



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

**“ESTIMACIÓN AUTOMÁTICA DE PARÁMETROS Y
SEGMENTACIÓN ESTOCÁSTICA EN PERCEPCIÓN
REMOTA”**

T E S I S

QUE PARA OBTENER EL GRADO DE:

**MAESTRO EN CIENCIAS
(COMPUTACIÓN)**

P R E S E N T A:

DAVID ISRAEL MORALES AVILA

DIRECTOR DE TESIS:

DR. MIGUEL MOCTEZUMA FLORES

México, D.F.

2007.



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Agradecimientos

A mis padres por su gran apoyo, cariño y comprensión a lo largo de mi vida, así como por los sabios consejos que me brindaron para ser una persona preparada con una base firme para afrontar la vida con mayor probabilidad de alcanzar el éxito.

A Jasiel, Daniel y Miguel por el amor, dedicación y apoyo que siempre me han brindado.

Al Dr. Miguel Moctezuma Flores por su confianza, apoyo y consejos brindados para la realización de este proyecto.

Al Dr. Flavio Parmiggiani del CNR-ISAO por el apoyo y material proporcionado.

Al Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas (IIMAS) por la formación recibida dentro y fuera de sus aulas.

Al Consejo Nacional de Ciencia y Tecnología (CONACYT) por el apoyo otorgado para la realización de este Postgrado.

Y a todos aquellos con quienes me he cruzado en el camino y han contribuido con mi preparación y crecimiento, muchas gracias.

Índice general

1. Introducción	7
1.1. Objetivo	7
1.2. Definición del problema	7
1.3. Esquema de análisis	8
1.4. Aportación de la tesis	8
2. Fundamentos	9
2.1. Imagen Digital	9
2.2. Segmentación y Clasificación	9
2.3. Probabilidad y variables aleatorias	9
2.4. Bayes	11
2.5. Teorema del límite central	13
2.6. Procesos estocásticos	13
2.6.1. Estacionaridad	15
2.6.2. Ergodicidad	15
2.7. Aplicaciones de percepción remota	16
3. Clasificación de imágenes	18
3.1. Clasificador de Bayes	19
3.2. Máxima Verosimilitud	20
3.3. Maximización de la Esperanza EM	21
3.3.1. Descripción del algoritmo EM	22
3.3.2. Aplicación del EM a la estimación de k-medias	23
3.3.3. Aplicación del EM a la estimación de Gaussianas	25
3.4. Algoritmo Estocástico de Maximización de la Esperanza SEM	27
3.4.1. Descripción del algoritmo SEM	27
3.5. Segmentación por campos de Markov	29
4. Desarrollo y Resultados	32
4.1. Pruebas k-medias	39
4.2. Pruebas Recocido Simulado	42
4.3. Pruebas EM	45
4.4. Pruebas SEM	48

<i>ÍNDICE GENERAL</i>	3
4.5. Comparación de Resultados	50
5. Conclusiones	52
A. Códigos	54
A.1. EM	54
A.2. K-medias	57
A.3. Recocido Simulado	59
Bibliografía	65

Indice de tablas

4.1. Medias estimadas para la imagen de prueba	34
4.2. Imágenes de prueba	35
4.3. Medias por k-medias para la imagen bosqueT	40
4.4. Medias por k-medias para la imagen bosque2	41
4.5. Medias por k-medias para la imagen iceband	41
4.6. Medias por EM para la imagen bosqueT	45
4.7. Medias por EM para la imagen bosque2	47
4.8. Medias por EM para la imagen iceband	48
4.9. Medias por SEM para la imagen bosqueT	48
4.10. Medias por SEM para la imagen bosque2	49
4.11. Medias por SEM para la imagen iceband	50
4.12. Medias estimadas para la imagen bosqueT	51
4.13. Medias estimadas para la imagen bosque2	51
4.14. Medias estimadas para la imagen iceband	51

Índice de figuras

2.1. Variables aleatorias	10
2.2. Variable aleatoria Gaussiana	11
2.3. Bayes	11
2.4. Procesos aleatorios	14
2.5. V.A. de distribución uniforme	14
3.1. Análisis 4 y 8 conexidad	30
3.2. Funciones trapezoidales para k clases	31
4.1. Imagen imgprueba	32
4.2. Histograma imgprueba	33
4.3. Resultados obtenidos por k-medias (izquierda) y Markov (derecha)	33
4.4. Resultados obtenidos por EM (izquierda) y SEM (derecha)	33
4.5. Histograma de medias imagen de prueba	34
4.6. Imagen bosqueT	35
4.7. Imagen bosque2	36
4.8. Imagen iceband	36
4.9. Histograma de la imagen bosqueT	37
4.10. Histograma de la imagen bosque2	37
4.11. Histograma de la imagen iceband	38
4.12. Imágenes de inicialización para k-medias	39
4.13. Imagen e histograma de bosqueT segmentada por k-medias	40
4.14. Imagen e histograma de bosque2 segmentada por k-medias	40
4.15. Imagen e histograma de iceband segmentada por k-medias	41
4.16. Ventanas de entrenamiento e inicialización bosqueT	42
4.17. Imagen bosqueT segmentada por Markov	43
4.18. Ventanas de entrenamiento e inicialización bosque2	43
4.19. Imagen bosque2 segmentada por Markov	44
4.20. Ventanas de entrenamiento e inicialización iceband	44
4.21. Imagen iceband segmentada por Markov	45
4.22. Imágenes de inicialización para EM	46
4.23. Imagen e histograma de bosqueT segmentada por EM	46
4.24. Imagen e histograma de bosque2 segmentada por EM	47

4.25. Imagen e histograma de iceband segmentada por EM	47
4.26. Imagen e histograma de bosqueT segmentada por SEM	48
4.27. Imagen e histograma de bosque2 segmentada por SEM	49
4.28. Imagen e histograma de iceband segmentada por SEM	49
4.29. Resultados obtenidos por k-medias (izquierda) y Markov (derecha)	50
4.30. Resultados obtenidos por EM (izquierda) y SEM (derecha) . .	51

Capítulo 1

Introducción

La presente propuesta está enfocada a definir un esquema automático de segmentación de imágenes con aplicaciones concretas en el área de la percepción remota. El esquema propuesto incorpora una etapa de estimación estadística a fin de realizar procesos de segmentación más automáticos y robustos en imágenes de radar y pancromáticas de fotografía aérea.

1.1. Objetivo

Realizar investigaciones a fin de resolver adecuadamente el problema de mezcla de modos en la distribución probabilística de clases. A partir de algoritmos de estimación estadística, EM (Expectation Maximization) y SEM (Stochastic Expectation Maximization), realizar estimaciones automáticas de parámetros representativos de la función de densidad condicional de datos. Definir un esquema de segmentación probabilístico y contextual que aplique los parámetros estimados.

