

# Universidad Michoacana de San Nicolás de Hidalgo.



Facultad de Ciencias Físico-Matemáticas.  
Mat. Luis Manuel Rivera Gutiérrez.

## Reconocimiento de animales silvestres por medio de la caracterización de patrones elípticos.

Tesis para optar al grado de Licenciada en Ciencias  
Físico-Matemáticas.

*Presenta:*

**Ana Isabel Castro Reyes**

**Asesor: Dra. Karina Mariela Figueroa Mora**

Morelia, Michoacán.  
Enero de 2016.

---

# Índice general

Índice de figuras	III
Índice de cuadros	V
Abstract	VI
Resumen	VII
<b>1. Introducción</b>	<b>1</b>
1.1. Antecedentes . . . . .	3
1.2. Objetivo general . . . . .	5
1.3. Objetivos particulares . . . . .	5
1.3.1. Trabajos publicados . . . . .	5
1.4. Organización de la tesis . . . . .	5
<b>2. Caracterización de patrones</b>	<b>6</b>
2.1. Conceptos básicos . . . . .	6
2.2. Segmentación . . . . .	8
2.3. Aplicación de filtros . . . . .	8
2.3.1. Filtro de Sobel . . . . .	11
2.3.2. Filtro de Canny . . . . .	13
2.4. Caracterización de las marcas con forma elíptica . . . . .	15
2.4.1. Cálculo de la elipse utilizando un método geométrico . . . . .	15
2.4.2. Cálculo de la elipse utilizando momentos de la imagen . . . . .	20
<b>3. Experimentación y resultados</b>	<b>24</b>
3.1. Segmentación de la imagen . . . . .	24
3.2. Selección del filtro digital para las imágenes . . . . .	25
3.3. Selección de áreas . . . . .	26
3.4. Cálculo de las elipses en las manchas . . . . .	28
3.4.1. Aplicación del método geométrico . . . . .	29

3.4.2. Momentos de la imagen . . . . .	34
3.5. Comparación de los métodos . . . . .	42
<b>4. Conclusión</b>	<b>45</b>
4.1. Trabajo a futuro . . . . .	46
<b>A. Implementación</b>	<b>48</b>

# Índice de figuras

1.1. Fotografía de un ocelote . . . . .	3
2.1. Imagen ampliada donde cada cuadro es un píxel. . . . .	7
2.2. Ejemplo de imagen procesada por MATLAB. . . . .	7
2.3. Imagen binarizada. . . . .	8
2.4. Imagen de un ocelote después de la segmentación. . . . .	9
2.5. Cambios horizontales en la imagen. . . . .	12
2.6. Cambios verticales en la imagen. . . . .	13
2.7. Ejemplos de animales con características físicas elípticas. . . . .	15
2.8. Elementos de la elipse. . . . .	16
2.9. Diámetros conjugados en una elipse. . . . .	17
2.10. Ejemplo del teorema de Tales. . . . .	17
2.11. Demostración del Teorema de Tales. . . . .	18
2.12. Construcción para encontrar el centro de la elipse. . . . .	19
2.13. Construcción de Rytz. . . . .	19
3.1. Ocelote en la fotografía 1. . . . .	24
3.2. Ocelote en la fotografía 2. . . . .	24
3.3. Contorno o perímetros de las manchas, rellenadas. . . . .	26
3.4. Manchas diferentes del torso después de que se rellenan las áreas. . . . .	27
3.5. Manchas iguales en las dos fotografías del torso. . . . .	27
3.6. Manchas regulares e irregulares en el torso del animal. . . . .	27
3.7. Construcción para encontrar el centro de la elipse. . . . .	29
3.8. Construcción de los diámetros conjugados. . . . .	29
3.9. Construcción para los tamaños y direcciones de los semiejes. . . . .	31
3.10. Semiejes mayores de las 10 manchas en las dos fotos. . . . .	34
3.11. Semiejes menores de las 10 manchas en las dos fotos. . . . .	34
3.12. Ángulos de las 10 manchas en las dos fotos. . . . .	35
3.13. Semiejes mayores de las 10 manchas en las dos fotos. . . . .	38
3.14. Semiejes menores de las 10 manchas en las dos fotos. . . . .	38
3.15. Ángulos de las 10 manchas en las dos fotos. . . . .	38
3.16. Semiejes mayores de las 18 manchas del torso. . . . .	39

3.17. Semiejes menores de las 18 manchas del torso. . . . .	39
3.18. Ángulos de las 18 manchas del torso en las dos fotos. . . . .	39
3.19. Manchas del torso en la foto 1. . . . .	40
3.20. Manchas del torso en la foto 2. . . . .	40
3.21. Semiejes mayores de las manchas en los dos torsos. . . . .	40
3.22. Semiejes menores de las manchas en los torsos. . . . .	40
3.23. Ángulos de las manchas de los dos torsos. . . . .	41
3.24. Centroides de las manchas en los dos torsos. . . . .	41
3.25. Torsos con su elipses correspondientes. . . . .	42
3.26. Ángulos de las 10 manchas en la foto 1. . . . .	42
3.27. Ángulos de las 10 manchas en la foto 2. . . . .	42
3.28. Semiejes mayores de las 10 manchas en la foto 1. . . . .	43
3.29. Semiejes menores de las 10 manchas en la foto 1. . . . .	43
3.30. Semiejes mayores de las 10 manchas en la foto 2. . . . .	43
3.31. Semiejes menores de las 10 manchas en la foto 2. . . . .	43
3.32. Centros de las 10 manchas en la foto 1. . . . .	44
3.33. Centros de las 10 manchas en la foto 2. . . . .	44
4.1. Centroides y vectores de las manchas caracterizadas en la foto 1. . . . .	46
4.2. Centroides y vectores de las manchas caracterizadas en la foto 2. . . . .	46
4.3. Ocelote de la figura 3.1 caracterizado. . . . .	47
4.4. Ocelote de la figura 3.2 caracterizado. . . . .	47

# Índice de cuadros

2.1. Tipos de bordes. Del lado derecho se muestran los bordes de algunas manchas correspondientes a su izquierda. . . . .	10
3.1. Aplicación de los filtros de Canny y de Sobel. . . . .	25
3.2. Grupo de manchas. . . . .	28
3.3. Resultados del método geométrico. Manchas del ocelote en la primer fotografía. . . . .	31
3.4. Resultados del método geométrico. Manchas del ocelote en la segunda fotografía. . . . .	32
3.5. Resultados para las manchas regulares 1 y 4. . . . .	33
3.6. Resultados para las manchas irregulares 8 y 9. . . . .	33
3.7. Resultados de los momentos de la imagen. Manchas del ocelote en la primer fotografía . . . . .	36
3.8. Resultados de los momentos de la imagen. Manchas del ocelote en la segunda fotografía. . . . .	36
3.9. Resultados de las manchas regulares 1 y 4. . . . .	37
3.10. Resultados de las manchas irregulares 8 y 9. . . . .	37

# Abstract

In this work we present a feature-based approach of a recognition system for wildlife. Specifically we study pictures of a kind of feline known as *ocelot* from Michoacan's state of Mexico. The feature extraction phase starts applying two methods of edge detectors: Sobel and Canny. After the edge detection, we find that the best way to extract the features of the patterns of our pictures is using elliptical shape. We use two methods to construct the ellipses, one is based on geometric construction and the other one is the method of image moments-based. The results show that the best extract the features with elliptical patterns of the pictures was obtaining with the method of moments.

# Resumen

En esta tesis se presenta la etapa de caracterización para un sistema de reconocimiento de animales silvestres. Las pruebas se realizan con fotografías de un tipo de felino conocido como *ocelote* del estado de Michoacán, México. Para abordar la etapa la caracterización, primero se utilizan los filtros de Sobel y de Canny para el reconocimiento de bordes en la imagen. Una vez filtradas las imágenes, se procede a la caracterización de los patrones en el animal con forma elíptica mediante dos métodos propuestos: el primero basado en una construcción geométrica y el segundo basado en los momentos propios de la imagen. Los resultados muestran que la mejor caracterización de los patrones elípticos de las imágenes se obtiene con el método de momentos.

Palabras clave: *Reconocimiento de patrones, procesamiento digital de imágenes, visión computacional, filtros digitales, caracterización digital.*

# Capítulo 1

## Introducción

El reconocimiento automático de animales puede ser una valiosa herramienta para una gran variedad de investigaciones biológicas y aplicaciones para el monitoreo ambiental, es decir, el senso de animales, ya que actualmente por la destrucción que ha generado la especie humana sobre la flora y la fauna, lamentablemente existe una cantidad significativa de especies en amenaza o en peligro de extinción.

En general los sistemas de reconocimiento automático, consisten en 4 etapas: obtención de datos, segmentación del área de interés en los datos, caracterización de esas áreas y por último la clasificación o reconocimiento.

La etapa de obtención de datos es afectada por la pureza o calidad con la cual los datos son extraídos. Para conocer la fauna en alguna zona, se emplean dos tipos de técnicas: las intrusivas y las no intrusivas. Las intrusivas requieren manipular los especímenes para marcarlos, mediante un proceso de captura y liberación. Este proceso puede modificar el comportamiento del ejemplar al sentirse amenazado, además de ser costoso y laborioso. En las técnicas no intrusivas se pretende obtener información de la especie, evitando interactuar con el ejemplar. Un ejemplo de una técnica no intrusiva, la cual es usada por biólogos, son las cámara trampa puestas en regiones de interés por periodos de tiempo prolongado. Las cámaras trampa cuentan con sensores de movimiento para que tomen la foto cuando detecten la presencia de un objeto. Después de un cierto tiempo se debe recoger la cámara trampa para extraer el conjunto de fotografías y realizar su análisis.

Una vez que se logra tener las imágenes, éstas normalmente son procesadas una por una manualmente por expertos para identificar a los miembros de la especie. El hecho de que ésta tarea sea realizada por personas resulta caro, lento y no excluye los errores humanos. Nosotros estamos interesados en un método que, utilizando este tipo de imágenes, sea capaz de identificar

automáticamente animales. Sabemos que es relativamente sencillo conseguir que una máquina capte lo que le rodea; pero el hecho de que sea capaz de interpretar lo que ha captado no es fácil. La complejidad involucrada radica en que, *"la percepción, si bien es una tarea que todos los seres humanos llevamos a cabo diariamente, nadie sabe realmente cómo lo conseguimos"* [Pertusa, 2003].

A continuación se describe cada una de las siguientes etapas en un sistema de reconocimiento.

- **Segmentación.** En esta etapa se debe extraer exclusivamente al animal del resto de la imagen. Es decir, limpiar el área donde no está presente el animal, puesto que la información importante es sólo el animal. El resto de la imagen se considera ruido, porque en este caso, no aporta información para el reconocimiento de cada individuo. Para el tratamiento de las fotografías se requiere de técnicas de procesamiento de imágenes para atender problemas específicos como cambios de iluminación, colores de la imagen, etc.
- **Caracterización.** El objetivo de esta fase es identificar los puntos estratégicos para la caracterización de un individuo respecto de otro. Observar que la segmentación debería dejar sólo la parte donde se hará esta caracterización. En este caso se tienen varios retos como la rotación, escalas, movimiento, etc.
- **Clasificación/reconocimiento.** Se pretende clasificar o agrupar a los objetos para conseguir alguna forma de distinguirlos. Es importante mencionar que de no conseguir una identificación se deberá regresar a las etapas anteriores.

En particular estamos interesados en la segunda de estas etapas (caracterización) y probar la propuesta con el ocelote, un mamífero de la familia de los félidos. Las fotografías que se utilizaron fueron captadas con cámaras trampa colocadas en el interior del estado de Michoacán, por un grupo de investigación en biología animal liderado por el Dr. Tiberio Monterrubio [Charre et al., 2013]. El ocelote es una especie de felino de tamaño mediano, con una longitud de cabeza y cuerpo de 70 – 90 cm, cola relativamente corta (30 – 40 cm, un 45 % de la longitud de cabeza y cuerpo) y un peso en el entorno de los 11 kg. Una de sus características más llamativas es su piel manchada, vea un ejemplo de este animal en la figura 1.1.



Figura 1.1: Fotografía de un ocelote en el estado de Michoacán captada con una cámara trampa.

En la etapa de caracterización de los ocelotes una tarea relevante es encontrar los puntos estratégicos que determinan a cada individuo. En las especies animales que tienen manchas o líneas, éstas sirven como marcaje natural para poder distinguir un espécimen de otro. En esta tesis se proponen dos métodos para la caracterización de las manchas que tienen los ocelotes usando patrones elípticos.

## 1.1. Antecedentes

Los sistemas de reconocimiento de imágenes son más populares en la investigación de huellas digitales, las señales de audio, etc. Sin embargo, el desafío aumenta cuando las imágenes son de animales silvestres tomadas de manera automática. En este caso hay variaciones en la posición, la escala, la rotación, los pliegues en la piel, etc.

Algunos trabajos que se han desarrollado para reconocimiento de animales silvestres son los siguientes (en general, la mayoría de ellos usan técnicas intrusivas y requieren de imágenes con ciertas especificaciones, algo difícil de conseguir con cámaras trampa y sensores):

Jurgen den Hartog et al. [Hartog and Reijns, 2014] implementaron el software *I<sup>3</sup>S Pattern*, (*Interactive Individual Identification System*), para el reconocimiento de patrones existentes en el cuerpo de animales, el cual requiere de interacción con el usuario. El usuario debe proporcionar el área de la fotografía donde se encuentra el patrón del animal e *I<sup>3</sup>S Pattern* genera una huella digital única para cada imagen; posteriormente ésta se compara con las

contenidas en la base de datos para su identificación. La principal desventaja de este software es que las fotografías deben cumplir con ciertas condiciones (grado de inclinación de 30 grados), por lo que resulta poco útil para nuestro problema.

En [Whitehead et al., 2000] y [Kelly, 2001] los autores implementaron sistemas de foto-identificación de animales marcados naturalmente, su base de datos se compone de imágenes de las especies en cuestión. En particular [Kelly, 2001], los autores han trabajado con guepardos del Parque Nacional Serengeti de Tanzania, éstos fueron reconocidos semiautomáticamente extrayendo manualmente el área entre el omóplato, la articulación de la cadera, la línea del vientre, y la columna vertebral, después se extrajo una muestra del patrón del pelaje en cada animal. Estos datos se almacenaron en una computadora como una matriz de números, que consiste en intensidades en la escala de grises, para ser comparadas finalmente. Esta técnica no es muy eficiente en términos de tiempo de procesamiento porque requiere del trabajo de un operador humano para procesar cada imagen.

En cuanto a animales silvestres capturados con cámaras trampa por considerarse con una técnica no intrusiva se tienen pocos avances en el mundo, uno muy reciente es el que se presenta en [Figuroa and Camarena, 2014], los autores proponen un método para extraer al animal de la fotografía. El método propuesto consiste en obtener la diferencia de dos imágenes cualesquiera tomadas de la misma cámara, (recuerde que son colocadas en objetos inamovibles).

Por otro lado, el tema de contornos cerrados ha sido estudiado ampliamente. Una manera de caracterizar un contorno cerrado, es representándolo mediante una elipse. En [Kuhl, 1982], los autores utilizan propiedades elípticas de los coeficientes de Fourier para su caracterización. En [Pacheco, 2015], identifican diatomeas y caracterizan las valvas utilizando descriptores geométricos que aproximan el contorno de éstas como una elipse. Por otro lado, Juan Carlos Espinosa [Espinosa, 2013], implementa un algoritmo con el método de cuerda tangente para detectar elipses en las imágenes, técnica que tiene muy buenos resultados en la práctica; pero sólo para dibujos de elipses regulares.

El reto del reconocimiento de animales silvestres es un problema abierto en el mundo pues el animal puede ser tomado en cualquier posición al momento de la activación de la cámara, además de que no es simétrica la distribución de manchas en su cuerpo, por lo que, la clasificación no es una tarea sencilla. Actualmente, los biólogos hacen esta tarea de manera visual, lo que hace que los resultados sean inexactos.

## **1.2. Objetivo general**

Diseñar un algoritmo o procedimiento para la caracterización de animales silvestres, representando sus patrones como elipses.

## **1.3. Objetivos particulares**

- Detección de contornos por medio de filtros digitales.
- Caracterización de los puntos de interés, o de los patrones mediante elipses.
- Aplicación de dos métodos para la caracterización de las elipses.
- Validación de la clasificación de los ocelotes usando sus manchas.

### **1.3.1. Trabajos publicados**

Los resultados de esta tesis fueron publicados en:

- Póster: Reconocimiento de animales silvestres mediante la caracterización de patrones elípticos, 10° Congreso Estatal de Ciencia, Tecnología e Innovación y Cuarto Encuentro de Jóvenes Investigadores del Estado de Michoacán.

