Wolfram Research, Inc. (1999-2017). Wolfram MathWorld: Ellipse. <http://mathworld.wolfram.com/Ellipse.html> (accesado febrero 2017).

- [39] Yi, W., and Marshall, S. (2000). Circle detection using Fast Finding and Fitting (FFF) algorithm. *Geo-spatial Information Science*, 3(1), 74-78.
- [40] Yip, R.K., Tam, P.K., and Leung, D.N. (1992). Modification of Hough transform for circles and ellipses detection using a 2-dimensional array. *Pattern Recognition*, 25(9), 1007-1022.
- [41] Young, D.(Marzo, 2016). Hough transform for circles. Mathworks. Obtenido el 8 de agosto, 2017, desde <https://es.mathworks.com/matlabcentral/fileexchange/26978-hough-transform-for-circles?focused=6019424&tab=example>.
- [42] Yuen, H.K., Princen, J., Illingworth, J., and Kittler, J. (1990). Comparative study of Hough transform methods for circle finding. *Image and vision computing*, 8(1), 71-77.
- [43] Zhang, S., Wootton, J., and Chisholm, A. (2005). Geometric feature extraction from point clouds obtained by laser scanning. *Proceedings of the EPIC/SPIE (International Society for Optics and Photonics) Workshop on Laser Applications in Europe*, 61570R-61570R.

FIRMAS

Directora de tesis
Dra. Petra Wiederhold Grauert
Departamento de Control Automático

Dra. Petra Wiederhold Grauert (Sinodal)
Departamento de Control Automático

Dra. Martha Rzedowski Calderón (Sinodal)
Departamento de Control Automático

Dr. Wen Yu Liu (Sinodal)
Departamento de Control Automático