1.2. Definición del problema

El objetivo del proceso de segmentación de una imagen es la partición de la misma en regiones significativas que corresponden a una parte o a un conjunto de objetos en la escena. Esto se puede hacer al dividir la imagen en regiones o áreas bajo criterios específicos de homogeneidad [1]. Los procesos de subdivisión deben de cesar cuando todas las regiones de interés han sido identificadas y ya no se debe hacer otra subdivisión más. Esto puede ser fácil de alcanzar en aplicaciones con cierto control, donde el proceso es realizado bajo supervisión de personal experto y en donde los resultados son evaluados por inspecciones visuales. La detección de contornos puede ser una etapa previa para identificar los segmentos que definen las fronteras entre regiones [2].

1.3. Esquema de análisis

Se analizarán esquemas de segmentación paramétricos bayesianos tipo máximo a posteriori (MAP), así como el caso de la segmentación óptima para casos de distribuciones gaussianas. Se estudiará el caso del algoritmo EM y de sus esquemas iterativos de estimación de momentos estadísticos. Se analizará la funcionalidad del algoritmo SEM, el cual es una variante estocástica del algoritmo EM, que pretende ser un mejor estimador en la mezcla de modos de funciones de densidad.

El modelo se inicializa suponiendo valores para la media m_k y la desviación estándar σ_k , siendo k el número de etiquetas o regiones típicas. La imagen se trata como un arreglo de $p(i, j) = 1, 2, \dots, n$ elementos, manejándose como variables aleatorias de tipo gaussiano. Los datos observados son la mezcla de x estados, siendo $p(k)$ la probabilidad del k - ésimo estado. Se supone que la mezcla (o probabilidad condicional) puede ser descrita por una variable gaussiana [3]. El propósito es definir un esquema adaptativo y con funcionalidad no supervisada de segmentación.

1.4. Aportación de la tesis

El empleo de métodos paramétricos probabilísticos tiene la ventaja de contar con fundamentos matemáticos sólidos. Se pueden estimar o asumir la función de probabilidad y un problema es estimar los parámetros de ésta.

En una aproximación máximo a posterior de segmentación, se deben modelar apropiadamente los procesos de obtención de datos y del campo de etiquetas. La última etapa es maximizar las funciones de probabilidad condicional y de probabilidad a priori. La estimación de los parámetros de la función de densidad sirve entonces a un proceso de segmentación o de análisis que puede realizarse automáticamente e incluso que puede conducir a un método óptimo de procesamiento (modelando adecuadamente las funciones de probabilidad conjunta).

La aportación y relevancia del tema de tesis radica en el análisis de métodos de estimación de parámetros funcionales ante la mezcla de modos probabilísticos en las observaciones.

Con fundamentos matemáticos y probabilísticos se propondrá un esquema basado en los algoritmos EM y SEM. La estimación de parámetros servirá en la definición de un algoritmo de segmentación aplicable a imágenes de percepción remota y de fotografía aérea de alta resolución. Dado que los parámetros estimados serían los dos primeros momentos estadísticos, la propuesta se centraría en métodos óptimos de segmentación para el caso de distribuciones gaussianas.

Capítulo 2

Fundamentos

2.1. Imagen Digital

Una imagen puede ser definida como una función de dos dimensiones $f(x, y)$ donde x y y son las coordenadas espaciales (plano) y la amplitud de la función f en algún par de coordenadas (x, y) es llamada intensidad o nivel de gris de la imagen en ese punto. Cuando x y y los valores de la amplitud de la función f son cantidades discretas finitas, a dicha imagen se le llama imagen digital. Una imagen digital está compuesta de un número finito de elementos y cada uno tiene una localidad y valor particulares. A estos elementos se les llama, *elementos de la imagen o pixels*, siendo este último el término más usado para denotar los elementos de una imagen digital.

2.2. Segmentación y Clasificación

Al proceso en el cual una imagen es dividida en segmentos inconexos (un pixel no puede pertenecer a dos segmentos al mismo tiempo) con el propósito de separar las partes de interés de la imagen del resto se le conoce como segmentación. A la tarea de medir los segmentos se le conoce como extracción de características y la asignación de las clases a partir de las mediciones se le llama clasificación.

2.3. Probabilidad y variables aleatorias

Históricamente a la forma más antigua de medir lo incierto se le conoce con el concepto de probabilidad, el cual tuvo su origen en los juegos de azar. Esta se entiende como una medida de nuestra confianza de que ocurra un evento futuro. El concepto de probabilidad es necesario cuando se trabaja con sucesos físicos, biológicos o sociales que generan observaciones que no pueden predecirse con certeza; se dice que si hay n resultados igualmente

probables, todos los cuales ocurren y s son considerados favorables o como un “éxito”, entonces la probabilidad de un “éxito” está dada por $\frac{s}{n}$.

Una variable aleatoria X es una función que asigna valores reales, $X(\alpha)$ a cada salida α en el espacio muestral de un experimento aleatorio. Esta función es simplemente una regla de asignar un valor numérico a cada elemento de un conjunto Fig. 2.1.

La especificación de una medida en el resultado de un experimento aleatorio define una función en el espacio muestral, y por lo tanto una variable aleatoria. El espacio muestral S es el dominio de la variable aleatoria, y el conjunto Sx de todos los valores tomados de X es el rango de la variable aleatoria. De esta forma, Sx es un subconjunto del conjunto de los números reales [7].

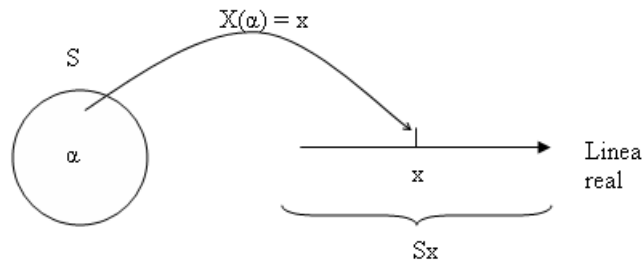


Figura 2.1: Variable aleatoria que asigna un número $X(\alpha)$ a cada salida α en el espacio muestral S de un experimento aleatorio.

Existen diferentes tipos de variables aleatorias, para el desarrollo de esta tesis se utiliza una variable aleatoria de tipo gaussiana, esta decisión se tomó ya que existen en la naturaleza diferentes tipos de fenómenos donde se trata con variables aleatorias X compuestas por la suma de otras variables aleatorias. La descripción de la función de densidad de probabilidad (pdf) de X en términos de las variables aleatorias que la componen puede ser compleja y poco manejable. De cualquier manera, bajo condiciones generales, mientras el número de componentes crece, la pdf de X se aproxima a una variable aleatoria de tipo Gaussiana¹. En especial este tipo de variable aleatoria aparece frecuentemente en problemas que involucran aleatoriedad por lo que se ha llegado a conocer como variable aleatoria normal.

La pdf de una variable aleatoria Gaussiana X está descrita por

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma_x} \exp\left[-\frac{(x-\mu_x)^2}{2\sigma_x^2}\right] \quad (2.1)$$

donde μ y $\sigma > 0$ son números reales y representan la media y la desviación estándar de X . La pdf Gaussiana es una campana simétrica con respecto a

¹Este resultado es basado en el teorema de límite central(Ver Cap. 2.5)

su media μ y su ancho incrementa con respecto a la desviación estándar σ Fig. 2.2.