## **1.4. Organización de la tesis**

La tesis está organizada de la siguiente manera:

En el capítulo 2 se muestra la teoría de los filtros digitales que se proponen ser aplicados en las imágenes y la teoría de los dos métodos sobre la obtención de los parámetros de las elipses. En el capítulo 3 se muestran los resultados obtenidos. Finalmente en el capítulo 4 se presentan las conclusiones de este trabajo. Los códigos utilizados en esta tesis pueden ser vistos en el apéndice.

# Capítulo 2

## Caracterización de patrones

Antes de comenzar se presentarán algunos conceptos básicos para poner en contexto al lector.

### 2.1. Conceptos básicos

Una **Imagen** se puede definir como una función de dos dimensiones  $f(x, y)$  donde la amplitud de la función está dada por la intensidad de gris en un punto  $(x, y)$  de la imagen. Si la imagen se representa por una matriz, los índices de renglón y columna identifican un punto en la imagen llamado píxel [Ramírez, 2006].

Un **píxel** (acrónimo del inglés *picture element*, elemento de la imagen) es la menor unidad homogénea en color que forma parte de una imagen digital. En las imágenes, cada píxel se codifica mediante un conjunto de bits; por ejemplo, puede codificarse un píxel con un byte (8 bits), de manera que cada píxel admite hasta 256 ( $2^8$ ) variaciones de color, de 0 a 255. Existen diferentes modelos para representar una imagen, en particular el modelo de intensidad de colores primarios RGB (*Red, Green, Blue*, “Rojo, Verde, Azul”) en una escala de 0 a 255, los colores rojo, verde y azul corresponden a las triadas  $(255,0,0)$ ,  $(0,255,0)$  y  $(0,0,255)$ . Si el espacio de colores está normalizado entonces la representación, para los colores anteriores, sería  $(1,0,0)$ ,  $(0,1,0)$  y  $(0,0,1)$ . Cuando una imagen es ampliada en la pantalla de la computadora, se pueden observar los píxeles que la componen, como se observa en la figura 2.1.

En una imagen representada como una matriz de píxeles, ésta se define por su altura y anchura (en píxeles) y por su profundidad de color (en bits

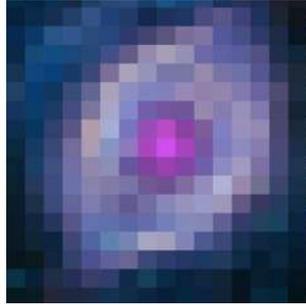


Figura 2.1: Imagen ampliada donde cada cuadro es un píxel.

por píxel), que determina el número de colores distintos que se pueden almacenar en cada píxel. En MATLAB por ejemplo, a la imagen de la figura 2.2 le correspondería una matriz de dimensiones  $57 \times 58 \times 3$ . Como se observa, la anchura va de 0 a 58 píxeles y la altura de 0 a 57, tal cual, la lectura de una matriz.

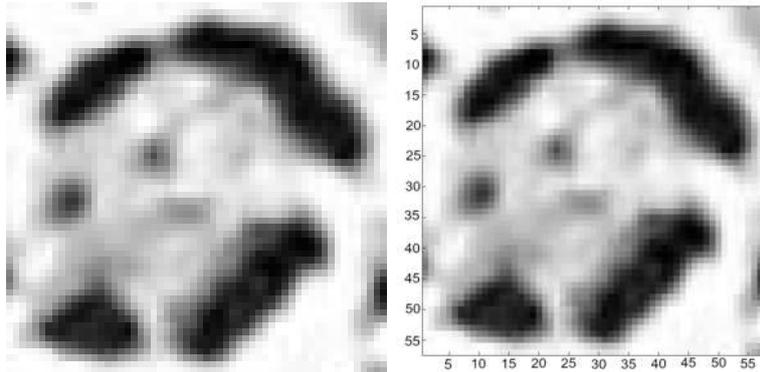


Figura 2.2: Ejemplo de imagen procesada por MATLAB.

Por otro lado, el **área** de un objeto en la imagen se define como el número de píxeles que lo forman, en la figura 2.3, el área sería igual a 30 píxeles.

Cuando se hace referencia a imágenes binarias en esta tesis, son aquellas imágenes como la mostrada en la figura 2.3, en las que el objeto está conformado de píxeles negros y el fondo de píxeles blancos o viceversa.

1	1	1	1	1	1	1	1	1	1
1	0	0	0	1	1	0	0	0	1
1	1	0	1	1	1	1	0	1	1
1	1	0	1	1	1	1	0	1	1
1	1	0	1	1	1	1	0	1	1
1	1	0	0	0	0	0	0	1	1
1	1	0	1	1	1	1	0	1	1
1	1	0	1	1	1	1	0	1	1
1	1	0	1	1	1	1	0	1	1
1	0	0	0	1	1	0	0	0	1
1	1	1	1	1	1	1	1	1	1

Figura 2.3: Ejemplo de una imagen binarizada. Los píxeles de color negro forman el objeto y los blancos el fondo.

## 2.2. Segmentación

El primer paso para extraer información de una imagen en un sistema de reconocimiento de objetos consiste en la segmentación de la imagen, ésta consiste en la descomposición o partición de la imagen en varias regiones homogéneas o disjuntas a partir de su contorno, su conectividad o en un conjunto de características de los píxeles que permita discriminar unas regiones de otras [Pérez, 2010]. En la fotografía que se muestra en la figura 1.1, note que, además del animal que se quiere caracterizar; se observa el escenario, compuesto de rocas principalmente, y la información de la foto capturada. Dado que solo se planea caracterizar al animal, el escenario no es importante para esta tarea.

En el reconocimiento automático de animales, la segmentación constituye una de las principales problemáticas a resolver, ya que permite la extracción de contornos, texturas y regiones lo cual facilita las etapas posteriores del procesamiento. En [Figuerola and Camarena, 2014] los autores mostraron que es posible eliminar el fondo donde se encuentra el objeto de interés y es posible llegar a obtener imágenes como la que se muestra en la figura 2.4. Por lo que, en las secciones siguientes solo se trabajará con estas imágenes segmentadas.

## 2.3. Aplicación de filtros

La aplicación de filtros digitales permite la manipulación de una imagen para resaltar u ocultar sus zonas, puesto que cuando se trata de implementar un sistema de reconocimiento, en la computadora para analizar imágenes,

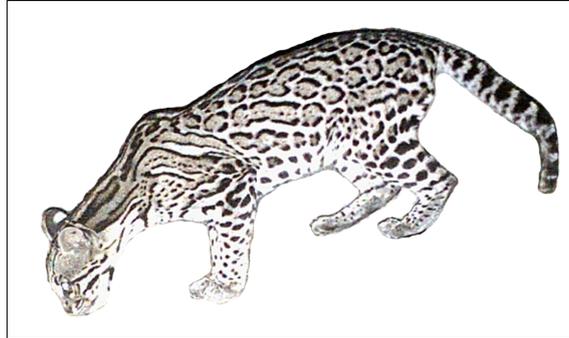
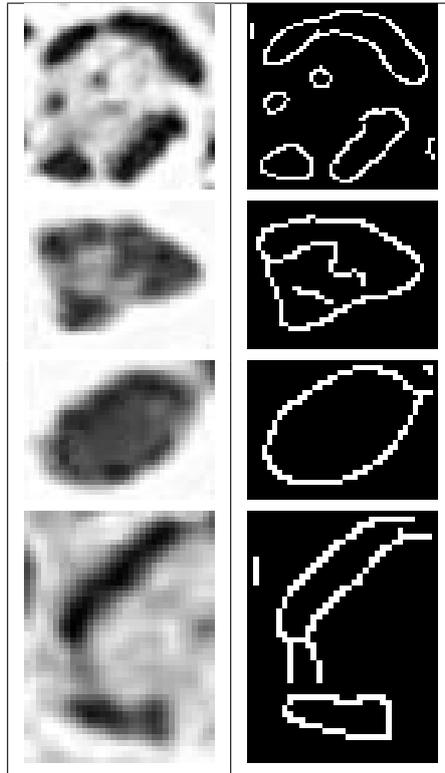


Figura 2.4: Imagen de un ocelote después de la segmentación.

es necesario que el algoritmo encuentre el objeto que queremos caracterizar o medir. Para eso su perímetro debe estar cerrado, es decir, los bordes del elemento deben estar claramente marcados. Esto pasa cuando la diferencia entre el fondo y el objeto está determinada por un cambio notorio de los píxeles que definen el perímetro del objeto. Lo ideal sería tener imágenes como la mostrada anteriormente en la figura 2.4. Sin embargo, este cambio no siempre es obvio ya que en la imagen se presentan distintos tipos de bordes, los cuales pueden estar difusos, mal marcados, indefinidos o interrumpidos, por ejemplo véase el cuadro 2.1, la columna de la izquierda tiene manchas más notorias que otras. La columna de la derecha muestra la imagen binarizada correspondiente a su izquierda. Note que en la imagen binarizada se pierden tonos de gris; pero se gana en la clara delimitación del área de interés, aunque infortunadamente no siempre resultan imágenes con formas cerradas.

Otro obstáculo en esta etapa es que la diferencia de intensidad de gris entre objetos es poca, o en algunos objetos se presentan sombras alrededor, por lo que se presentan cambios de nivel de gris consecutivos y la tarea de delimitar el perímetro del objeto o de decidir qué píxeles pertenecen a su perímetro, no es sencilla. El incorporar o no incorporar píxeles al momento de marcar el borde, puede producir importantes variaciones en las dimensiones del objeto.



Cuadro 2.1: Tipos de bordes. Del lado derecho se muestran los bordes de algunas manchas correspondientes a su izquierda.

Un método adecuado para delimitar los bordes en cualquier caso es el uso de filtros que enfatizan la diferencia entre píxeles. A partir de la función de nivel de gris de una imagen bidimensional, en cada píxel, se pueden calcular las derivadas (o razones de cambio) de cada color, para esto es posible aplicar un conjunto de filtros que simulan el resultado del cálculo de las derivadas. De esta manera, si consideramos que los puntos que ocupan la posición media en la curva de la región corresponden a los puntos de inflexión de la función, la primera derivada nos mostraría los puntos de inflexión como máximos o mínimos según se trate de puntos de inflexión en regiones ascendentes o descendentes.

La aplicación de un filtro matricial implica una serie de operaciones matemáticas sobre la imagen, operando sobre pequeños grupos de píxeles (por lo general grupos de nueve elementos), que tiene como objeto calcular la media ponderada de los píxeles de cada grupo, y cuyo resultado se asigna a un solo píxel en la nueva imagen resultante (el píxel central del grupo).

Por lo general las matrices de coeficientes son cuadradas y de orden impar. El filtro se aplica sobre la matriz de la imagen comenzando a operar desde la

esquina superior izquierda. El píxel central del filtro indica el lugar, el píxel, en el que se sustituirá el valor de la operación correspondiente. La mayoría de los filtros contienen operaciones, como las del siguiente ejemplo con un filtro de  $3 \times 3$ .

$$\begin{bmatrix} g_{11} & g_{12} & g_{13} & g_{14} & g_{15} & \dots \\ g_{21} & g_{22} & g_{23} & g_{24} & g_{25} & \dots \\ g_{31} & g_{32} & g_{33} & g_{34} & g_{35} & \dots \\ g_{41} & g_{42} & g_{43} & g_{44} & g_{45} & \dots \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix} * \begin{bmatrix} k_1 & k_2 & k_3 \\ k_8 & k_0 & k_4 \\ k_7 & k_6 & k_5 \end{bmatrix}$$

$$\Rightarrow g'_{22} = \frac{(k_1 \cdot g_{11} + k_2 \cdot g_{12} + k_0 \cdot g_{22} + \dots + k_5 \cdot g_{33})}{\sum k_i}$$

Los valores de gris de la nueva imagen que se está generando son los calculados en el filtrado. El píxel  $g'_{22}$  de la nueva imagen contendrá el valor resultante de la suma de los productos de los factores de la matriz del filtro por él mismo y sus vecinos, dividido por la suma de los factores del filtro matricial. Se trata de una medida ponderada del píxel central y sus vecinos adyacentes [Pertusa, 2003]. En el ejemplo, el filtro fue aplicado sólo a un píxel, el correspondiente a la posición  $g_{22}$ . La aplicación del filtro es a toda la matriz, recorriendo esta máscara cada vez en una posición a la derecha y posteriormente hacia abajo.

A continuación se explican los dos filtros utilizados en este trabajo, el filtro de Sobel y el filtro de Canny.

### 2.3.1. Filtro de Sobel

El filtro de Sobel u operador de Sobel [Sobel and Feldman, 1973] es utilizado en procesamiento de imágenes, especialmente como algoritmo de detección de bordes. El filtro Sobel consiste en el cálculo de derivadas en una imagen para detectar bordes. Un cambio alto en el gradiente es indicador de un cambio importante en la imagen. Matemáticamente, el gradiente de una función de dos variables (en este caso, la función de intensidad de la imagen) para cada punto es un vector bidimensional cuyos componentes están dados por las primeras derivadas de las direcciones verticales y horizontales. Para cada punto de la imagen, el vector gradiente apunta en la dirección del incremento máximo posible de la intensidad, y la magnitud del vector gradiente corresponde a la cantidad de cambio de la intensidad en esa dirección. Básicamente el operador Sobel es un operador de diferenciación discreta. Se

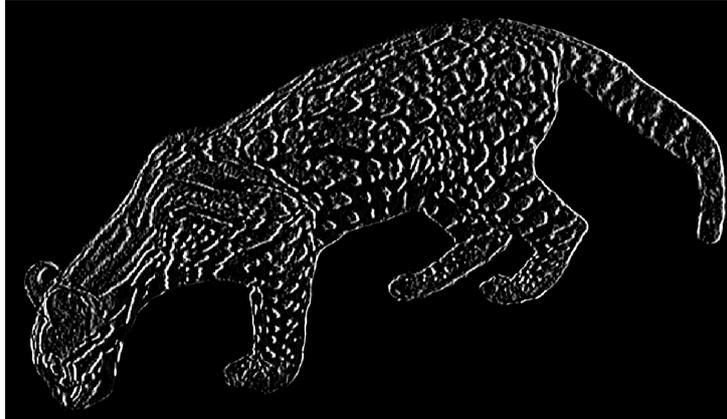


Figura 2.5: Cambios horizontales en la imagen.

calcula una aproximación del gradiente de una función de intensidad de la imagen. En las implementaciones de OpenCV y MATLAB el operador se combina con el filtro de Gauss y la diferenciación.

### Cálculo del filtro

El filtro de Sobel se calcula de la siguiente manera. Sea  $I$  la matriz de la Figura 2.4, a continuación se calculan las derivadas en ambos ejes por medio de núcleos (filtros) de tamaño impar de  $3 \times 3$ , para el cambio horizontal y vertical:

- a) Los cambios horizontales,  $G_x$ , se calculan mediante la convolución de  $I$  con un núcleo de Sobel.

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * I$$

Cuando se aplica la convolución a la figura 2.4 el resultado visual es el que se muestra en la figura 2.5, en la que se observa un énfasis en los bordes verticales.

- b) Los cambios verticales,  $G_y$ , se calculan mediante la convolución de  $I$  con el segundo núcleo de Sobel. El resultado de esta operación se puede observar en la figura 2.6 . Donde, en este caso, se resaltan los bordes horizontales.

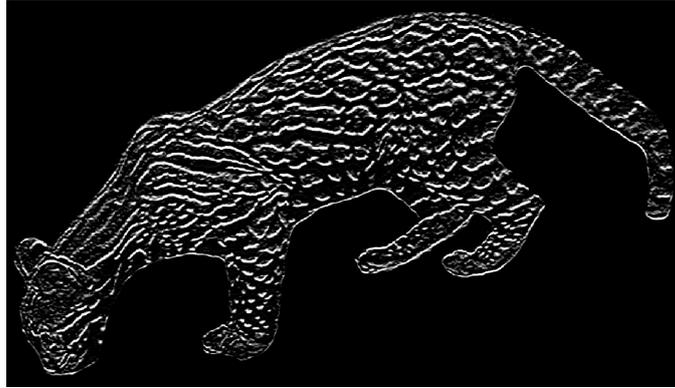


Figura 2.6: Cambios verticales en la imagen.

$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} * I$$

En cada punto de la imagen se calcula una aproximación del gradiente en ese punto mediante la combinación de los resultados anteriores:

$$G = \sqrt{G_x^2 + G_y^2}$$

Cuando el operador Sobel se aplica sobre una región con intensidad de imagen constante el resultado es un vector cero, y al aplicarlo en un punto sobre un borde resulta un vector que cruza el borde (perpendicular) cuyo sentido es de los puntos más oscuros a los más claros [OpenCV, 2011b].