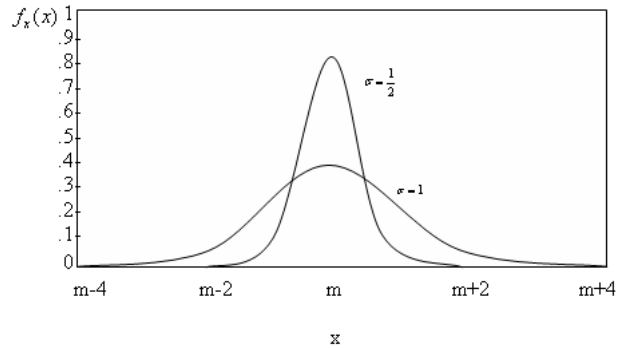


Figura 2.2: pdf de una variable aleatoria de tipo Gaussiana.

Una variable aleatoria X se denomina *discreta* si puede adoptar sólo una cantidad finita o infinita contable de valores distintos.

2.4. Bayes

En el año 1763, dos años después de la muerte del matemático Thomas Bayes(1702-1761), se publicó en “*Philosophical Transactions of the Royal Society of London*” el ensayo “*Essay Towards Solving a Problem in the Doctrine of Chances*” en el que aparece, por primera vez, la determinación de la probabilidad de las causas a partir de los efectos que han podido ser observados. El cálculo de dichas probabilidades recibe el nombre de Teorema de Bayes.

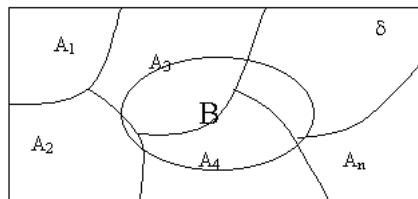


Figura 2.3: Espacio muestral con eventos mutuamente excluyentes.

Sea δ un espacio muestral formado por los eventos A_1, A_2, \dots, A_n , mutuamente excluyentes, como se muestra en la Fig. 2.3 entonces,

$$\delta = A_1 \cup A_2 \cup \dots \cup A_n$$

Si ocurre un evento B definido por:

$$B = B \cap \delta$$

$$B = B \cap (A_1 \cup A_2 \cup \dots \cup A_n)$$

$$B = (B \cap A_1) \cup (B \cap A_2) \cup \dots \cup (B \cap A_n)$$

Donde cada uno de los eventos $B \cap A_i$ son eventos mutuamente excluyentes, tenemos que

$$P(B) = P(B \cap A_1) + P(B \cap A_2) + \dots + P(B \cap A_n) \quad (2.2)$$

Por otro lado sabemos que la probabilidad condicional de que el evento B suceda dado que sucedió el evento A está dada por la expresión 2.3, siempre y cuando $P(A) > 0$ [9]

$$P(B/A) = \frac{P(B \cap A)}{P(A)} \quad (2.3)$$

despejando $P(B \cap A)$ tenemos

$$P(B \cap A) = P(B/A)P(A) \quad (2.4)$$

y sustituyendo la ecuación anterior en 2.2 tenemos

$$P(B) = P(B/A_1)P(A_1) + P(B/A_2)P(A_2) + \dots + P(B/A_n)P(A_n) \quad (2.5)$$

Ahora si se desea calcular la probabilidad de que ocurra el evento A_i dado que B ya ocurrió de la expresión 2.3 tenemos que:

$$P(A_i|B) = \frac{P(A_i \cap B)}{P(B)}$$

y utilizando en la ecuación anterior 2.4 y 2.5

$$P(A_i|B) = \frac{P(B/A_i)P(A_i)}{P(B/A_1)P(A_1) + P(B/A_2)P(A_2) + \dots + P(B/A_n)P(A_n)} \quad (2.6)$$

la ecuación anterior permite enunciar el siguiente Teorema:

Teorema de Bayes

Sea A_1, A_2, \dots, A_n un sistema completo de sucesos, tales que la probabilidad de cada uno de ellos es distinta de cero, y sea B un suceso cualquiera del que se conocen las probabilidades condicionales $P(B|A_i)$. Entonces la probabilidad $P(A_i|B)$ viene dada por la expresión 2.7.

$$P(A_i|B) = \frac{P(B|A_i)P(A_i)}{\sum_{i=1}^k P(B|A_i)P(A_i)} \quad (2.7)$$

2.5. Teorema del límite central

El Teorema del Límite Central o Teorema Central del Límite indica que, bajo condiciones muy generales, la distribución de la suma de variables aleatorias tiende a una distribución gaussiana cuando la cantidad de variables es muy grande.

Existen diferentes versiones del teorema, en función de las condiciones utilizadas para asegurar la convergencia. Una de las más simples establece que es suficiente que las variables que se suman sean independientes, idénticamente distribuidas, con valor esperado y varianza finitas.

Sean X_1, X_2, \dots, X_n variables aleatorias independientes idénticamente distribuidas con $E(X) = \mu$ y $V(X) = \sigma^2 < \infty$. Se define

$$U_n = \sqrt{n} \left(\frac{\bar{X} - \mu}{\sigma} \right) \text{ donde } \bar{X} = \frac{1}{n} \sum_{i=1}^n X_i \quad (2.8)$$

Entonces, la función de distribución de U_n converge a la función de distribución normal estándar conforme $n \rightarrow \infty$. La prueba puede ser encontrada en [7] y [8].

2.6. Procesos estocásticos

Como se mencionó en el apartado 2.3, una variable aleatoria es una regla para asignar a las salidas ζ de un experimento S un valor $X(\zeta)$. Un proceso estocástico $X(\zeta)$ es una regla para asignar a cada ζ una función $X(t, \zeta)$ por esto, un proceso estocástico o proceso aleatorio representa una familia de funciones temporales que dependen de ζ . El dominio de ζ es el conjunto de todas las salidas experimentales, mientras que el dominio de t es el conjunto de números R [8].

Si R es el conjunto de los números reales se dice que $X(\zeta)$ es un proceso continuo, en cambio si R está compuesta por valores enteros decimos que $X(\zeta)$ es un proceso discreto. Un proceso discreto está compuesto por una secuencia de variables aleatorias, esta secuencia se puede escribir como x_n ó $x[n]$.

$x(t)$ puede ser utilizado para representar un proceso estocástico al igual que las variables aleatorias al omitir su dependencia en ζ , de esta manera $x(t)$ puede tener las siguiente interpretaciones:

- Familia o conjunto de funciones $x(t, \zeta)$. En este caso t y ζ representan variables Fig. 2.4.
- Es una única función (o una muestra de un proceso dado). En este caso t es variable y ζ es fija. Es llamada realización, camino muestra, o función muestra del proceso aleatorio.

- Si t es fija y ζ es variable, $x(t)$ es una variable aleatoria con valor igual al proceso en el estado t Fig. 2.5 .
- Si t y ζ son fijas, entonces $x(t)$ es un número.