### 2.3.2. Filtro de Canny

El otro filtro usado en este trabajo fue el desarrollado por John F. Canny [Canny, 1986], conocido como detector de bordes Canny o el detector óptimo. Este filtro intenta satisfacer tres criterios:

1. Tasa de error baja: una buena detección de la existencia de bordes únicos.
2. Buena localización: la distancia entre los píxeles del borde detectado y los píxeles del borde real tiene que ser minimizada.
3. Respuesta mínima: solamente se tiene una respuesta del detector por borde.

## Cálculo del filtro

**Paso uno: eliminación del ruido de la imagen.** Para esto se utiliza el filtro de Gauss, como el del ejemplo de un núcleo gaussiano de tamaño  $5 \times 5$ :

$$K = \frac{1}{159} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix}$$

**Paso dos: Encontrar el gradiente de intensidad de la imagen.** Para esto, se sigue un procedimiento análogo al de Sobel:

a) Se aplican las máscaras de convolución en la dirección  $x$  e  $y$ :

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix}$$

$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix}$$

b) Se calcula una aproximación del gradiente y la dirección con:

$$G = \sqrt{G_x^2 + G_y^2}$$
$$\theta = \frac{G_x}{G_y}$$

La dirección se redondea a uno de los cuatro ángulos posibles (0, 45, 90 ó 135).

**Paso tres: no supresión máxima** Esto elimina los píxeles que no se consideran parte de un borde. Por lo tanto, sólo se mantendrán las líneas finas (bordes candidatos).

**Paso cuatro: Histérisis.** En el paso final el filtro de Canny hace uso de dos umbrales (superior e inferior), esto se calcula de la siguiente manera [OpenCV, 2011a]:

a) Si un gradiente de píxel es mayor que el umbral superior, el píxel se acepta como un borde.

- b) Si el valor de un gradiente de píxel está por debajo del umbral inferior, entonces se rechaza.
- c) Si el gradiente de píxel está entre los dos umbrales, entonces será aceptado sólo si está conectado a un píxel que está por encima del umbral superior.

## 2.4. Caracterización de las marcas con forma elíptica

Una vez que se han marcado los bordes en las imágenes binarizadas, la segunda parte de la propuesta es la caracterización de las marcas del animal con forma elíptica. Es decir, se toman como base las singularidades físicas del animal, y se busca asignar a cada patrón una figura geométrica: en este caso una elipse. La elección de formas elípticas es porque existen varios animales que tienen forma elíptica, o bien, patrones de este tipo. Un ejemplo son las diatomeas, cuya morfología es muy parecida a una elipse, algunos felinos que poseen pieles manchadas o las tortugas cuya piel tiene escamas, como puede verse en la Figura 2.7, por mencionar algunos.



Figura 2.7: Ejemplos de animales con características físicas elípticas.

Básicamente se probaron dos métodos para la caracterización de los patrones en forma elíptica. El primero, basado en una construcción geométrica que nos permite obtener los parámetros de la elipse (centro, eje mayor, eje menor y dirección del eje mayor). Y el segundo basado en el cálculo de *momentos* de primer y segundo orden de la imagen de cada mancha.

### 2.4.1. Cálculo de la elipse utilizando un método geométrico

Las construcciones geométricas han estado presentes a lo largo de la historia y han sido una gran herramienta para la resolución de problemas utilizando solamente regla y compás. Ayudándonos de las propiedades de la elipse

y algunas de sus construcciones se implementa un código para encontrar los cuatro parámetros (centro, tamaños de los ejes y dirección del eje mayor) que la definen.

## Elipse

Una elipse es el lugar geométrico de un punto que se mueve en un plano de tal manera que la suma de sus distancias a dos puntos fijos de ese plano es siempre igual a una constante, mayor que la distancia entre esos dos puntos [Lehmann, 1989].

Véase la figura 2.8, en ésta los dos puntos fijos ( $F$  y  $F'$ ) se llaman focos de la elipse. La recta  $l$  que pasa por los focos se llama eje focal. El eje focal corta a la elipse en dos puntos,  $V$  y  $V'$ , llamados vértices. La porción del eje focal comprendida entre los vértices, o sea el segmento  $VV'$ , se llama **eje mayor**. El punto  $C$  del eje focal, punto medio del segmento que une los focos, se llama centro. La recta  $l'$  que pasa por  $C$  y es perpendicular al eje focal  $l$  se llama *eje normal*. El eje normal corta a la elipse en dos puntos  $A$  y  $A'$ , el segmento  $AA'$  se llama **eje menor**. Un segmento tal como  $BB'$ , que une dos puntos diferentes cualesquiera de la elipse, se llama *cuerda*. En particular, una cuerda que pasa por uno de los focos, tal como  $EE'$ , se llama *cuerda focal*. Una cuerda que pasa por  $C$ , tal como  $DD'$ , se llama un **diámetro**.

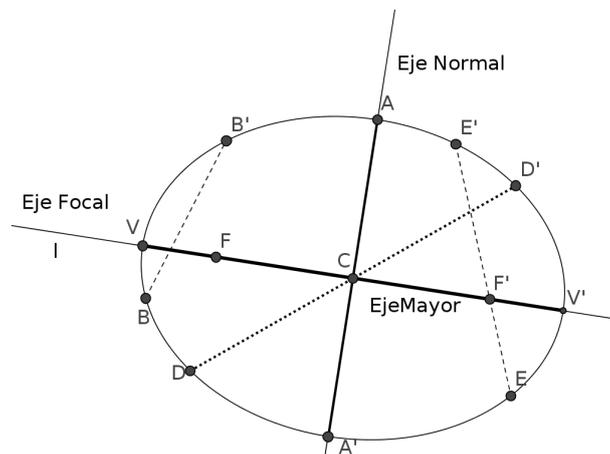


Figura 2.8: Elementos de la elipse.

En la figura 2.9 se pueden observar dos **diámetros conjugados**. Son diámetros conjugados en la elipse aquellos en donde cada uno de ellos divide en dos partes iguales a las cuerdas de la elipse trazadas paralelas al otro

$(DD', D_1D_1')$ . Estos diámetros se cortan en su punto medio. Los ejes de la elipse son los únicos diámetros conjugados normales entre sí.

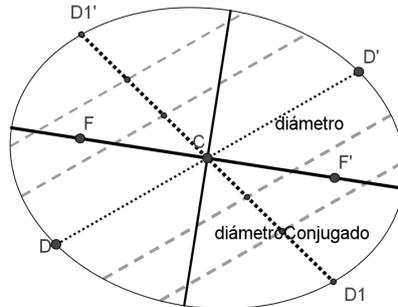


Figura 2.9: Diámetros conjugados en una elipse.

### Circunferencia de Tales

Para la obtención de los tamaños de los semiejes y dirección de la elipse se requiere de la circunferencia de Tales.

El segundo teorema de **Tales de Mileto** es un teorema de geometría enfocado a los triángulos rectángulos, las circunferencias y los ángulos inscritos, que se enlista en el siguiente enunciado:

*“Sea  $C$  un punto de la circunferencia de diámetro  $AB$ , distinto de  $A$  y de  $B$ . Entonces el triángulo  $ABC$ , es un triángulo rectángulo”.*

En la figura 2.10 se puede observar un ejemplo del teorema de Tales.

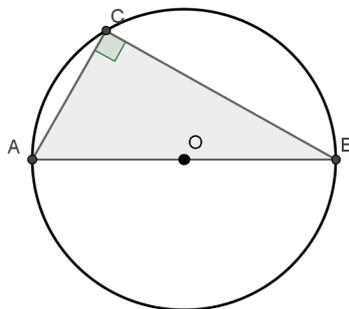


Figura 2.10: Ejemplo del teorema de Tales.

### Demostración:

En la circunferencia de centro  $O$  y radio  $r$  de la figura 2.11, los segmentos  $OA$ ,  $OB$  y  $OC$  son iguales por ser radios de la misma circunferencia. Por lo tanto los triángulos  $AOC$  y  $BOC$  son isósceles. La suma de los ángulos del triángulo  $ABC$  es:

$$2\alpha + 2\beta = 180^\circ$$

Dividiendo ambos lados de la ecuación por dos, se obtiene:

$$\widehat{ABC} = \alpha + \beta = \frac{180^\circ}{2} = 90^\circ \blacksquare$$

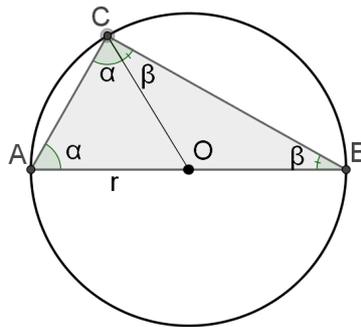


Figura 2.11: Demostración del Teorema de Tales.

### Construcción

Si queremos encontrar el centro de una elipse dada se trazan dos cuerdas  $S_1$  y  $S_2$ , por definición la cuerda que pasa por los puntos medios de esas cuerdas,  $M_1$  y  $M_2$ , es un diámetro de la elipse y por lo tanto pasa por el centro; así que el punto medio  $O$  del diámetro  $D$ , es el centro de la elipse, véase la figura 2.12.

Si se tiene que construir una elipse, conociendo un par de diámetros conjugados, en posición y magnitud, la construcción de Rytz nos da los ejes mayor y menor de la elipse en posición y magnitud. En la figura 2.13, se puede seguir esta construcción como se indica: a partir de los diámetros conjugados  $s$  y  $t$ , el segmento denotado por  $BM$  se hace girar  $90^\circ$  ahora se tiene

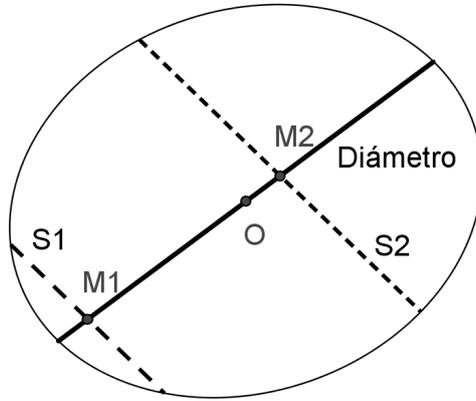


Figura 2.12: Construcción para encontrar el centro de la elipse.

el segmento  $(B'M)$ ; luego se une  $B'$  con  $A$  y se traza la circunferencia  $C_1$  de diámetro  $AB'$  y centro en el punto medio,  $T$ , de  $AB'$ . Las dimensiones de los ejes están dados por la distancia de los puntos de intersección de la recta  $L_1$  con  $C_1$  es decir  $(p_1, p_2)$ . Para encontrar las direcciones de los ejes, por  $B'$  y  $A$  se traza la recta  $u$ . La circunferencia de Tales con centro en el punto medio  $T$  que pasa por el centro  $M$  de la elipse corta a  $u$  en  $X$  y en  $Y$ . Las direcciones de los ejes están dadas por  $YM$  y  $XM$ .

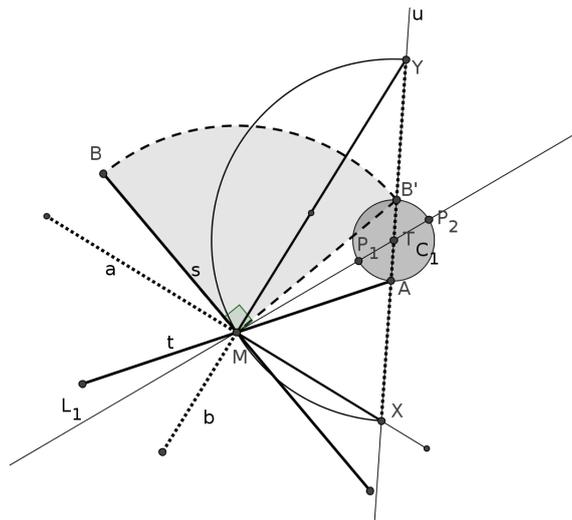


Figura 2.13: Construcción de Rytz.

### 2.4.2. Cálculo de la elipse utilizando momentos de la imagen

Los momentos de la imagen y los momentos invariantes juegan un papel importante en el análisis de la forma y el reconocimiento de un objeto porque están relacionados con el tamaño, la posición, la orientación y la forma de una región plana. En esta sección veremos que para que los descriptores sean invariantes a traslación se calculan los momentos centrales a partir de una imagen centrada con respecto a su centroide.

A continuación veremos como se calculan estos momentos. En [HU, 1962] los momentos de orden  $p + q$  de una imagen en escala de grises  $f(x, y)$  se definen como:

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y) dx dy \quad (2.1)$$
$$p, q = 0, 1, 2 \dots$$

Como las imágenes sólo están en blanco y negro, se puede definir la función como sigue:

$$f(x, y) = 1 \text{ si el píxel forma parte del objeto (píxel blanco).} \quad (2.2)$$
$$f(x, y) = 0 \text{ si el píxel forma parte del fondo (píxel negro).}$$

Por lo tanto la ecuación 2.1 queda de la siguiente forma:

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q dx dy \quad (2.3)$$
$$p, q = 0, 1, 2 \dots$$

en nuestro caso  $f(x, y)$  es una imagen digital así que la doble integral de 2.1 y 2.3 se reemplaza por una doble sumatoria, véase la ecuación 2.4:

$$m_{pq} = \sum_{x=1}^N \sum_{y=1}^N x^p y^q \quad (2.4)$$
$$p, q = 0, 1, 2 \dots$$

En imágenes binarias el momento de orden cero,  $m_{00}$ , representa el área de la figura.

Como nos interesa describir cada mancha como una elipse, para definir su centro, se calcula el **centroide** (también llamado centro de masa), que se

define como el centro de simetría de una figura geométrica y para ubicarlo se utilizan los momentos de orden uno,  $m_{10}$ ,  $m_{01}$ .

De la ecuación 2.4 los momentos  $m_{00}$ ,  $m_{10}$ ,  $m_{01}$  se obtienen como se muestra en la ecuación 2.5 :

$$\begin{aligned}
 m_{00} &= \sum_{x=1}^N \sum_{y=1}^N f(x, y), \text{ el área de la figura} & (2.5) \\
 m_{10} &= \sum_{x=1}^N x f(x, y) \\
 m_{01} &= \sum_{y=1}^N y f(x, y)
 \end{aligned}$$

De esa manera el cálculo para el centroide está dado por la ecuación 2.6.

$$\begin{aligned}
 \bar{x} &= \frac{m_{10}}{m_{00}} \\
 \bar{y} &= \frac{m_{01}}{m_{00}}
 \end{aligned}
 \tag{2.6}$$

Para que los descriptores sean invariantes a traslación se calculan los momentos centrales a partir de una imagen centrada con respecto a su centroide (véase la ecuación 2.7).

$$\mu_{p,q} = \sum_{x=1}^N \sum_{y=1}^N (x - \bar{x})^p (y - \bar{y})^q f(x, y) \tag{2.7}$$

Conociendo los momentos de primer orden podemos calcular los momentos centrales sin tener que evaluar la ecuación 2.7. Como se muestra en [HU, 1962] las expresiones para los primeros órdenes son las siguientes:

$$\begin{aligned}
\mu_{00} &= m_{00} & (2.8) \\
\mu_{01} &= 0 \\
\mu_{10} &= 0 \\
\mu_{20} &= m_{20} - \frac{m_{10}^2}{m_{00}} \\
\mu_{11} &= m_{11} - \frac{m_{10}m_{01}}{m_{00}} \\
\mu_{02} &= m_{02} - \frac{m_{01}^2}{m_{00}}
\end{aligned}$$

Otros descriptores que buscamos son los **tamaños de los ejes principales**, éstos, se pueden calcular a partir de los *valores propios* de la **matriz de inercia  $\mathbf{I}$** , la cual se compone de los momentos centrales  $\mu_{11}$ ,  $\mu_{20}$  y  $\mu_{02}$ .

$$\mathbf{I} = \begin{pmatrix} \mu_{20} & \mu_{11} \\ \mu_{11} & \mu_{02} \end{pmatrix}$$

Estos valores propios son los momentos principales y vienen dados por:

$$\lambda_{1,2} = \frac{(\mu_{20} + \mu_{02}) \pm \sqrt{\mu_{20}^2 + \mu_{02}^2 - 2\mu_{20}\mu_{02} + 4\mu_{11}^2}}{2} \quad (2.9)$$

En [Rocha et al., 2004] los **semiejes**,  $a$  (mayor) y  $b$  (menor), se expresan en términos de los momentos centrales de cero, primer y segundo orden en 2.10 y 2.11 respectivamente:

$$a = 2 * \sqrt{\frac{\lambda_1}{\mu_{00}}} \quad (2.10)$$

$$b = 2 * \sqrt{\frac{\lambda_2}{\mu_{00}}} \quad (2.11)$$

La **dirección** de la elipse (ángulo del eje mayor, respecto a la horizontal)  $\theta$  se calcula a partir del valor propio más grande. Es decir:

$$\theta = \tan^{-1} \left( \frac{\lambda_{\text{máx}} - \mu_{20}}{\mu_{11}} \right)$$

Otra descripción para la dirección  $\theta$ , se encuentra en [Rocha et al., 2004], y es la de la ecuación 2.12:

$$\theta = \frac{1}{2} * \tan^{-1} \left( \frac{2 * \mu_{11}}{\mu_{20} - \mu_{02}} \right) \quad (2.12)$$

En este capítulo se presentaron algunos conceptos básicos sobre el manejo de imágenes y se explicó la teoría detrás de los dos filtros que se aplicaron a las fotografías con el objetivo de resaltar los bordes presentes. También se presentaron los dos métodos que se utilizaron para la caracterización de las manchas en los ocelotes.

## Capítulo 3

# Experimentación y resultados

En este capítulo se presentan las etapas del procesamiento de las imágenes para la caracterización de las elipses en la misma. Así como los resultados con el método geométrico y el método de los momentos en la imagen. Para el procesamiento se utilizaron dos fotografías, en las que aparece el mismo ocelote pero en diferente posición.

### 3.1. Segmentación de la imagen

Por no ser el tema central de la tesis, solo se mostrará el resultado de la segmentación de la imagen. El desarrollo para la segmentación se puede ver en [Figuroa and Camarena, 2014]. En las figuras 3.1 y 3.2 se muestran las dos imágenes segmentadas con las que se trabajó.

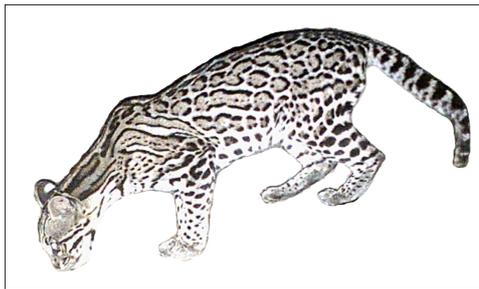


Figura 3.1: Ocelote en la fotografía 1.

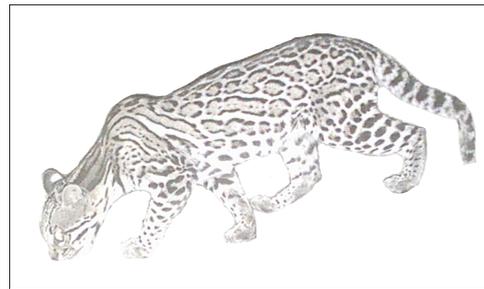
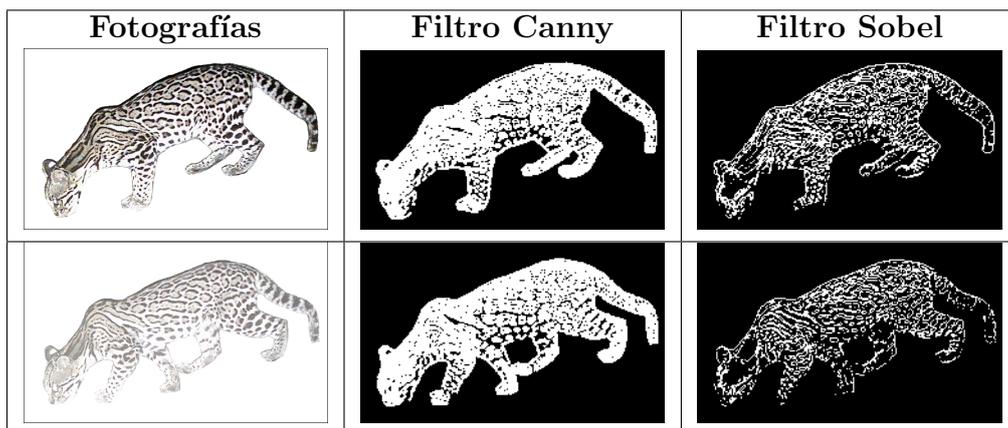


Figura 3.2: Ocelote en la fotografía 2.

### 3.2. Selección del filtro digital para las imágenes

En esta sección se analizará cual de los filtros presentados en el capítulo anterior marca de manera eficiente los contornos del animal, en particular, cada mancha. Los filtros se aplicaron utilizando la implementación desarrollada en la caja de herramientas de visión computacional de MATLAB [MathWorks, 1994a].

En el cuadro 3.1, podemos observar que una de las imágenes a filtrar está más iluminada que la otra (esto depende de la captura de la cámara, una de las características que presentan este tipo de imágenes y que pueden influir positiva o negativamente en el tratamiento posterior). Las siguientes dos columnas de este cuadro corresponden al resultado de aplicar el filtro de Canny a la imagen de la izquierda. La última de las columnas corresponde al resultado de aplicar el filtro de Sobel a la imagen de la primera columna.



Cuadro 3.1: Aplicación de los filtros de Canny y de Sobel.

Note que al aplicar el filtro de Canny es evidente a simple vista, que están bien definidos los bordes de ésta imagen, es decir, son más marcados que los de la otra. La iluminación parece no afectar al filtro de Canny, y en las dos imágenes se reconocen la mayoría de los bordes. Por otro lado, al aplicar el filtro de Sobel en ambas imágenes se observa que, en comparación con el de Canny la cantidad de manchas abiertas es mayor y se observan algunas partes completamente negras o en la imagen de la esquina inferior derecha no se marcan los bordes de las patas del animal, con lo cual la pérdida de información es evidente. De las observaciones anteriores se puede concluir que el filtro de Canny define con mayor precisión los bordes de las manchas

por lo que las manchas a caracterizar serán de las imágenes filtradas con éste.

### 3.3. Selección de áreas

Ya que se decide que filtro utilizar, el siguiente problema es el de elegir las manchas para su caracterización. El criterio tomado para seleccionar las manchas a procesar fue el de tomarlas de aquellas partes del animal que parecen estar fijas en ambas imágenes; si observamos las dos fotografías presentadas en el cuadro 3.1, se puede notar que las patas cambian de posición, lo cual hace que las manchas luzcan completamente diferentes de una imagen a otra en esas partes, además en la primer imagen se observan tres patas mientras que en la segunda se muestran cuatro. En cambio, si nos centramos en el torso podemos apreciar que se encuentra casi en la misma posición en las dos fotografías.

Una vez que se define el área de interés se emplea la función *imfill* de MATLAB para rellenar los contornos, una descripción de esta se puede consultar en [MathWorks, 1994b]. Con este proceso se tiene un indicador de las manchas que son cerradas en ambas imágenes y así se descartan las que parecieran serlo a simple vista. En la figura 3.3, algunos contornos parecen ser cerrados; pero no lo están, como los encerrados en los círculos azules; u otros que están cerrados en una imagen y en la otra no (los encerrados en los rectángulos rojos), aún así hay un número importante de manchas que coinciden en las dos imágenes, por ejemplo, las delimitadas con amarillo.

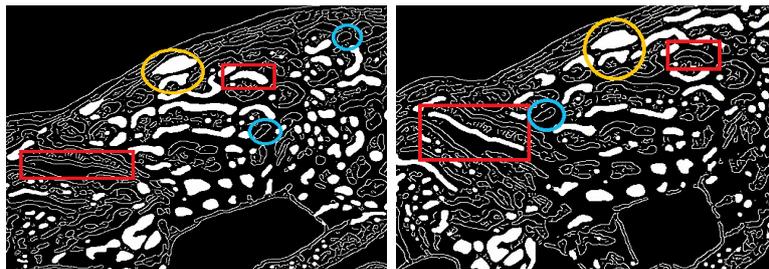


Figura 3.3: Contorno o perímetros de las manchas, rellenadas.

Con la función *bwareaopen* de MATLAB podemos eliminar algunas áreas y quedarnos con aquellas que representan las manchas rellenadas. En la figura 3.4 podemos ver que después de este proceso hay diferencias entre los torsos, o sea, las manchas que están marcadas con amarillo aparecen en una imagen y en la otra no.

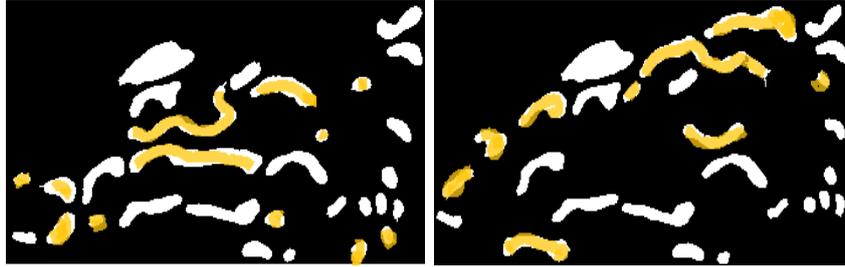


Figura 3.4: Manchas diferentes del torso después de que se rellenan las áreas.

Si se eliminan las manchas que son diferentes en los dos torsos quedan las 18 manchas de la figura 3.5 (por el momento la eliminación de estas manchas se realiza manualmente.)



Figura 3.5: Manchas iguales en las dos fotografías del torso.

Al elegir las manchas se decidió clasificarlas en regulares e irregulares, ya que algunas tienen formas muy parecidas a la elipse (en la figura 3.6 las manchas enmarcadas en color verde); sin embargo, hay otras que no, como las encerradas en los rectángulos de color amarillo. Es importante esta clasificación ya que esas características pueden impactar en los resultados de los dos métodos de caracterización.

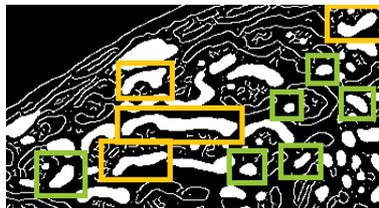
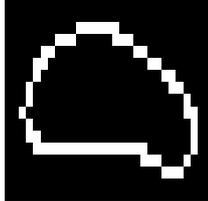
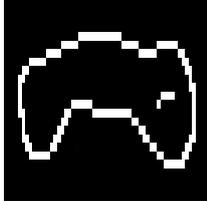
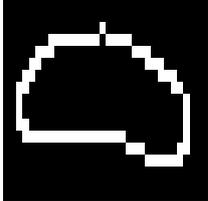
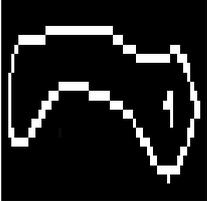
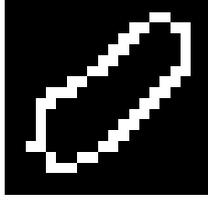
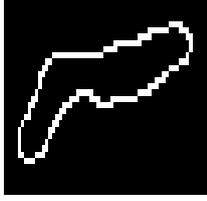
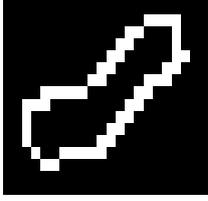
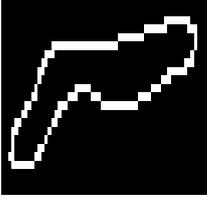
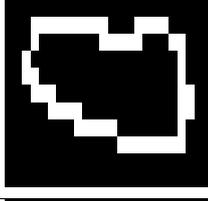
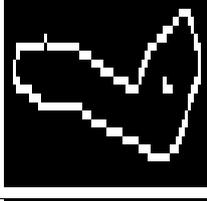
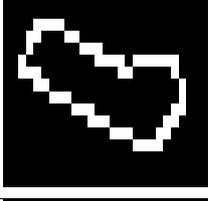
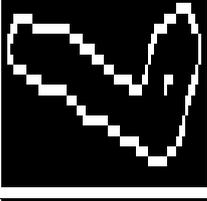
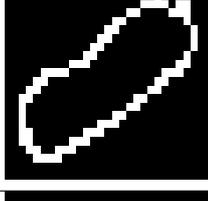
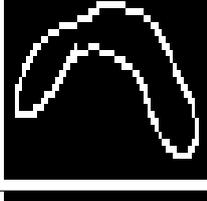
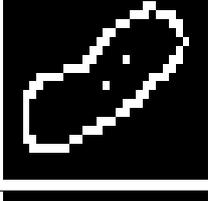
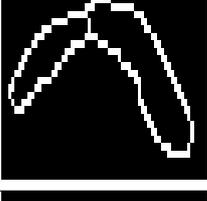
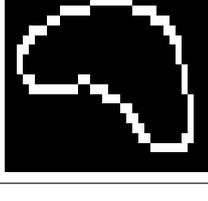
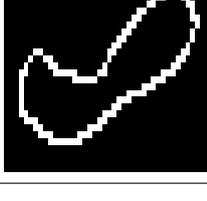
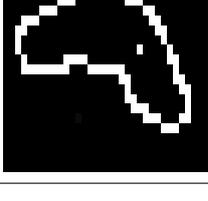
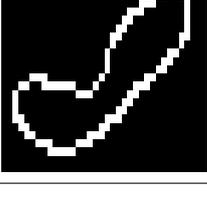


Figura 3.6: Manchas regulares e irregulares en el torso del animal.

### 3.4. Cálculo de las elipses en las manchas

Para realizar las pruebas se eligen 10 de las 18 manchas en cada una de las imágenes de la sección anterior, figura 3.5. En el cuadro 3.2 se pueden ver los dos grupos de manchas filtradas. Enumeraremos de arriba hacia abajo, las manchas regulares del 1 al 5 y las irregulares del 6 al 10.

Manchas regulares Foto 1	Manchas irregulares Foto 1	Manchas regulares Foto 2	Manchas irregulares Foto 2
			
			
			
			
			

Cuadro 3.2: Grupo de manchas.

### 3.4.1. Aplicación del método geométrico

Cuando se elige un número importante de manchas, el algoritmo para la función **construye** recibe las manchas ya filtradas y regresa los datos para cada elipse, es decir, el centro, el tamaño del semieje mayor, el tamaño del semieje menor (en píxeles) y el ángulo del semieje mayor (en radianes) respecto a la horizontal. Para simular la construcción geométrica sobre la mancha, se implementaron algunos códigos en MATLAB que puede consultar en el apéndice y que en resumen, hacen lo siguiente:

- **Cálculo del centro de la mancha**

En el código de la función **construyecentro**, (véase la figura 3.7) se grafican dos rectas paralelas ( $l_1$  y  $l_2$ ) sobre la imagen y se toma como punto de intersección, de éstas con la mancha, los puntos correspondientes a las coordenadas de los píxeles blancos por donde pasan dichas rectas, de esta manera se definen las cuerdas paralelas  $s$  ( $ab$ ) y  $t$  ( $cd$ ). Se toman los puntos medios de las cuerdas y se grafica la recta  $l_3$  que pasa por esos puntos, tomando nuevamente la “intersección” de la recta con la mancha, se define el diámetro  $D$  y así el punto medio de  $D$  corresponde al centro  $O$  de la elipse.

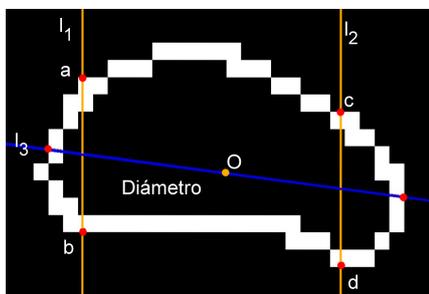


Figura 3.7: Construcción para encontrar el centro de la elipse.

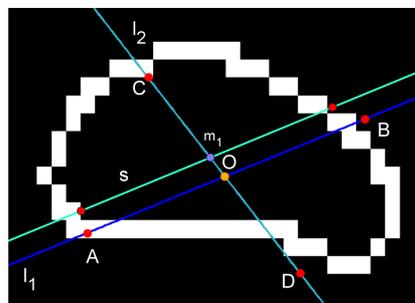


Figura 3.8: Construcción de los diámetros conjugados.

- **Diámetro conjugados**

Con las funciones, **dospuntosrectadiámetro**, **puntoscuerdaconjugada** y **asignarpuntos**, se definen dos diámetros conjugados de la siguiente manera: una vez que se obtiene el centro  $O$  se grafica una recta  $l_1$  que pase por  $O$ , se toman los puntos de intersección con la mancha y se obtiene el primer diámetro  $D1$  para obtener su conjugado se grafica una recta paralela a  $l_1$ , se toman los puntos de intersección y se obtiene la cuerda  $s$  paralela a  $D1$ . Se toma el punto medio de  $s$

( $m_1$ ) y se traza una recta  $l_2$  que pase por  $O$  y por  $m_1$  de esta forma ya construimos los dos diámetros conjugados sobre la mancha ( $AB$  y  $CD$ ), véase la figura 3.8.