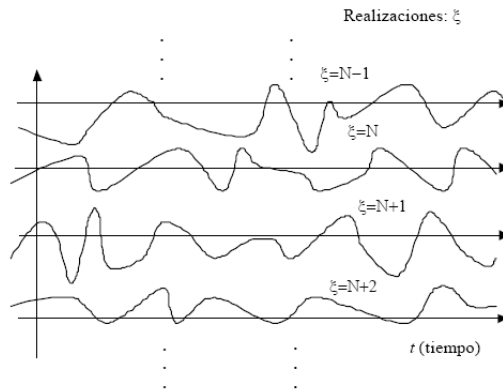


Figura 2.4: Proceso aleatorio representado por sus realizaciones, resultado de distintos experimentos.

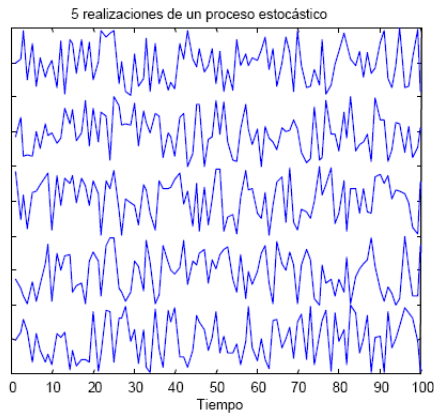


Figura 2.5: Cinco realizaciones de un proceso, el corte a t fijo produce una variable aleatoria de distribución uniforme

2.6.1. Estacionaridad

La mayor parte de los fenómenos que dan lugar a procesos aleatorios mantienen sus propiedades estadísticas constantes con el tiempo, por lo que se dice que son estacionarios. De esta manera, si la media del proceso no depende del tiempo se dice que el proceso es estacionario de orden uno, y si la correlación no depende del instante t , es estacionario de orden 2. En definitiva, si se puede decir que el proceso es estacionario de orden dos, se verificarán las expresiones 2.9 y 2.10

$$r_x(t, \tau) = E\{x * (t)x(t + \tau)\} = r_x(\tau) \forall (t) \quad (2.9)$$

$$\sigma_x^2(t) = \sigma_x^2 = r_x(0) \forall (t) \quad (2.10)$$

Es claro que la estacionaridad del momento de orden dos no implica la estacionaridad de la función de probabilidad conjunta.

Dado que si se verifica lo siguiente

$$f(x_1, x_2; t, t + \tau) = f(x_1, x_2; \tau) \forall (t) \quad (2.11)$$

se está ante un criterio de estacionaridad más estricto, puesto que ésta última implica por supuesto que su centro de gravedad también se mantiene constante con el tiempo. Cuando se verifique la expresión 2.11 se dirá que el proceso es estacionario de orden dos en sentido estricto, mientras que si tan sólo se verifica la expresión 2.9 se dirá que es estacionaria, se suele sobrentender que es de orden dos en sentido amplio.

La estacionaridad en sentido amplio, al margen de ser de más fácil verificación, se cumple para la mayor parte de los procesos con los que trabaja en procesado de señales.

2.6.2. Ergodicidad

Se puede demostrar que, cuando un proceso estocástico cumple ciertas condiciones, es posible estimar consistentemente sus características a partir de una realización del mismo. Los procesos que cumplen tales condiciones se denominan ergódicos [8].

Supongamos que sólo tenemos acceso a una realización del proceso estocástico y pretendemos conocerlo a partir de esa única función temporal; para ello nos interesará conocer sus estadísticas (media, autocorrelación, etc).

Para un proceso estocástico discreto $X[k]$, donde $X[k]$ es un proceso estocástico estacionario en sentido amplio, y definido por valores enteros de k , con $k \geq 0$, la media temporal y la autocorrelación temporal se calcularían mediante las siguientes expresiones respectivamente:

$$M_x = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=0}^{N-1} X[k]$$

Llamaremos autocorrelación temporal a:

$$A_x(\tau) = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=0}^{N-1} X[k]X[k + \tau]$$

Diremos que un proceso es ergódico si sus promedios estadísticos se pueden calcular a partir de una realización: es decir, si sus promedios estadísticos coinciden con sus promedios temporales.

2.7. Aplicaciones de percepción remota

La percepción remota se aplica principalmente en la evaluación de los recursos naturales del planeta y gestión del medio ambiente. Sin embargo, también tiene usos en otras áreas como la medicina, biología e industria, por mencionar sólo algunas.

El grado de complejidad de las aplicaciones depende del recurso que este bajo estudio y del tipo de contribuciones que el sensor remoto pueda realizar. En los casos más simples hay situaciones que involucran a un solo objeto de fácil detección y evaluación, por ejemplo, la evaluación de las regiones de un bosque devastado por un incendio. En los casos más complejos se encuentra el análisis de todo un sistema relacionado con un recurso natural, como lo es la prospección petrolera y minera, ambos análisis requieren de distintos tipos de sensores y plataformas, así como medidas y muestreos en campo y técnicas estadísticas especializadas.

Las condiciones globales del clima requieren de información rápida y confiable de los océanos, bosques, desiertos y capas polares así como regiones circundantes para la predicción del tiempo en diversas zonas del planeta. La percepción remota, especialmente el uso de imágenes de satélite y fotografía aérea de alta resolución, es una manera sumamente práctica de cartografiar las condiciones del planeta sobre una base regional y repetitiva [16].

El análisis de imágenes de percepción remota es útil en los siguientes campos:

- Producción de información geográfica y cartográfica,
- Visión artificial,
- Silvicultura,
- Agricultura,

- Oceanografía y recursos marinos,
- Prevención y evaluación de desastres,
- Inspección industrial [20], [21],
- Identificación de hielo [22].

Las características de las imágenes de la superficie terrestre que las hacen adecuadas para ser útil en estos campos son las siguientes:

- Uniformidad de vista sobre grandes extensiones de terreno;
- Ángulo de vista casi vertical;
- Fidelidad geométrica y radiométrica;
- Buena definición de los rasgos del terreno;
- Capacidad para obtener un producto final en muy poco tiempo;
- Forma digital de los datos.

El último punto mencionado es de gran importancia ya que garantiza la reproducibilidad de los mapas y la capacidad de comparación de éstos en diferentes lapsos de tiempo, así como su manipulación matemática por computadora [16].

Capítulo 3

Clasificación de imágenes

Para poder realizar análisis e interpretación de imágenes generalmente se requiere realizar la clasificación de éstas para obtener información útil en diversas aplicaciones, por ejemplo, para estudiar los daños provocados por un incendio forestal se realiza la clasificación de imágenes multiespectrales y multitemporales.

Cuando se dispone de los datos en forma digital, se puede realizar un análisis cuantitativo basado en los atributos numéricos de los datos. Este análisis cuantitativo consiste primordialmente en la clasificación de los datos. Los datos clasificados se pueden representar como una tabla o como un mapa temático. Esta última opción es la más común ya que facilita su visualización. La clasificación de imágenes está basada en la teoría del reconocimiento de patrones.

Un clasificador asigna un vector de características desconocido, es decir, sin categoría asignada, a una de las M clases de patrones. Las M clases se denotan como $\Omega = w_1, w_2, \dots, w_M$. El clasificador divide el espacio de características en M regiones, llamadas regiones de decisión R^1, R^2, \dots, R^M .

El problema central en el diseño de un clasificador consiste en especificar un conjunto apropiado de funciones discriminantes $g_1(x), \dots, g_M(x)$ [12]. Las funciones discriminantes pueden seleccionarse de diferentes maneras. En ocasiones pueden ser calculadas con precisión si se dispone de un conocimiento a priori completo de los patrones a ser clasificados. Otras veces, se hacen supuestos razonables basándose en un conocimiento cuantitativo sobre los patrones. En cada uno de los casos, especialmente en el segundo, pudiera ser necesario ajustar las funciones discriminantes para obtener resultados aceptables en los patrones a clasificar. Usualmente, este proceso de ajuste se realiza utilizando un conjunto de patrones, los cuales son representativos de los patrones a clasificar. A este proceso se le conoce como entrenamiento, y el conjunto de patrones representativo se conoce como conjunto de entrenamiento.

Un clasificador entrenable de patrones es aquel cuyas funciones discrimi-

nantes pueden ser obtenidas por entrenamiento.

Existen dos tipos de métodos de entrenamiento: paramétricos y no paramétricos.

Los métodos de entrenamiento paramétricos son apropiados para tareas de clasificación donde se sabe de antemano que cada categoría de patrones ω_i , $i = 1, \dots, M$, se puede caracterizar por un conjunto de parámetros. Si los valores de los parámetros fueran conocidos, se podrían especificar directamente las funciones discriminantes adecuadas, pero si no lo son, se usa el conjunto de entrenamiento para estimar los valores de los parámetros y determinar las funciones discriminantes mediante estas estimaciones [12]. En la siguiente sección se verá con más detalle un clasificador paramétrico conocido como clasificador de Bayes.

Los métodos de entrenamiento no paramétricos se aplican más apropiadamente cuando no se pueden hacer supuestos sobre los parámetros que caracterizan las funciones discriminantes. En las aplicaciones comunes de estos métodos se asumen formas funcionales para las funciones discriminantes, por ejemplo lineales o cuadráticas. Estas funciones discriminantes se desempeñan adecuadamente en el conjunto de entrenamiento.

3.1. Clasificador de Bayes

El clasificador de Bayes es un método de entrenamiento paramétrico basado en la teoría de decisiones.

Si cada una de las M clases de patrones se pueden caracterizar por conjuntos de parámetros, estos patrones son variables aleatorias gobernadas por M funciones distintas de probabilidad denotadas como $p(x_i|k)$, $k = 1, \dots, M$. La función $p(x_i|k)$ es la probabilidad de ocurrencia del patrón x dado que pertenece a la categoría k . Se supone que las funciones $p(x_i|k)$ son funciones conocidas de las que se desconocen los valores de los parámetros que las caracterizan.

Además de obtener los parámetros para caracterizar las funciones $p(x_i|k)$, normalmente también se desconocen las probabilidades a priori para cada clase $p(k)$, $k = 1, \dots, M$. De esta manera, el conjunto de patrones en cada categoría se caracteriza por un conjunto de parámetros: las probabilidades $p(k)$ y los parámetros de las funciones $p(x_i|k)$.

El método de entrenamiento paramétrico para el diseño de las funciones discriminantes consiste en los siguientes pasos:

- Las funciones discriminantes se expresan en términos de los parámetros de $p(x_i|k)$ y $p(k)$;
- Se estiman los valores de los parámetros de $p(x_i|k)$ y $p(k)$ a partir de un conjunto de patrones de entrenamiento;

- Se supone que estos estimados son los valores verdaderos de los parámetros y se usan en las expresiones para las funciones discriminantes desarrolladas en el primer paso.

3.2. Máxima Verosimilitud

La técnica denominada método de máxima verosimilitud, elige como estimaciones de un parámetro p a los valores del parámetro que maximiza la verosimilitud (la función de probabilidad conjunta o la función de densidad conjunta) de la muestra observada.

Suponiendo que una función de verosimilitud depende de k parámetros $\theta_1, \theta_2, \dots, \theta_k$. Se eligen como estimaciones los valores de los parámetros que maximizan la verosimilitud $L(y_1, y_2, \dots, y_n | \theta_1, \theta_2, \dots, \theta_k)$.

Para destacar el hecho de que la función de verosimilitud es una función de los parámetros $\theta_1, \theta_2, \dots, \theta_k$, a veces se expresa la función de verosimilitud como $L(\theta_1, \theta_2, \dots, \theta_k)$. A menudo se hace referencia a los estimadores de máxima verosimilitud como MLE (Maximum Likelihood Estimator).

El siguiente es un ejemplo de la forma en cómo se emplea esta técnica para obtener el MLE de μ y σ^2 .

Suponiendo que Y_1, Y_2, \dots, Y_n son una muestra aleatoria de una distribución normal con media μ y varianza σ^2 . Se obtendrán los MLE de μ y σ^2 .

Como Y_1, Y_2, \dots, Y_n son variables aleatorias continuas, $L(\mu, \sigma^2)$ es la densidad conjunta de la muestra expresada como $L(\mu, \sigma^2) = f(y_1, y_2, \dots, y_n | \mu, \sigma^2)$. Y lo que necesitamos es maximizar el valor de μ y σ^2 que maximizan $L(\mu, \sigma^2)$ al igualar la derivada $\partial L(\mu, \sigma^2) / \partial \mu$ y $\partial L(\mu, \sigma^2) / \partial \sigma^2$ a cero y despejar μ y σ^2 respectivamente.

Se puede obtener el $\ln[L(\mu, \sigma^2)]$ ya que esta es una función monótona creciente de $L(\mu, \sigma^2)$ y por consiguiente tanto $\ln[L(\mu, \sigma^2)]$ como $L(\mu, \sigma^2)$ maximizan el mismo valor de μ y σ^2 . Como $L(\mu, \sigma^2)$ es el producto de funciones de μ y σ^2 , y obtener la derivada de estos productos resulta laborioso, es más fácil encontrar el valor de μ y σ^2 que maximicen el $\ln[L(\mu, \sigma^2)]$.