- **Cálculo del tamaño y dirección de los semiejes**

Finalmente con la función **datoselipse** se determinan los tamaños de los ejes y la dirección de la elipse. En la figura 3.9 se muestran los siguientes trazos: si  $l_1$  es la recta que contiene al diámetro mayor  $AB$ , se traza una recta perpendicular  $l_p$  a ésta que pase por el centro  $ce$ . Para trasladar la distancia  $Ace$  se toma la ecuación del círculo  $C_0$  con centro en  $ce$  y radio  $r = Ace$  y se toman los puntos de intersección de  $l_p$  con  $C_0$ , es decir, las soluciones del sistema de sus ecuaciones  $ps_1$  y  $ps_2$ .

Para unir uno de estos puntos con  $C$ , extremo del diámetro conjugado  $CD$ , se elige el punto más cercano a  $C$ , suponiendo que ese punto es  $ps_1$ , se calcula el punto medio  $M$  del segmento  $ps_1C$  y con su distancia a  $C$  o a  $ps_1$ , se obtiene la ecuación del círculo  $C_1$ . Para obtener el tamaño de los semiejes basta resolver el sistema de ecuaciones del círculo  $C_1$  y la recta que pase por  $M$  y el centro de la elipse  $ce$ . Tomando la distancia de  $ce$  a las dos soluciones obtendremos lo deseado.

Partiendo de los puntos  $ce$ ,  $ps_1$  y  $M$ , se traza una recta que pase  $ps_1$  y  $M$ ; y, un círculo  $C_2$  con centro  $M$  y radio  $\|ceM\|$ . Tomando las soluciones del sistema de ecuaciones del círculo y la recta se obtienen dos puntos,  $X$  y  $Y$ , la dirección de un eje está dada por la pendiente de la recta que pasa por  $ce$  y  $X$ ; y, la otra por la pendiente de la recta que pasa por  $ce$  y  $Y$ .

Nota: el círculo  $C_0$  no se muestra en la imagen de la figura 3.9 ya que sólo sirve para obtener los puntos  $ps_1$  y  $ps_2$  que tampoco aparecen porque, en este caso, quedan fuera del dominio de la ventana generada por MATLAB, lo mismo pasa con el punto  $Y$ . Las rectas en beige tienen la misma pendiente que los semiejes.

## Resultados

Para realizar la comparación de las elipses definidas de los grupos de manchas del cuadro 3.2, se presenta el cuadro 3.3. En este se muestran los resultados para diez manchas de un ocelote en una fotografía y en el cuadro 3.4 los resultados para las manchas del mismo ocelote en una segunda fotografía.

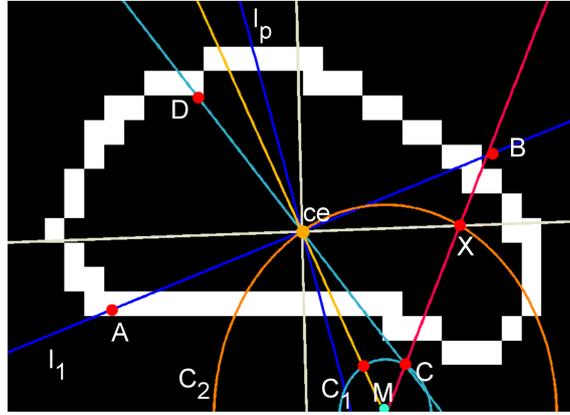


Figura 3.9: Construcción para los tamaños y direcciones de los semiejes.

Manchas	Centro	Semieje Mayor	Semieje Menor	Ángulo	Razón de los semiejes
Mancha 1	(15.4500,10.0592)	10.9212	6.3097	-0.0301	0.5777
Mancha 2	(10.9500,9.1854)	9.7114	3.3576	-0.5885	0.3457
Mancha 3	(12.4500,5.7469)	10.2842	3.6084	0.1616	0.3509
Mancha 4	(14.8000,12.5534)	14.1022	4.7040	-0.5749	0.3336
Mancha 5	(16.4500,11.1326)	14.8068	6.3433	0.2127	0.4284
Mancha 6	(29.4500,13.1935)	25.3685	4.7351	0.0239	0.1867
Mancha 7	(30.4500,15.0813)	25.3892	4.7703	-0.2302	0.1879
Mancha 8	(32.4500,12.4222)	28.2365	4.3286	0.1426	0.1533
Mancha 9	(28.9500,16.7990)	23.5161	7.6541	0.0037	0.3255
Mancha 10	(21.9500,14.1107)	18.5130	3.3894	-0.1992	0.1831

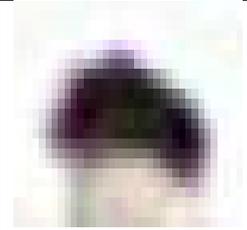
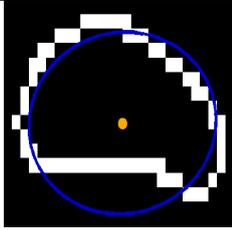
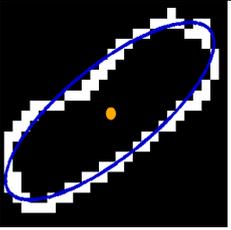
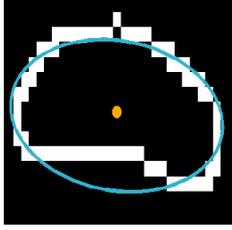
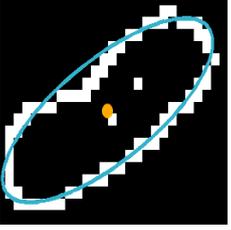
Cuadro 3.3: Resultados del método geométrico. Manchas del ocelote en la primer fotografía.

Manchas	Centro	Semieje Mayor	Semieje Menor	Ángulo	Razón de los semiejes
Mancha 1	(15.9500,9.4448)	13.8130	5.0415	0.0569	0.365
Mancha 2	(10.9500,8.7246)	9.5770	2.6771	-0.5199	0.2795
Mancha 3	(13.9500,8.2972)	12.2985	3.9972	0.2647	0.325
Mancha 4	(15.4500,12.2202)	13.6688	4.5540	-0.4918	0.3332
Mancha 5	(16.9500,9.9402)	13.7884	5.7120	0.1429	0.4143
Mancha 6	(29.9500,11.1474)	28.3911	3.5687	0.1005	0.1257
Mancha 7	(31.4500,11.1093)	26.9668	4.4388	-0.1388	0.1646
Mancha 8	(31.9500,9.3705)	28.4326	3.9399	0.2049	0.1386
Mancha 9	(31.9500,15.3405)	26.5867	7.6919	0.067	0.2893
Mancha 10	(18.4500,13.0021)	16.4927	2.9389	-0.4205	0.1782

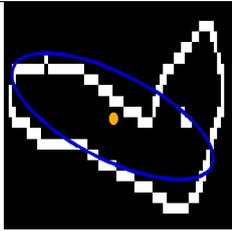
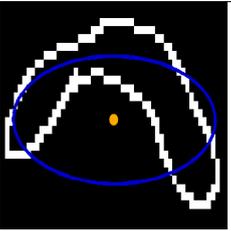
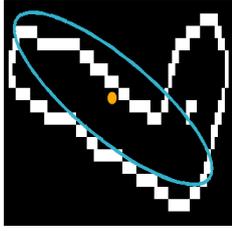
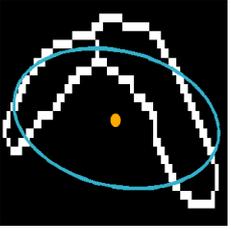
Cuadro 3.4: Resultados del método geométrico. Manchas del ocelote en la segunda fotografía.

Los cuadros también muestran en la última columna, la razón entre los ejes, para saber que tanto se conservan las proporciones de la misma mancha entre una fotografía y otra.

Para validar los resultados, se grafica la elipse resultante sobre cada mancha. En los cuadros, 3.5 y 3.6, se muestran manchas (1, 4, 8 y 9) extraídas directamente de las dos fotografías de las figuras 3.1 y 3.2. Y a la derecha de éstas, la elipse resultante. En particular en el cuadro 3.5 se puede ver el resultado para las manchas regulares 1 y 4 del cuadro 3.2, mientras que el cuadro 3.6 muestra resultados para las manchas irregulares 8 y 9 también del cuadro 3.2. Las manchas en la primera pertenecen a la foto de la figura 3.1 y las de la segunda fila a la foto en la figura 3.2. El punto en color amarillo es el centro de la elipse encontrada.

Mancha 1	Mancha Caracterizada	Mancha 4	Mancha Caracterizada
			
			

Cuadro 3.5: Resultados para las manchas regulares 1 y 4.

Mancha 8	Mancha Caracterizada	Mancha 9	Mancha Caracterizada
			
			

Cuadro 3.6: Resultados para las manchas irregulares 8 y 9.

En el cuadro 3.5 podemos ver como la elipse describe la forma del borde en la mayor parte de la mancha 4 y en algunas partes, el borde de la mancha 1. Algunas manchas como las del cuadro 3.6 podrían descartarse o bien, tratarse de otra manera ya que sus formas podrían dividirse en dos o más partes para su caracterización individual.

Con la finalidad de mostrar la relación de los resultados de los cuadros 3.3 y 3.4 se presentan a continuación tres gráficas de dispersión. En las gráficas de las figuras 3.10, 3.11 y 3.12 se muestran los resultados para los tamaños de los semiejes mayores, semiejes menores y ángulos respectivamente. En los ángulos se puede apreciar que, salvo la primera mancha, éstos coinciden en gran medida en las dos fotografías (en promedio la diferencia de los ángulos respecto a cada fotografía es de  $40,3115^\circ$ , debido a la gran diferencia en la primera mancha, si no admitimos ese dato la diferencia es de  $5,3444^\circ$ ). Para los tamaños de los semiejes mayores podemos ver que solo en las manchas 2, 4, 5 y 8 las longitudes se aproximan, aún así, existe una tendencia entre los tamaños de una foto y otra, es decir, de la primera a la segunda mancha la longitud decrece, de la segunda a la tercera crece y así sucesivamente (la diferencia promedio es de 1,6379 píxeles). En la gráfica de semiejes menores se sigue el mismo comportamiento en este caso la diferencia promedio es de 0,5494 píxeles.

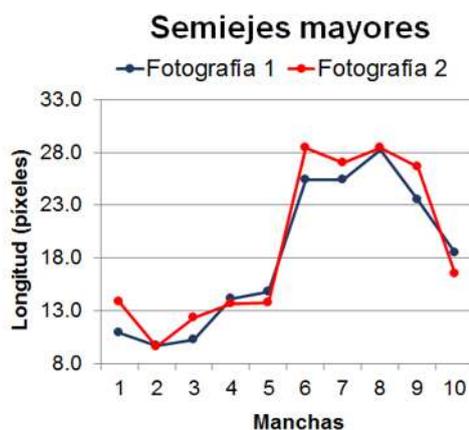


Figura 3.10: Semiejes mayores de las 10 manchas en las dos fotos.

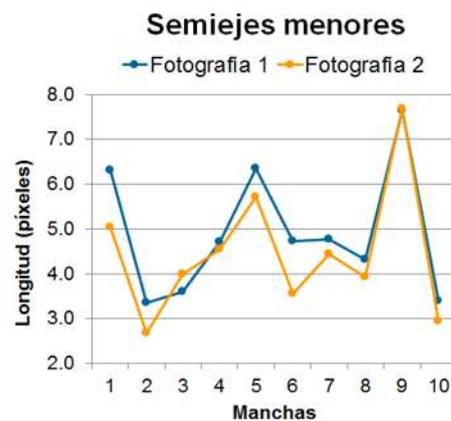


Figura 3.11: Semiejes menores de las 10 manchas en las dos fotos.

### 3.4.2. Momentos de la imagen

En la librería de MATLAB ya existe la implementación para la búsqueda de las propiedades de regiones en una imagen. En particular se encuentran las funciones *centroid* (centroide de la región), *MajorAxisLength* (longitud del eje mayor en píxeles), *MinorAxisLength* (longitud del eje menor en píxeles) y *Orientation* (ángulo entre el eje  $x$  y el eje mayor de la elipse). Las funciones se basan en la teoría mencionada en la sección 2, es decir,

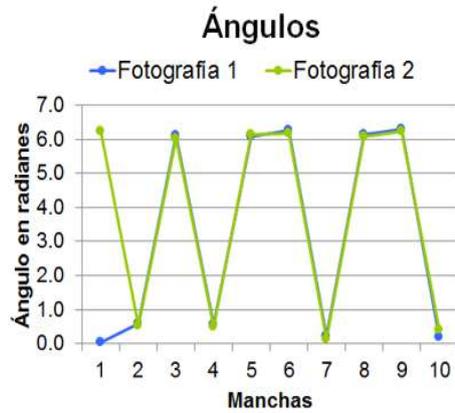


Figura 3.12: Ángulos de las 10 manchas en las dos fotos.

utilizando momentos de la imagen. En el apéndice A, en el algoritmo **momentosMATLAB**, se puede ver como se utilizan las funciones mencionadas anteriormente. O bien, puede consultar el algoritmo **momentos**, donde se utilizan las expresiones naturales de los momentos para el cálculo de los parámetros a encontrar.

## Resultados

Utilizando las mismas imágenes que en el método anterior (cuadro 3.2) se obtuvieron los resultados del cuadro 3.7 para manchas en la primera fotografía, y los del cuadro 3.8 para manchas en la segunda fotografía.

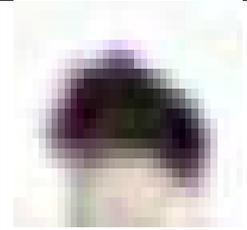
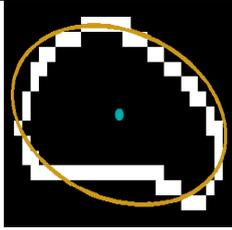
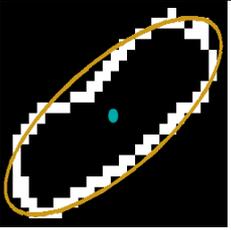
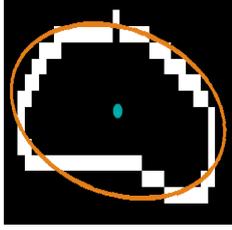
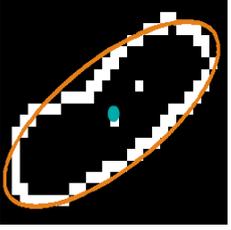
En el cuadro 3.9 se presentan los resultados para las manchas regulares 1 y 4 y en el cuadro 3.10 los resultados para las manchas irregulares 8 y 9. En ambos cuadros las manchas de la primer fila corresponden a la fotografía 1 y las de la segunda a la fotografía 2. El punto en azul es el centro de la elipse. Como se ve, las gráficas de las elipses para las manchas 1, 4 y 9 parecen idénticas en las dos fotografías; sin embargo, existe una diferencia entre las razones de los cuadros 3.7 y 3.8.