Lo anterior se puede ver en el siguiente desarrollo:

$$L(\mu, \sigma^2) = f(y_1, y_2, \dots, y_n | \mu, \sigma^2)$$

Desarrollando $L(\mu, \sigma^2)$

$$= f(y_1 | \mu, \sigma^2) * f(y_2 | \mu, \sigma^2) * \dots * f(y_n | \mu, \sigma^2)$$

Para una función de densidad normal sustituimos 2.1 en la ecuación anterior y obtenemos:

$$= \left\{ \frac{1}{\sigma\sqrt{2\pi}} \exp \left[\frac{-(y_1 - \mu)^2}{2\sigma^2} \right] \right\} * \dots * \left\{ \frac{1}{\sigma\sqrt{2\pi}} \exp \left[\frac{-(y_n - \mu)^2}{2\sigma^2} \right] \right\}$$

$$= \left(\frac{1}{2\pi\sigma^2} \right)^{n/2} \exp \left[\frac{-1}{2\sigma^2} \sum_{i=1}^n (y_i - \mu)^2 \right]$$

Al obtener el ln de lo anterior queda:

$$L(\mu, \sigma^2) = -\frac{n}{2} \ln \sigma^2 - \frac{n}{2} \ln 2\pi - \frac{1}{2\sigma^2} \sum_{i=1}^n (y_i - \mu)^2.$$

Los estimadores de máxima verosimilitud de μ y σ^2 son los valores que maximizan $\ln[L(\mu, \sigma^2)]$. Al obtener la derivada respecto a μ y σ^2 , nos queda:

$$\frac{\partial \{\ln[L(\mu, \sigma^2)]\}}{\partial \mu} = \frac{1}{\sigma^2} \sum_{i=1}^n (y_i - \mu)$$

$$\frac{\partial \{\ln[L(\mu, \sigma^2)]\}}{\partial \sigma^2} = -\left(\frac{n}{2}\right) \left(\frac{1}{\sigma^2}\right) + \frac{1}{2\sigma^4} \sum_{i=1}^n (y_i - \mu)^2.$$

Igualando estas derivadas a cero y resolviendo simultáneamente, obtenemos de la primera ecuación lo siguiente:

$$\frac{1}{\hat{\sigma}^2} \sum_{i=1}^n (y_i - \mu) = 0 \quad \text{o} \quad \sum_{i=1}^n y_i - n\hat{\mu} = 0, \quad \text{y} \quad \hat{\mu} = \frac{1}{n} \sum_{i=1}^n y_i = \bar{y}.$$

Al sustituir \bar{y} por $\hat{\mu}$ en la segunda ecuación y despejar $\hat{\sigma}^2$, obtenemos

$$-\left[\frac{n}{\hat{\sigma}^2}\right] + \frac{1}{\hat{\sigma}^4} \sum_{i=1}^n (y_i - \bar{y})^2 = 0 \quad \text{o} \quad \hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^n (y_i - \bar{y})^2.$$

Por consiguiente, \bar{Y} y $\hat{\sigma}^2$ son los estimadores de máxima verosimilitud de μ y σ^2 , respectivamente. Podemos observar que \bar{Y} es insesgado para μ y $\hat{\sigma}^2$ no es insesgado para σ^2 , aunque se puede ajustar fácilmente al estimador insesgado S^2 .

3.3. Maximización de la Esperanza EM

El algoritmo EM (Expectation Maximization), inicialmente propuesto por Dempster (Dempster, Laird, y Rubin 1977) [11], presenta una técnica iterativa general para realizar una estimación de máxima verosimilitud de parámetros de problemas en los que existen ciertos datos ocultos. A continuación se describe el algoritmo, y un ejemplo de su aplicación a la resolución de un problema concreto.

3.3.1. Descripción del algoritmo EM

El algoritmo EM puede aplicarse en muchas situaciones en las que se desea estimar un conjunto de parámetros θ que describen una distribución de probabilidad subyacente, dada únicamente una parte observada de los datos completos producidos por la distribución. En general, supongamos que en cada iteración del experimento aleatorio se observa un parámetro z_i y existe un parámetro oculto x_i . Denotamos entonces por $Z = \{z_1, \dots, z_m\}$ al conjunto de datos observados en m realizaciones del experimento, por $X = \{x_1, \dots, x_m\}$ al conjunto de datos no observados y por $Y = Z \cup X$ al conjunto completo de datos. Los datos X pueden considerarse una variable aleatoria cuya distribución de probabilidad depende de los parámetros a estimar θ y de los datos observados Z . De la misma forma, Y es una variable aleatoria ya que está definida en términos de la variable aleatoria X . Llamemos h a la hipótesis actual de los valores de los parámetros θ , y denotemos por h' la hipótesis revisada que se estima en cada iteración del algoritmo EM.

El algoritmo EM busca la hipótesis h' que maximiza $E[\ln p(Y|h')]$, siendo $p(Y|\theta)$ la distribución de probabilidad que define Y y que depende de los parámetros desconocidos θ . Esta distribución de probabilidad define la verosimilitud de los datos completos Y dada una hipótesis h' de los parámetros ocultos. Al maximizar el logaritmo de la distribución se está maximizando la verosimilitud. Se introduce el valor esperado $E[\ln p(Y|h')]$, debido a que el conjunto completo de datos Y es una variable aleatoria. Dado que el conjunto completo de datos Y contiene datos X no observados, se deben considerar todos los posibles valores de X , ponderándolos según su probabilidad. En otras palabras, se calcula el valor esperado $E[\ln p(Y|h')]$ sobre la distribución de probabilidad que gobierna la variable aleatoria Y . Esta distribución está determinada por los valores observados Z más que por la distribución de los valores no observados X .

En general, se desconoce la distribución de Y , determinada por los parámetros θ que se intentan estimar. Por ello, el algoritmo EM usa la hipótesis actual h para estimar la distribución de Y . Se define entonces una función $Q(h|h')$ que proporciona $E[\ln p(Y|h')]$ como una función de h' , bajo la suposición de que $\theta = h$ y dada el conjunto de observaciones Z del conjunto completo de datos Y .

$$Q(h'|h) = E[\ln p(Y|h')|h, Z]. \quad (3.1)$$

En la función $Q(h'|h)$ se supone que la hipótesis h y los datos observados Z tiene unos valores fijos y que éstos definen la distribución de probabilidad de la variables ocultas X (por tanto, sus valores esperados). La distribución de probabilidad de Y definida por Z y h es, entonces, la que se utiliza para calcular $E[\ln p(Y|h')]$ para una hipótesis cualquiera h' . En su forma general, el algoritmo EM repite la siguiente pareja de pasos hasta que converge.

Paso 1: Paso de estimación (E): Calcular $Q(h|h')$ utilizando la hipótesis actual h y los datos observados Z para estimar la distribución de probabilidad de Y

$$Q(h'|h) \leftarrow E[\ln p(Y|h')|h, Z]. \quad (3.2)$$

Paso 2: Paso de maximización (M): Sustituir h por la hipótesis h' que maximiza la función Q

$$h \leftarrow \operatorname{argmax}_{h'} Q(h'|h) \quad (3.3)$$

3.3.2. Aplicación del EM a la estimación de k-medias

Un ejemplo de utilización del algoritmo EM se presenta a continuación donde se utiliza para derivar un método que estime las medias de una mezcla de k distribuciones normales $\theta = (\mu_1, \dots, \mu_k)$ con igual desviación típica σ , que se supone conocida. Los datos observados $Z = \{z_j\}$ son los datos producidos por la distribución. Los datos no observados

$$X = \{x_{1j}, \dots, x_{kj}\}, \quad x_{ij} \in (0, 1), \quad \sum_{i=1}^k x_{ij} = 1$$

indican cuál de las k distribuciones normales se ha utilizado para obtener el dato z_j .