Manchas	Centro	Semieje Mayor	Semieje Menor	Ángulo	Razón de los semiejes
Mancha 1	(15.0950,9.1041)	12.8648	5.8060	-0.1374	0.4513
Mancha 2	(10.8871,10.1694)	9.9725	4.0860	0.7061	0.4097
Mancha 3	(13.0397,6.2381)	10.5828	3.9713	-0.1243	0.3753
Mancha 4	(14.8086,12.8047)	15.1620	5.5478	0.5562	0.3659
Mancha 5	(18.7509,10.6332)	15.5445	6.7761	-0.2324	0.4359
Mancha 6	(30.5387,11.6919)	29.4140	7.3952	-0.0166	0.2514
Mancha 7	(28.6597,15.0903)	29.3941	6.3851	0.2674	0.2172
Mancha 8	(37.6929,12.7482)	33.9568	7.2824	-0.0244	0.2145
Mancha 9	(30.8111,14.4415)	29.0773	10.2581	-0.1348	0.3528
Mancha 10	(21.8992,15.9395)	22.2088	7.0618	0.4061	0.318

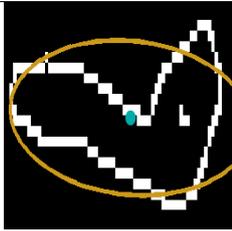
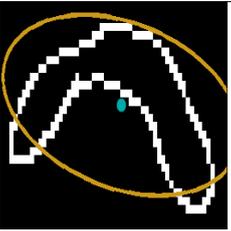
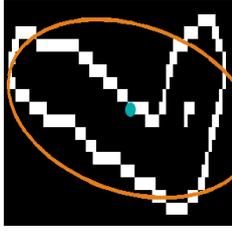
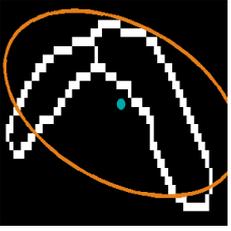
Cuadro 3.7: Resultados de los momentos de la imagen. Manchas del ocelote en la primer fotografía

Manchas	Centro	Semieje Mayor	Semieje Menor	Ángulo	Razón de los semiejes
Mancha 1	(16.2478,8.7743)	14.5894	5.2329	-0.0974	0.3587
Mancha 2	(11.2946,9.2321)	10.6324	3.6866	0.5443	0.3467
Mancha 3	(14.0216,8.9784)	12.0834	3.9973	-0.2485	0.3308
Mancha 4	(15.5938,12.4375)	14.3980	5.1158	0.483	0.3553
Mancha 5	(18.1070,9.0247)	15.5762	5.9579	-0.2466	0.3825
Mancha 6	(31.4432,10.1439)	33.7430	6.9037	-0.0744	0.2046
Mancha 7	(30.6339,10.8993)	31.9050	5.5418	0.1838	0.1737
Mancha 8	(37.4755,10.0352)	35.0520	6.8008	-0.0572	0.194
Mancha 9	(33.8411,13.1208)	31.8707	9.6376	-0.148	0.3024
Mancha 10	(18.8993,13.9676)	19.0921	5.8160	0.5038	0.3046

Cuadro 3.8: Resultados de los momentos de la imagen. Manchas del ocelote en la segunda fotografía.

Mancha 1	Mancha Caracterizada	Mancha 4	Mancha Caracterizada
			
			

Cuadro 3.9: Resultados de las manchas regulares 1 y 4.

Mancha 8	Mancha Caracterizada	Mancha 9	Mancha Caracterizada
			
			

Cuadro 3.10: Resultados de las manchas irregulares 8 y 9.

A comparación de los resultados del método geométrico las gráficas de dispersión de las figuras 3.13, 3.14 y 3.15 para los datos de los cuadros 3.7 y 3.8 nos muestran una relación más estrecha entre las manchas de las dos fotografías. Nuevamente la semejanza entre los ángulos es muy marcada (con una diferencia promedio de  $4,0027^\circ$ ) y más que la que existe entre los tamaños de los semiejes (la diferencia promedio para los semiejes mayores es

de 1,8526 píxeles y para los semiejes menores de 0,5206 píxeles). Al igual que en el método geométrico el comportamiento entre los semiejes se mantiene.

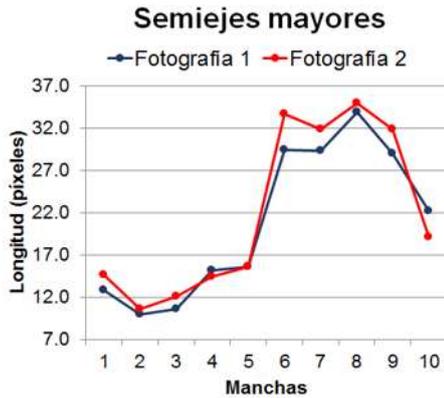


Figura 3.13: Semiejes mayores de las 10 manchas en las dos fotos.

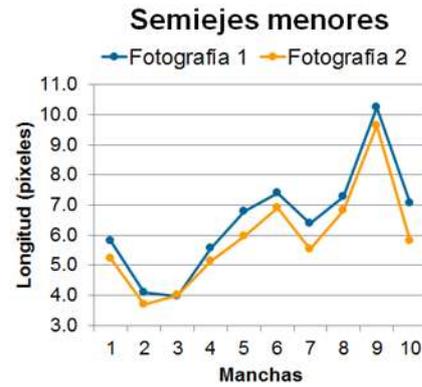


Figura 3.14: Semiejes menores de las 10 manchas en las dos fotos.

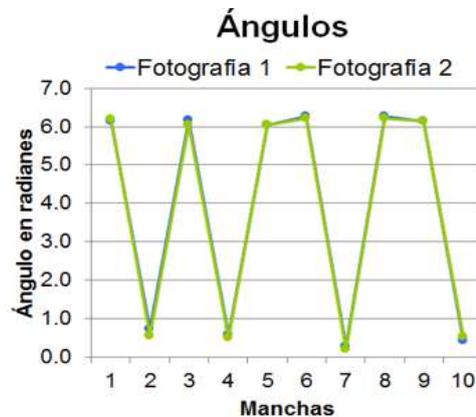


Figura 3.15: Ángulos de las 10 manchas en las dos fotos.

Como el algoritmo de **momentosMATLAB** puede calcular los parámetros en una fotografía con más de un objeto, el método se aplicó en el torso del animal de las imágenes de la figura 3.5.

En la gráficas de dispersión de las figuras 3.16 y 3.17 se puede notar que en la mancha 4 y 5 así como en las manchas 13, 14, 15, 16 y 18 hay variaciones evidentes. Si vemos la gráfica de la figura 3.18 éstas diferencias son más notorias en la dirección de las elipses.

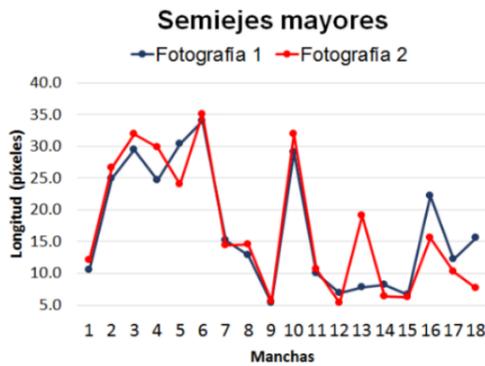


Figura 3.16: Semiejes mayores de las 18 manchas del torso.

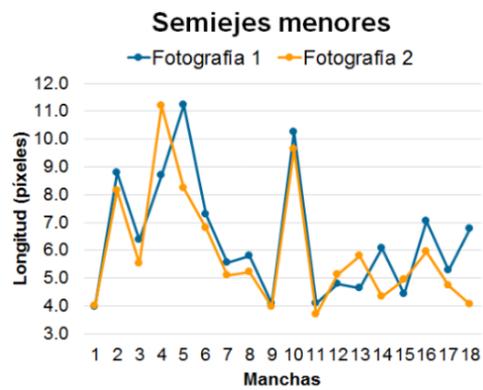


Figura 3.17: Semiejes menores de las 18 manchas del torso.

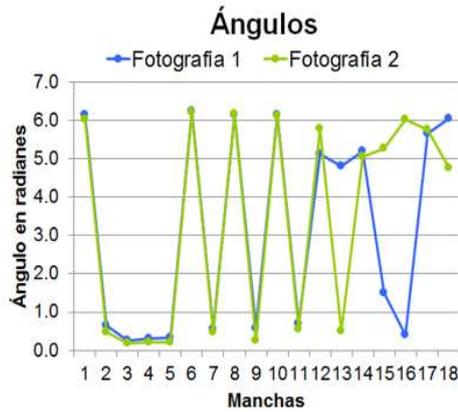


Figura 3.18: Ángulos de las 18 manchas del torso en las dos fotos.

Al analizar los datos se notó que la posición de los centroides de las manchas 4 y 5 cambian, es decir, en la foto de la izquierda de la figura 3.4 el centroide de la mancha 4 está a la izquierda del de la mancha 5 y en la foto de la derecha el centroide de la mancha 4 está a la derecha de la mancha 5. Lo mismo pasa con las manchas 13, 14, 15, 16 y 18 (el centroide de una cambia de posición respecto a las otras). Véanse las figuras 3.19 y 3.20, en las dos las manchas están enumeradas de acuerdo a la posición de su centroide.

Ya que nos dimos cuenta que la diferencia se debió a que se estaban comparando manchas diferentes, ordenando esos datos, las gráficas de las figuras 3.21, 3.22 y 3.23, nos muestran una relación más clara entre los parámetros de las manchas. En los ángulos, las manchas 12, 13 y 14 son diferentes; pero no tanto como los de la mancha 15. En promedio la diferencia en las

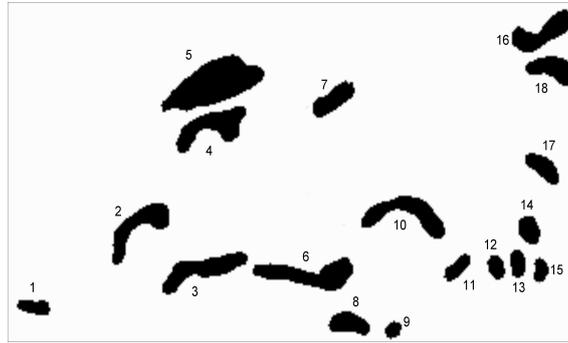


Figura 3.19: Manchas del torso en la foto 1.

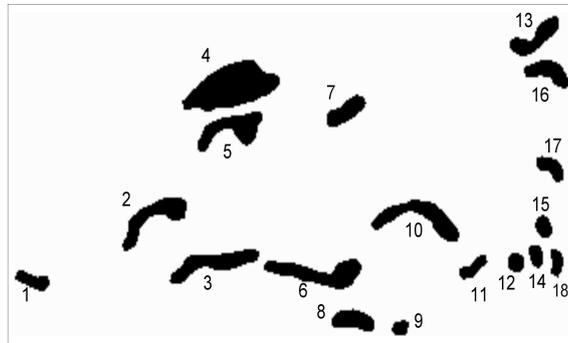


Figura 3.20: Manchas del torso en la foto 2.

longitudes de los semiejes es de 1,3997 píxeles para los mayores y de 0,5245 píxeles para los menores. En el caso de los ángulos el promedio es de  $18,0400^\circ$ , tomando en cuenta los resultados de la mancha 15, y de  $8,1000^\circ$  ignorándolos.

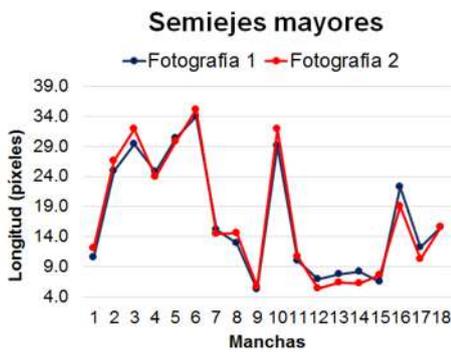


Figura 3.21: Semiejes mayores de las manchas en los dos torsos.

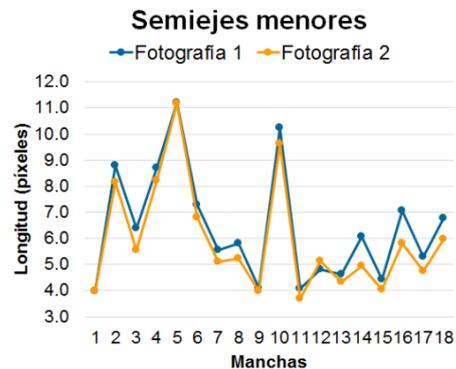


Figura 3.22: Semiejes menores de las manchas en los torsos.

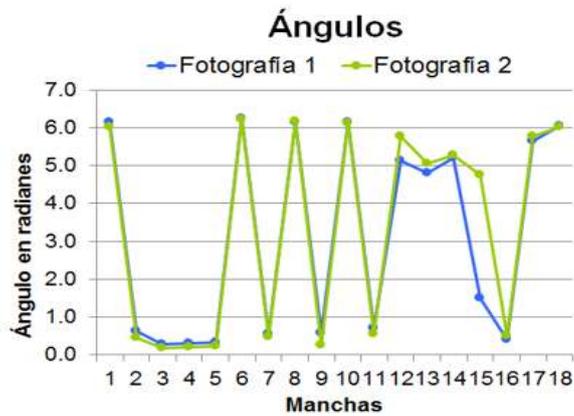


Figura 3.23: Ángulos de las manchas de los dos torsos.

En la figura 3.24 se puede ver la gráfica de dispersión para los centroides de las mismas 18 manchas en las dos fotografías del torso. Note que se tiene una correspondencia entre ambas manchas respecto a estos centroides. En promedio los centroides de la primer imagen están a una distancia de 21,1961 píxeles de los de la segunda.

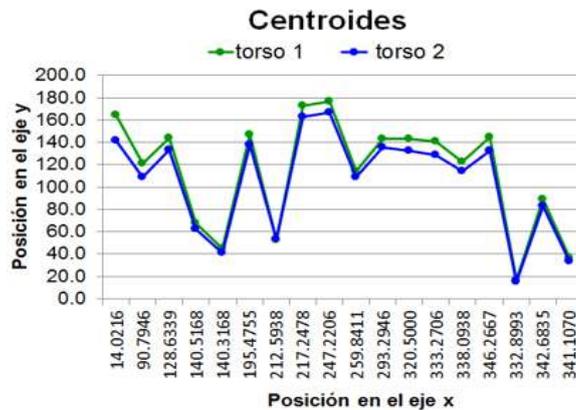


Figura 3.24: Centroides de las manchas en los dos torsos.

En las imágenes de la figura 3.25 se muestra el torso en las dos fotos con sus elipses correspondientes.

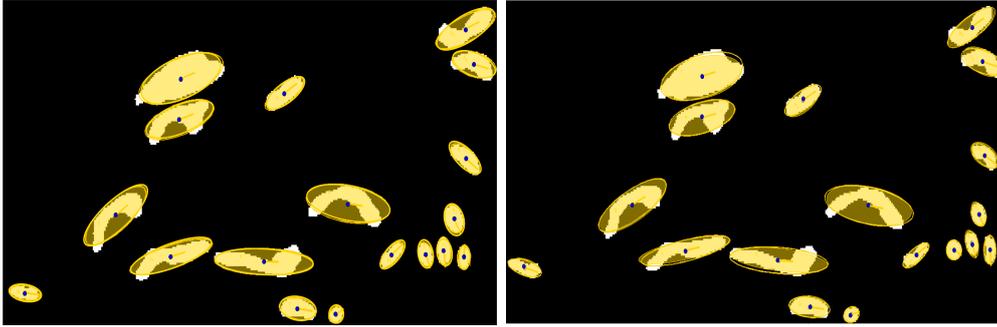


Figura 3.25: Torsos con su elipses correspondientes.

### 3.5. Comparación de los métodos

Para comparar los métodos, se muestran en las figuras 3.26 y 3.27, las gráficas de dispersión para los ángulos en los grupos de 10 manchas del cuadro 3.2. En la foto 1, la diferencia promedio para los ángulos entre los dos métodos es de 0,6809 radianes ( $39,0166^\circ$ ) y para la foto 2 la diferencia es de 0,05767 radianes ( $3,3042^\circ$ ), la gran diferencia en la foto 1 se debe al resultado para la mancha 1, si eliminamos ese dato el promedio sería de  $4,4181^\circ$  para la primer foto y de  $3,4135^\circ$  para la segunda.

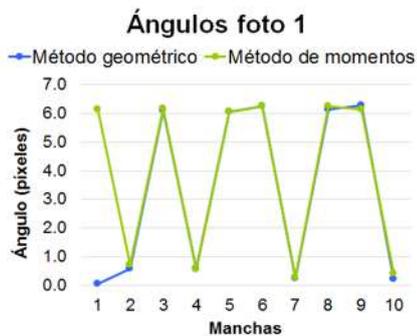


Figura 3.26: Ángulos de las 10 manchas en la foto 1.

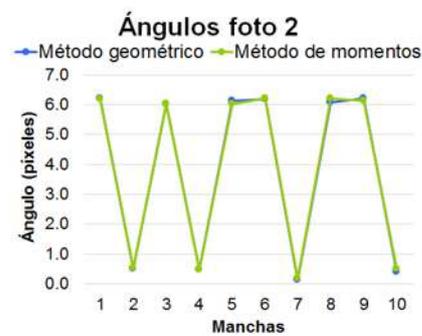


Figura 3.27: Ángulos de las 10 manchas en la foto 2.

En cuanto a los tamaños de los semiejes tenemos para los mayores las gráficas de las figuras 3.28 y 3.30 y para los menores las gráficas de las figuras 3.29 y 3.31; podemos notar que los del método de momentos son mayores que los del método geométrico. Otra cosa interesante es que en las primeras cinco manchas de la fotografía 1, si bien se nota una diferencia entre las longitudes de los semiejes (diferencia promedio de 0,8602 píxeles para los semiejes

mayores y de 0,5743 píxeles para los semiejes menores) ésta es menor que en las siguientes cinco manchas que clasificamos como irregulares (diferencia promedio de 4,6055 píxeles para los semiejes mayores y de 2,7010 píxeles para los semiejes menores).