Para aplicar EM primero se necesita derivar una expresión de $Q(h|h')$ para el problema, primero la formulación de $\ln P(Y|h')$. Para un único conjunto de datos $y_j = (z_j, x_{1j}, \dots, x_{kj})$, la verosimilitud de que estos datos hayan sido obtenidos con una hipótesis $h' = (\mu'_1, \dots, \mu'_k)$ se puede escribir como:

$$\begin{aligned} p(y_j|h') &= p(z_j, x_{1j}, \dots, x_{kj}|h') \\ &= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^k x_{ji}(z_j - \mu'_i)^2\right) \end{aligned} \quad (3.4)$$

La expresión 3.4 proporciona la probabilidad de que el valor z_j haya sido generado por la distribución normal seleccionada por los datos ocultos. La probabilidad para todas las instancias m de los datos es

$$\begin{aligned} \ln p(Y|h') &= \ln \prod_{j=1}^m p(y_j|h') \\ \ln p(Y|h') &= \sum_{j=1}^m \ln p(y_j|h') \end{aligned}$$

$$= \sum_{j=1}^m \ln\left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^k x_{ji}(z_j - \mu'_i)^2\right)\right) \quad (3.5)$$

Ahora se debe calcular el valor esperado de esta expresión $\ln p(Y|h')$ sobre toda la distribución de probabilidad que gobierna Y o, de forma equivalente, sobre la distribución de los datos ocultos Y, x_{ij} . Al ser la expresión 3.5 una expresión lineal en función de estos datos, es posible derivar la siguiente expresión

$$\begin{aligned} E[\ln p(Y|h')] &= E\left[\sum_{j=1}^m \ln\left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^k x_{ji}(z_j - \mu'_i)^2\right)\right)\right] \\ &= \sum_{j=1}^m \ln\left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^k E[x_{ji}](z_j - \mu'_i)^2\right)\right) \end{aligned} \quad (3.6)$$

Para resumir, la función $Q(h|h')$ del problema de las k-medias es

$$Q(h'|h) = \sum_{j=1}^m \ln\left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^k E[x_{ji}](z_j - \mu'_i)^2\right)\right), \quad (3.7)$$

donde $h' = (\mu'_1, \dots, \mu'_k)$ y donde los valores esperados de los datos ocultos $E[x_{ij}]$ se calculan a partir de la hipótesis actual y de los datos observados Z . Este valor $E[x_{ij}]$ es simplemente la probabilidad de que la muestra z_j haya sido generada por la distribución normal i .

$$\begin{aligned} E[x_{ij}] &= \frac{p(x = z_j | \mu = \mu_i)}{\sum_{n=1}^k p(x = z_j | \mu = \mu_n)} \\ &= \frac{\exp\left(-\frac{1}{2\sigma^2}(z_j - \mu_i)^2\right)}{\sum_{n=1}^k \exp\left(-\frac{1}{2\sigma^2}(z_j - \mu_n)^2\right)} \end{aligned} \quad (3.8)$$

La ecuación 3.8 completa el primer paso del algoritmo EM, en el que se define la función Q a partir de los datos ocultos esperados. El segundo paso (maximización) consiste en encontrar los valores (μ'_1, \dots, μ'_k) que maximizan la función Q así definida. En este caso,

$$\begin{aligned} \operatorname{argmax}_{h'} Q(h'|h) &= \\ &= \operatorname{argmax}_{h'} \sum_{j=1}^m \ln\left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^k E[x_{ij}](z_j - \mu'_i)^2\right)\right) \\ &= \operatorname{argmin}_{h'} \sum_{i=1}^k \sum_{j=1}^m E[x_{ij}](z_j - \mu'_i)^2 \end{aligned} \quad (3.9)$$

Esto es, la hipótesis de máxima verosimilitud es la que minimiza la suma ponderada de los errores al cuadrado, donde la contribución de cada instancia z_j al error que define μ'_i está ponderada por $E[x_{ij}]$. Esta hipótesis se puede calcular de forma analítica con la expresión 3.10

$$\mu_j \leftarrow \frac{1}{m} \sum_{i=1}^m E[x_{ij}] z_j. \quad (3.10)$$

3.3.3. Aplicación del EM a la estimación de Gaussianas

Si consideramos un conjunto de medidas $x_i, i = 1, \dots, n$. y asumimos que estas pueden ser descritas mediante una distribución de tipo normal, entonces para representarlas con una Gaussiana es necesario calcular su media como:

$$\mu = \frac{1}{n} \sum_{i=1}^n x_i$$

y su varianza como:

$$\sigma^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \mu)^2$$

Supongamos que los datos observados son la mezcla de dos o más estados, $k = 1, 2, \dots, K$, y que $P(k)$ es la probabilidad del estado k entonces como los datos siguen una distribución de tipo normal pueden ser descritos por:

$$p(x_i|k) = \frac{1}{\sqrt{2\pi}\sigma_k} \exp \left\{ -\frac{[x_i - \mu_k]^2}{2\sigma_k^2} \right\} \quad (3.11)$$

Utilizando la regla de Bayes tenemos:

$$p(x_i) = \sum_k p(x_i|k)P(k) \quad (3.12)$$

Entonces si se desea estimar la probabilidad de que un dato pertenezca a un determinado estado se puede utilizar la regla de Bayes de la siguiente forma:

$$P(k|x_i) = \frac{p(x_i|k)P(k)}{\sum_j p(x_i|j)P(j)} \quad (3.13)$$

Para aplicar el algoritmo EM a la estimación de gaussianas primero se inicializa el modelo suponiendo valores para μ_k y σ_k , siendo k el número de clases a discriminar.

Paso E

Obtener la probabilidad de la clase k_i de la siguiente forma:

$$P(k) = \frac{1}{n} \sum_i P(k|x_i) \quad (3.14)$$

donde k es el número de la clase, x_i es el píxel en estudio y n es el número total de elementos en la muestra.

$$p(x_1, \dots, x_n) = \prod_i \sum_k p(x_i|k)P(k) \quad (3.15)$$

Para elegir los parámetros para cada iteración se usa un estimador de máxima verosimilitud al maximizar el log de la función dada por:

$$\log L = \log p(x_1, \dots, x_n) = \sum_i \log \sum_k p(x_i|k)P(k) \quad (3.16)$$

Paso M

Al obtener las derivadas parciales con respecto a μ y σ^2 , e igualando a 0 obtenemos su máximo:

$$\frac{\partial \log L}{\partial m_k} = \sum_i P(k|x_i) \frac{(x_i - m_k)}{\sigma_k^2} = 0 \quad (3.17)$$

$$\frac{\partial \log L}{\partial \sigma_k^2} = \sum_i P(k|x_i) \frac{(x_i - m_k)^2 - \sigma_k^2}{2\sigma_k^3} = 0 \quad (3.18)$$