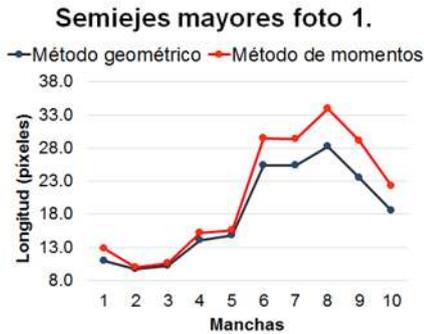


Figura 3.28: Semiejes mayores de las 10 manchas en la foto 1.

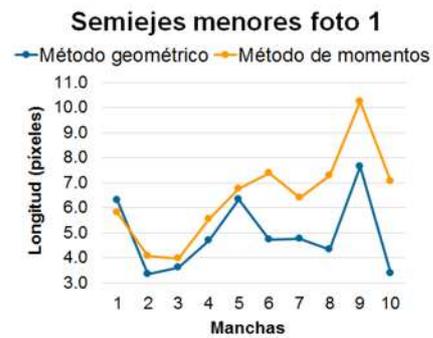


Figura 3.29: Semiejes menores de las 10 manchas en la foto 1.

En el caso de las manchas de la fotografía 2, la diferencia promedio entre semiejes mayores y menores para las primeras cinco manchas es de 0,9128 y 0,4017 píxeles respectivamente, y de 4,9585 y 2,4243 píxeles para las siguientes.

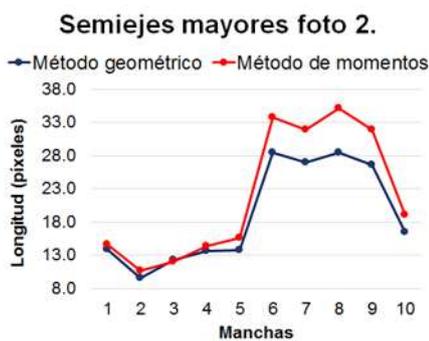


Figura 3.30: Semiejes mayores de las 10 manchas en la foto 2.

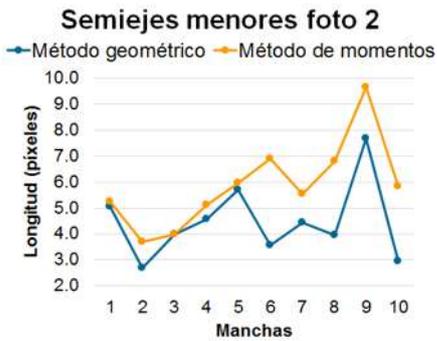


Figura 3.31: Semiejes menores de las 10 manchas en la foto 2.

Midiendo la cercanía entre los centros, o sea, la distancia entre ellos en la gráfica de la figura 3.32 se muestra que los centros de las manchas 1, 3, 4, 5, 7, y 8, en la primera fotografía, son parecidos en ambos métodos. Aún así los centros están alejados en promedio 1,9110 píxeles. En la foto dos, los

centros de las manchas 4 y 7 son los que parecen más cercanos, en este caso, el promedio de diferencia entre las distancias es de 1,5956 píxeles, figura 3.33.

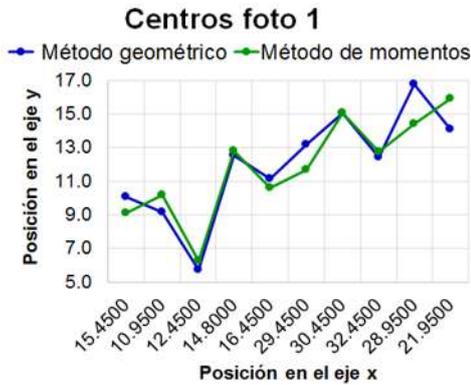


Figura 3.32: Centros de las 10 manchas en la foto 1.

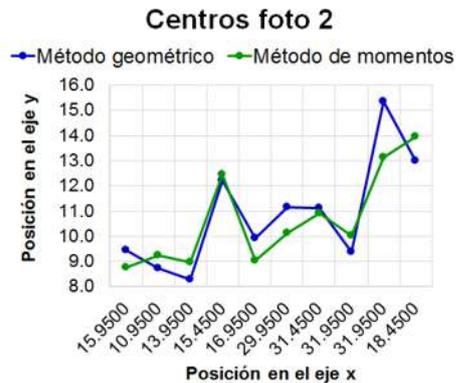


Figura 3.33: Centros de las 10 manchas en la foto 2.

Durante la experimentación, el filtro de Canny definió mejor los contornos del ocelote, especialmente en sus manchas.

Tanto en el método geométrico como en el de momentos, se obtuvieron diferentes resultados en las longitudes de los semiejes mayores, los semiejes menores y los ángulos entre una fotografía y otra; pero analizando las elipses sobre las manchas, el método de momentos caracteriza mejor los patrones del ocelote, además, éste puede trabajar con más de un objeto en la imagen. En cuanto al tiempo computacional, el algoritmo para el método geométrico requiere un tiempo de orden lineal mientras que el de momentos un tiempo de orden cuadrático, es decir, el método geométrico es más barato, sin embargo, como se utilizó la implementación de momentos de MATLAB, el algoritmo requiere menos tiempo que el geométrico.

# Capítulo 4

## Conclusión

De acuerdo a los resultados obtenidos se concluye lo siguiente:

El filtro que mejor define los contornos es el de Canny, que sin importar la iluminación de las fotografías, resalta la mayor parte de los bordes del ocelote. Y la aplicación de este filtro facilita la caracterización de los patrones.

Con los dos métodos se logra la caracterización de las manchas mediante elipses; sin embargo, el método de momentos aproxima mejor a cada patrón y tiene la gran ventaja de trabajar con más de un objeto en la imagen, otra cosa a favor es que requiere menos tiempo computacional.

Ya que en las direcciones de las elipses se obtuvieron buenos resultados y el comportamiento entre las posiciones de los centroides se mantiene, la mejor manera para caracterizar al animal es mediante estos dos parámetros. En la figuras 4.1 y 4.2 se muestran las gráficas de los puntos de los centroides y las direcciones de las manchas para los torsos de las figuras 3.19 y 3.20, respectivamente. Los vectores enumerados corresponden al grupo de las diez manchas con las que se estuvo trabajando.

La estimación numérica no se realizará en esta tesis ya que esta comparación se analiza con algoritmos para huellas digitales como en [Maltoni, 2005].

En las figuras, 4.3 y 4.4, se puede observar el resultado al aplicar el método a toda la imagen. Estas imágenes son muy interesantes pues como puede apreciarse hay un gran número de manchas que coinciden exactamente con las elipses y el programa las detecta perfectamente. Hay otras elipses, que si bien no corresponden con las manchas, tampoco afectarían en gran escala los resultados, pensando en algoritmos de búsqueda por similitud.

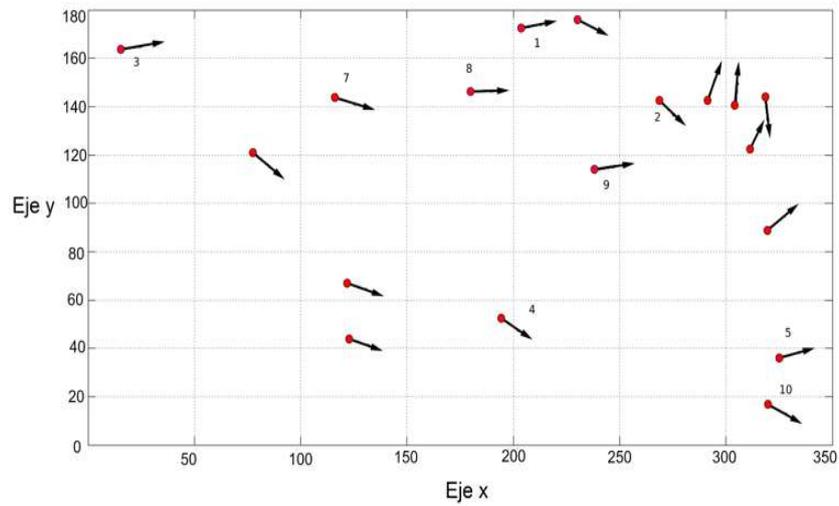


Figura 4.1: Centroides y vectores de las manchas caracterizadas en la foto 1.

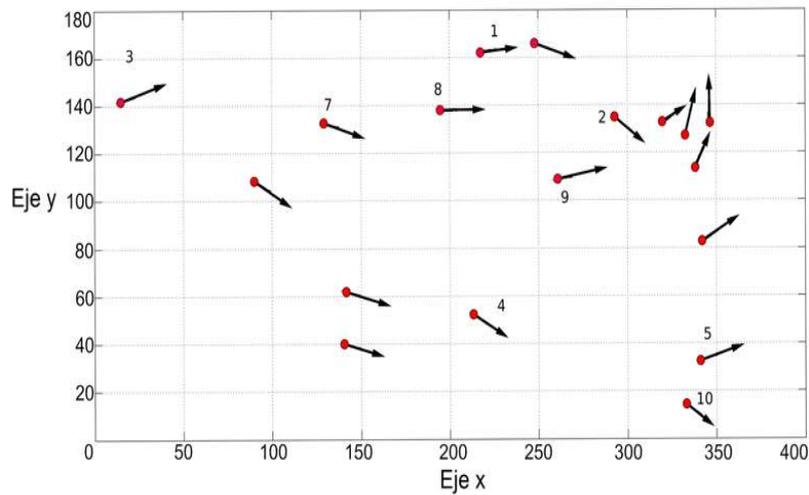


Figura 4.2: Centroides y vectores de las manchas caracterizadas en la foto 2.

## 4.1. Trabajo a futuro

- 1). Medir la similitud entre los grupos de vectores que representan a cada ocelote.
- 2). Extender la experimentación a otros animales silvestres con patrones elípticos.



Figura 4.3: Ocelote de la figura 3.1 caracterizado.

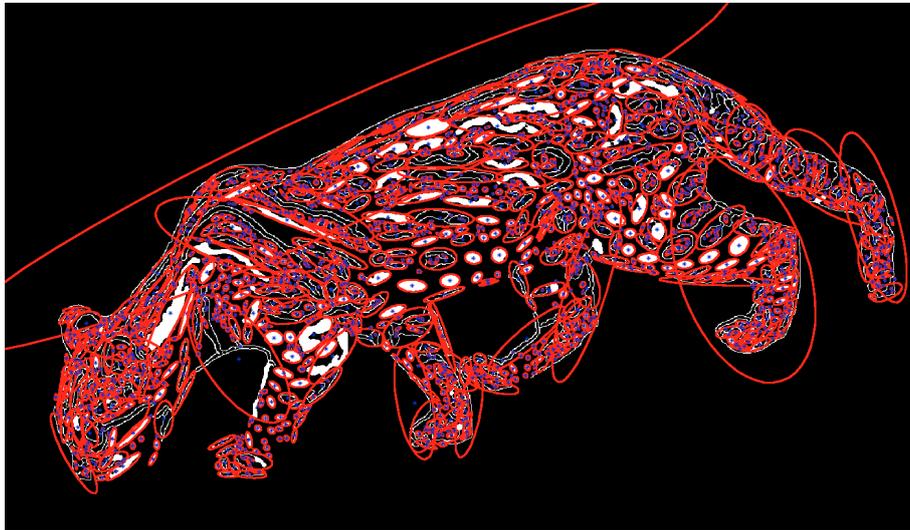


Figura 4.4: Ocelote de la figura 3.2 caracterizado.

# Apéndice A

## Implementación

1. Códigos para la construcción de las elipses.

```
1 function [ce,ejea,ejeb,angulo,razon]=construye(imagen)
2
3 image(imagen); %muestra la imagen
4 hold on;
5
6 syms x;
7
8 % tam de imagen
9 [ejey,ejex,ejj]=size(imagen);
10
11 % Dominio de x1
12 x1=0:.1:ejex;
13
14
15 % Centro
16 [ce]=construyecentro(imagen,ejex,ejey);
17
18 % cuerdas conjugadas
19 y1=(pi/3)*(x1-ce(1))+ce(2);
20 m1=(pi/3);
21
22 % Graficos para el primer diametro
23 f1='m1*(x-ce(1))+ce(2)';
24 plot(x1,y1,'r-');
25
26 % puntos del primer diametro
27 [A0,B0]=dospuntosrectadiámetro(imagen,ejex,m1,ce(1), ...
28 ce(2),ce);
29
30 % puntos para el segundo diametro
31 [C0,D0]=puntoscuerdaconjugada(imagen,ejex,m1,ce);
```

```

32
33 % diametros conjugados
34 [A,B,C,D]=asignarpuntos(A0,B0,C0,D0);
35
36 % ecuaciones de las rectas para los diametros conjugados
37 m2=(B(2)-A(2))/(B(1)-A(1));
38 y2=m2*(x1-B(1))+B(2);
39 m3=(D(2)-C(2))/(D(1)-C(1));
40 y3=m3*(x1-D(1))+D(2);
41
42 % Graficos para los diametros conjugados
43 graf1=plot(x1,y2,'linewidth',3);
44 set(graf1,'Color',[0,0,.98]);
45 graf2=plot(x1,y3,'linewidth',3);
46 set(graf2,'Color',[.2,.7,.8]);
47 graf3=plot(ce(1),ce(2),'*');
48 set(graf3,'Color',[1,.75,0]);
49 plot(A(1),A(2),'r*');
50 plot(B(1),B(2),'r*');
51 plot(C(1),C(2),'r*');
52 plot(D(1),D(2),'r*');
53
54 % funcion que determina los ejes y el angulo
55 function [ejea,ejeb,angulo]=datoselipse(ejex,m2,A,C,ce);
56
57 razon=ejeb/ejea;
58 hold off;
59 elipsefinal(imagen,ce,ejea,ejeb,-angulo);

```

```

1 function [ce]=construyecentro(imagen,ejex,ejey)
2
3 image(imagen);
4 hold on
5 x1=0:.1:ejex;
6
7 % Rectas iniciales para las cuerdas paralelas
8 y01=ejex/5;
9 y02=4*y01;
10 y1=y01;
11 y2=y02;
12
13 % Puntos de interseccion de las rectas con la mancha
14 [p1,p2]=dospuntosrectapendiente(imagen,ejex,0,0,y01);
15 [p3,p4]=dospuntosrectapendiente(imagen,ejex,0,0,y02);
16
17
18 % Puntos medios de las cuerdas

```

```

19 pm1=[(p1(1)+p2(1))/2,(p1(2)+p2(2))/2];
20 pm2=[(p3(1)+p4(1))/2,(p3(2)+p4(2))/2];
21
22 % Recta que pasa por los puntos medio de las cuerdas
23 m=(pm2(2)-pm1(2))/(pm2(1)-pm1(1));
24 y3=m*(x1-pm1(1))+pm1(2);
25
26
27 % Puntos de interseccion de la recta anterior con la ...
    mancha
28 [p5,p6]=dospuntosrectapendiente(imagen,ejex,m, ...
29 pm1(1),pm1(2));
30
31
32 % Centro de la elipse
33 ce=[(p5(1)+p6(1))/2;(p5(2)+p6(2))/2];
34
35 % Grafica de las rectas y los puntos
36 c1=plot(y1,x1,'linewidth',3);
37 c2=plot(y2,x1,'linewidth',3);
38 plot(p1(1),p1(2),'r*');
39 plot(p2(1),p2(2),'r*');
40 plot(p3(1),p3(2),'r*');
41 plot(p4(1),p4(2),'r*');
42 plot(pm1(1),pm1(2),'b*');
43 plot(pm2(1),pm2(2),'b*');
44 c3=plot(x1,y3,'linewidth',3);
45 plot(p5(1),p5(2),'r*');
46 plot(p6(1),p6(2),'r*');
47 d9=plot(ce(1),ce(2),'*');
48 set(d9,'Color',[1,.75,0]);
49 set(c1,'Color',[1,0.5,.1]);
50 set(c2,'Color',[1,0.5,.1]);
51 set(c3,'Color',[0,0,.98]);
52 hold off;

```

```

1 function ...
    [p1,p2]=dospuntosrectapendiente(imagen,ejex,m,x0,y0)
2
3 cont=1;
4 f='m*(x-x0)+y0';
5
6 for x=0:.1:ejex
7     yf=eval(f);%valor de f en x
8     pixel=impixel(imagen,yf,x);%pixel sobre f1
9
10    if pixel(1) == 255,

```

```

11         M(cont,1)=yf;
12         M(cont,2)=x;
13         cont=cont+1;
14         %plot(yf,x,'r*');
15     end
16
17 end
18 M;
19 TM=length(M);
20 p1=[M(1,1),M(1,2)];
21 p2=[M(TM,1),M(TM,2)];

```

```

1 function ...
   [p1,p2]=dospuntosrectapendiente(imagen,ejex,m,x0,y0)
2
3 cont=1;
4 f='m*(x-x0)+y0';
5
6 for x=0:.1:ejex
7     yf=eval(f);%valor de f en x
8     pixel=impixel(imagen,x,yf);%pixel sobre f1
9
10    if pixel(1) == 255,
11        M(cont,1)=x;
12        M(cont,2)=yf;
13        cont=cont+1;
14        %plot(x,yf,'r*');
15    end
16
17 end
18 M;
19 TM=length(M);
20 p1=[M(1,1),M(1,2)];
21 p2=[M(TM,1),M(TM,2)];
22 %plot(p1(1),p1(2),'b*');
23 %plot(p2(1),p2(2),'b*');

```

```

1 function ...
   [p1,p2]=dospuntosrectadiámetro(imagen,ejex,m,x0,y0,ce)
2
3 cont=1;
4
5 % Ecuacion de la recta que interseca la mancha
6 f='m*(x-x0)+y0';
7
8 % Puntos blancos por donde pasa la recta

```

```

9 for x=0:.1:ejex
10     yf=eval(f); %valor de f en x
11     pixel=impixel(imagen,x,yf); %pixel sobre f1
12
13     if pixel(1) == 255,
14         M(cont,1)=x;
15         M(cont,2)=yf;
16         cont=cont+1;
17     end
18 end
19
20 M; % Matriz de puntos blancos
21 TM=length(M);
22
23 % Primer y ultimo puntos de la matriz M
24 s1=[M(1,1),M(1,2)];
25 s2=[M(TM,1),M(TM,2)];
26
27 % Distancia de los puntos al centro
28 d1=norm(ce'-s1);
29 d2=norm(ce'-s2);
30
31 % promedio de las distancias d1 y d2
32 r=(d1+d2)/2;
33
34 % puntos para determinar los diametros
35 [p1,p2]=interrecic(ce,m,r);
36
37 % grafica de los puntos
38 plot(p1(1),p1(2),'b*');
39 plot(p2(1),p2(2),'b*');

```

```

1 function [in1,in2]=interrecic(ce,m,r1)
2
3 % Ecuacion circulo y recta
4 syms xx yy mt x0a y0a c1 c2 r;
5 circ='(xx-c1)^2+(yy-c2)^2=r^2';
6 recta='yy=mt*(xx-x0a)+y0a';
7
8
9 % Sustitucion de valores
10 circ=subs(circ,{c1,c2,r},{ce(1),ce(2),r1});
11 recta=subs(recta,{mt,x0a,y0a},{m,ce(1),ce(2)});
12
13
14 % Solucion interseccion
15 [xin1,yin2]=solve(circ,recta);

```

```

16 s1=double(xin1(1));
17 s2=double(xin1(2));
18 s3=double(yin2(1));
19 s4=double(yin2(2));
20 in1=[s1,s3];
21 in2=[s2,s4];
22
23
24 % Grafica de los puntos
25 plot(in1(1),in1(2),'r*');
26 plot(in2(1),in2(2),'r*');

```

```

1 function ...
   [p1,p2]=dospuntosrectadiámetro(imagen,ejex,m,x0,y0,ce)
2
3 cont=1;
4 % Ecuacion de la recta que interseca la mancha
5 f='m*(x-x0)+y0';
6
7 % Puntos blancos por donde pasa la recta
8 for x=0:.1:ejex
9     yf=eval(f); %valor de f en x
10    pixel=impixel(imagen,x,yf); %pixel sobre f1
11
12    if pixel(1) == 255,
13        M(cont,1)=x;
14        M(cont,2)=yf;
15        cont=cont+1;
16        %plot(x,yf,'r*');
17    end
18 end
19
20 M; % Matriz de puntos blancos
21 TM=length(M);
22
23 % Primer y ultimo puntos de la matriz M
24 s1=[M(1,1),M(1,2)];
25 s2=[M(TM,1),M(TM,2)];
26
27 % Distancia de los puntos al centro
28 d1=norm(ce'-s1);
29 d2=norm(ce'-s2);
30
31 % promedio de las distancias d1 y d2
32 r=(d1+d2)/2;
33
34 % puntos para determinar los diametros

```

```

35 [p1,p2]=interrecic(ce,m,r);
36
37 % grafica de los puntos
38 plot(p1(1),p1(2),'b*');
39 plot(p2(1),p2(2),'b*');

```

```

1 function [C,D]=puntoscuerdaconjugada(imagen,ejex,m1,ce)
2
3 % puntos de interseccion de la cuerda paralela al ...
  diametro AB
4 [i1,i2]=dospuntosrectapendiente(imagen,ejex,m1,ce(1)+ ...
5 (ejex/4),ce(2));
6
7 % punto medio de la cuerda paralela a AB
8 mil=[(i1(1)+i2(1))/2,(i1(2)+i2(2))/2];
9
10 % pendiente de la recta que pasa por mil y ce
11 m3=(mil(2)-ce(2))/(mil(1)-ce(1));
12
13 % puntos que determinan el diametro CD
14 [C,D]=dospuntosrectadiámetro(imagen,ejex,m3,mil(1), ...
15 mil(2),ce);

```

```

1 function [A,B,C,D]=asignarpuntos(p1,p2,p3,p4)
2
3 % Eleccion del diametro mayor y el diametro menor
4 tamejea=norm(p2-p1);
5 tamejeb=norm(p3-p4);
6 if tamejea>tamejeb
7     A=[p1(1),p1(2)];
8     B=[p2(1),p2(2)];
9
10    C=[p3(1),p3(2)];
11    D=[p4(1),p4(2)];
12 else
13
14    A=[p3(1),p3(2)];
15    B=[p4(1),p4(2)];
16
17    C=[p1(1),p1(2)];
18    D=[p2(1),p2(2)];
19 end

```

```

1 function [ejea,ejeb,angulo]=datoselipse(ejex,m2,A,C,ce)

```

```

2
3 x1=0:.1:ejex;
4 hold on;
5
6 % radio del primer circulo y pendiente de la ...
   perpendicular de AB
7 r1=norm(A-ce');
8 mt1=-(1/m2);
9 y1=mt1*(x1-ce(1))+ce(2);
10 graf1=plot(x1,y1,'linewidth',3);
11 set(graf1,'Color',[0,0,.98]);
12
13 % Inerseccion del circulo r=ACe recta perpendicular a AB
14 [M1,M2]=interrecic(ce,mt1,r1);
15 d1=norm(M1-C);
16 d2=norm(M2-C);
17 % Punto M
18 if d1<d2
19     M=[M1(1),M1(2)];
20 else
21     M=[M2(1),M2(2)];
22 end
23
24
25 % centro del circulo para determinar el tamaño de ...
   los ejes
26 O1=[(C(1)+M(1))/2,(C(2)+M(2))/2];
27 plot(O1(1),O1(2),'b*');
28
29 % circulo O1M
30 r2=norm(M-O1);
31
32 xc1=O1(1); yc1=O1(2); rc1=r2;
33 n = 50; k=0:1:n;
34 fi=2*pi*k/n;
35 xcir1=xc1+rc1*cos(fi); ycir1=yc1+rc1*sin(fi);
36 c=plot(xcir1,ycir1,'linewidth',3) ;
37 set(c,'Color',[.2,.7,.8]);
38
39 % Recta CeO1
40 m1=(ce(2)-O1(2))/(ce(1)-O1(1));
41 y2=m1*(x1-ce(1))+ce(2);
42 graf2=plot(x1,y2,'linewidth',3) ;
43 set(graf2,'Color',[1,.75,0]);
44
45
46 % Interseccion del circulo O1M y recta ce O1
47 [R,L]=interrecic(O1,m1,r2);
48

```

```

49 % Eje mayor y eje menor
50 a=norm(ce'-L);
51 b=norm(ce'-R);
52
53 if a>b
54     ejea=a;
55     ejeb=b;
56 else
57     ejea=b;
58     ejeb=a;
59 end
60
61 % Circulo de radio O1Ce centro en O1 y pendiente ...
    recta MO1
62 r3=norm(O1-ce');
63 m2=(M(2)-O1(2))/(M(1)-O1(1));
64 y3=m2*(x1-M(1))+M(2);
65 graf3=plot(x1,y3,'linewidth',3) ;
66 set(graf3,'Color',[1,0,.3]);
67
68
69 xc=O1(1); yc=O1(2); rc=r3;
70 xcir=xc+rc*cos(fi); ycir = yc+rc*sin(fi);
71 c=plot(xcir,ycir,'linewidth',3) ;
72 set(c,'Color',[1,.5,0]) ;
73
74
75 % Interseccion de la recta y el circulo anterior
76 [N,S]=interrecic(O1,m2,r3);
77
78 dis1=norm(ce'-N);
79 dis2=norm(ce'-S);
80 if dis1<dis2
81     pX=[N(1), N(2)];
82     pX1=[S(1), S(2)];
83 else
84     pX=[S(1), S(2)];
85     pX1=[N(1), N(2)];
86 end
87
88 % angulo
89 mfina1=(ce(2)-pX(2))/(ce(1)-pX(1));
90 yx=mfina1*(x1-pX(1))+pX(2);
91 graf4=plot(x1,yx,'linewidth',3) ;
92 set(graf4,'Color',[.9,.9,.8]);
93
94 mfina11=(ce(2)-pX1(2))/(ce(1)-pX1(1));
95 yx1=mfina11*(x1-pX1(1))+pX1(2);
96 graf5=plot(x1,yx1,'linewidth',3) ;

```

```

97 set (graf5, 'Color', [.9, .9, .8]);
98
99
100 plot (M(1),M(2), 'r*');
101
102 angulo=atan(mfinal);

```

```

1 function elipsefinal (imagen, centro, semiejemayor, ...
2 semiejemenor, angulo)
3
4 image (imagen);
5 hold on;
6 set (gcf, 'NextPlot', 'add');
7
8 theta=linspace (0,2*pi,100);
9 xx=semiejemayor*cos (theta);
10 yy=semiejemenor*sin (theta);
11 t=angulo; M=[cos (t) sin (t);-sin (t) cos (t)];
12 R=M*[xx;yy];
13 g1=plot (R(1,:),centro(1),R(2,:),centro(2), 'linewidth',4);
14 set (g1, 'Color', [.2, .7, .8]);
15 g2=plot (centro(1),centro(2), '*');
16 set (g2, 'Color', [1, .75, 0]);
17 hold off;

```

## 2. Códigos para el cálculo de momentos.

```

1 function [centro,a,b,theta,r]=momentosMATLAB (CC)
2 % CC es la imagen
3 image (CC);
4 hold on;
5
6 % Binarizar la imagen
7 bw = im2bw (CC, graythresh (getimage));
8
9 % Rellenar los huecos
10 bw2 = imfill (bw, 'holes');
11
12 % eliminar areas de 60 pixeles
13 bw3=bwareaopen (bw2, 60, 4);
14 hold on;
15 % propiedades que quiero encontrar en la imagen bw3
16 s=regionprops (bw3, 'centroid', 'MajorAxisLength', ...
17 'MinorAxisLength', 'Orientation');
18
19

```

```

20  for k = 1:length(s)
21
22      % Coordenadas del centroide
23      xbar = s(k).Centroid(1)
24      ybar = s(k).Centroid(2)
25
26      % Tamanios de los semiejes
27      a = s(k).MajorAxisLength/2;
28      b = s(k).MinorAxisLength/2;
29
30      % angulo en radianes
31      theta = pi*s(k).Orientation/180;
32
33  end
34
35  centro=[xbar,ybar];
36
37  % razon de los ejes
38  r=b/a;
39  hold off;
40
41  % Grafica de la elipse
42  ellipsefinal1(CC,centro,a,b,theta);

```

```

1  function [ce,ejea,ejeb,theta,r]=momentos(im)
2
3  image(im);
4  hold on;
5
6  % Imagen blanco y negro (ceros y unos)
7  bw = im2bw(im, graythresh(getimage));
8  % Rellenando huecos
9  bw2 = imfill(bw, 'holes');
10 imshow(bw2);
11 % eliminar areas de 60 pixeles
12 bw3=bwareaopen(bw2,60,4);
13 hold on;
14
15 % Coordenadas de pixeles blancos de la imagen (la mancha)
16 [Mx,My]=puntoselipse(bw3);
17
18 O=ones(size(Mx));
19 % Momentos de la imagen
20 m00=length(Mx);
21 m10=Mx'*O;
22 m01=My'*O;
23 m11=Mx'*My;

```

```

24     m20=Mx'*Mx;
25     m02=My'*My;
26
27     % centroide
28     xc=m10/m00;
29     yc=m01/m00;
30     ce=[xc,yc];
31
32     % Ejes
33     a=m20-((xc^2)*m00);
34     b=m02-((yc^2)*m00);
35     c=m11-(xc*yc*m00);
36
37     I1=((a+b)+sqrt((a-b).^2+(4*(c.^2))))/2;
38     I2=((a+b)-sqrt((a-b).^2+(4*(c.^2))))/2;
39
40     ejea=2*sqrt(I1/m00);
41     ejeb=2*sqrt(I2/m00);
42
43
44     % Angulo
45     theta=(atan(2*c/(a-b)))/2;
46
47
48     % Razon
49     r=ejeb/ejea;
50
51     hold off;
52     elipsefinal(im,ce,ejea,ejeb,-theta);

```

# Bibliografía

- [Canny, 1986] Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 8, pages 679–714.
- [Charre et al., 2013] Charre, J., Monterrubio, T., Botello, F., León, L., and Núñez, R. (2013). First records of jaguar (panthera onca) from the state of michoacán, méxico. *BioOne Research Evolved*, pages 264–268.
- [Espinosa, 2013] Espinosa, J. (2013). Juan carlos espinosa ceniceros.
- [Figueroa and Camarena, 2014] Figueroa, K. and Camarena, J. (2014). Segmentation. *CIARP*.
- [Hartog and Reijns, 2014] Hartog, J. and Reijns, R. (2014). *I<sup>3</sup>S PATTERN MANUAL. Intractive Individual Identification System*.
- [HU, 1962] HU, M.-K. (1962). Visual pattern recognition by moment invariants. *IRE TRANSACTIONS ON INFORMATION THEORY*, pages 179–187.
- [Kelly, 2001] Kelly, M. J. (2001). Computer-aided photograph matching in studies using individual identification: An example from serengeti cheetahs. *Journal of Mammalogy*, 82(2):440–449.
- [Kuhl, 1982] Kuhl, F. (1982). Elliptic fourier features of a closed contour. *Computer Graphics and Image Processin* 18, pages 236–258.
- [Lehmann, 1989] Lehmann, C. (1989). *Geometría Analítica*, page 173. LIMUSA.
- [Maltoni, 2005] Maltoni, D. (2005). A tutorial in fingerprint recognition.
- [MathWorks, 1994a] MathWorks, T. (1994a). Edge.
- [MathWorks, 1994b] MathWorks, T. (1994b). Imfill.

- [OpenCV, 2011a] OpenCV, . (2011a). Canny edge detector.
- [OpenCV, 2011b] OpenCV, . (2011b). Sobel derivatives.
- [Pacheco, 2015] Pacheco, E. (2015). Identificación de diatomeas del lago de pátzcuaro mediante filtros compuestos no lineales y caracterización de las valvas utilizando descriptores geométricos. Master's thesis, Universidad Michoacana de San Nicolás de Hidalgo.
- [Pertusa, 2003] Pertusa, J. (2003). *Técnicas de análisis de imagen. Aplicaciones en Biología*. Publicacions de la Universitat de València.
- [Pérez, 2010] Pérez, J. M. (2010). Segmentación de imágenes.
- [Ramírez, 2006] Ramírez, B. (2006). *Apuntes del curso impartido por el Dr. Boris Escalante Ramírez*.
- [Rocha et al., 2004] Rocha, L., Velho, L., and Carvallido, P. P. (2004). Motion reconstruction using moments analysis. *Computer Graphics and Image Processing*.
- [Sobel and Feldman, 1973] Sobel, I. and Feldman, G. (1973). A  $3 \times 3$  isotropic gradient operator for image processing. *R. Duda and P. Hart (Eds.), Pattern Classification and Scene Analysis*.
- [Whitehead et al., 2000] Whitehead, H., Christal, J., and Tyack, P. (2000). Studying cetacean social structure in space and time. *Cetacean Societies: Field Studies of Dolphins and Whales*.