De estas ecuaciones despejamos a μ_k y σ_k^2 y obtenemos las ecuaciones para realizar la estimación de $p(x_i|k)$

$$m_k = \frac{\sum_i P(k|x_i)x_i}{\sum_i P(k|x_i)} \quad (3.19)$$

$$\sigma_k^2 = \frac{\sum_i P(k|x_i)(x_i - m_k)^2}{\sum_i P(k|x_i)} \quad (3.20)$$

Para determinar la convergencia, se propone un valor de α tal que $0,5 \leq \alpha \leq 1$. Es decir el algoritmo se detiene cuando se cumple

$$|m_{k+1} - m_k| < \alpha$$

3.4. Algoritmo Estocástico de Maximización de la Esperanza SEM

El problema de estimación de clases es un caso particular del siguiente: si consideramos un par de variables aleatorias (E, X) , donde E toma sus valores del espacio finito $E = \{e_1, \dots, e_k\}$ y $X \in R^v$, con m igual al número de clases en la imagen a ser segmentada. Si escribimos a $P(k) = \frac{1}{N} \sum P(k|x_i)$ como la distribución a priori y a $p(x_i|k)$ como la función de densidad gaussiana (media m_k y desviación estándar σ_k) de la distribución de X condicionada a e_k . Podemos escribir:

$$P(k|x_i) = \frac{p(x_i|k)P(k)}{\sum_j p(x_i|j)P(j)}$$

la distribución a posteriori. Se puede observar una muestra x_1, x_2, \dots, x_N de realizaciones de X que tienen por función de distribución de densidad a:

$$p(x_i) = \sum_k p(x_i|k)P(k)$$

de donde debemos estimar los parámetros $(P(k), \mu_k, \sigma_k), 1 \leq k \leq m$.

Existen diferentes métodos iterativos para tratar este tipo de problemas; se pueden encontrar algunas referencias en [13]. Estos métodos presentan a diferencia del SEM las siguientes limitaciones:

- el número de clases se supone conocido,
- la solución depende fuertemente de la inicialización.

La primera limitación es una gran desventaja en procesamiento de imágenes en Percepción remota ya que en imágenes reales es frecuentemente difícil determinar el número de clases a priori.

3.4.1. Descripción del algoritmo SEM

El algoritmo SEM es una implementación mejorada del algoritmo EM que se genera gracias a la adición de una etapa de tipo estocástica. Esta adición resulta, con respecto al algoritmo EM, en las siguientes mejoras:

- es suficiente conocer un límite superior para el número de clases;
- la solución no depende de la inicialización;
- el tiempo de convergencia es menor.

El algoritmo SEM trabaja de la siguiente forma:

Si M es el límite superior del número de clases y $\alpha \in [0, 1]$ es un umbral.

Inicialización

Tomar para cada dato observado x_i la probabilidad $p^0(k|x_i)$ de pertenencia a la clase e_m , $1 \leq m \leq K$. Si no se cuenta con información previa se toma $p^0(k|x_i)$ de forma uniforme. El superíndice denota el número de la iteración.

Para cada $n \geq 0$:

Paso S (Stochastic):

Para cada x_i seleccionar del conjunto de clases $\{e_1, \dots, e_k\}$ un elemento de acuerdo a la distribución $p^n(e_1|x_i), \dots, p^n(e_k|x_i)$, esta selección define una partición del tipo Q_1^n, \dots, Q_k^n de las muestras x_1, \dots, x_N .

Paso M (Maximization):

La idea del algoritmo SEM es suponer que cada x_i que pertenece a Q_m^n , para cada $1 \leq m \leq K$, se puede representar como la distribución definida por $p(x_i|k)$, la densidad correspondiente a las clase e_m . Al denotar a $C_m^n = \text{card}(Q_m^n)$, $Q_m^n = (x_{1,m}^n, \dots, x_{C_m^n,m}^n)$ se puede estimar la media μ_m y la varianza σ_m de la siguiente forma:

$$\mu_m^{n+1} = \frac{1}{C_m^n} \sum_{i=1}^{C_m^n} x_{i,m}^n \quad (3.21)$$

$$\sigma_m^{n+1} = \frac{1}{C_m^n} \sum_{i=1}^{C_m^n} (x_{i,m}^n - \mu_m^n + 1)^2 \quad (3.22)$$

Y las probabilidades a priori,

$$P(k) = \frac{C_m^n}{N} \quad (3.23)$$

Si para m la estimación de 3.23 es menor que la cota α se elimina la clase correspondiente y se regresa a la inicialización.

Paso E (Estimation):

Para cada x_i se define la siguiente distribución $[P^{n+1}(1|x_i), \dots, P^{n+1}(k|x_i)]$ del conjunto de clases por la distribución a posteriori basada en los parámetros actuales $(P^{n+1}(k), \mu_k^{n+1}, \sigma_k^{n+1})$ con $1 \leq m \leq K$:

$$p_m^{n+1}(x_i) = \frac{P^{n+1}(k)p^{n+1}(x_i|k)}{\sum_{q=1}^K P^{n+1}(q)p^{n+1}(x_i|q)} \quad (3.24)$$

donde $p^{n+1}(x_i|k)$ designa a la distribución Gaussiana correspondiente a μ_m^{n+1} y σ_m^{n+1} .

Regresar al paso S.

Diferentes simulaciones realizadas muestran el correcto comportamiento del algoritmo SEM. El algoritmo SEM a sido presentado como una forma alternativa al algoritmo EM utilizado en la estimación de mezcla de modos [10].

3.5. Segmentación por campos de Markov

Uno de los grandes intereses de los métodos markovianos reside en la posibilidad de efectuar estimaciones en el sentido máximo a posteriori (MAP). El uso práctico de un modelo de Campo Aleatorio de Markov se atribuye a la equivalencia entre la distribución de probabilidad de un Campo Aleatorio de Markov y la distribución de probabilidad de Gibbs. La segmentación se considera como un problema de etiquetado en el marco de estudio de los Campos Aleatorios de Markov. A cada uno de los *sites* $s \in S$ en una *lattice* regular, le corresponderá una etiqueta en el dominio L . Cada *site* tiene un conjunto de posibles etiquetas $L = \lambda_1, \lambda_2, \dots, \lambda_k$. Cada una de estas etiquetas corresponde a las clases en las que se desea asignar a los *sites*. Los *sites* están relacionados entre sí por un sistema de vecindades. Las vecindades más comunes en segmentación de imágenes son las de primer y segundo orden. Un Campo Aleatorio discreto de Markov sobre una trama finita se define como una colección de variables aleatorias, que corresponden a los *sites* de una *lattice*, cuya distribución de probabilidad es tal que la probabilidad condicional de que una variable determinada tenga un valor particular, dado los valores del resto de las variables, es idéntica a la probabilidad condicional dados los valores del campo en un conjunto pequeño de *sites*, es decir, su *vecindad* [14].

$$P(f_i|f_{s-i}) = P(f_i|f_{N_i})$$

La probabilidad de que el pixel f tome el nivel de gris f_i , dado todos los niveles de gris f_{s-i} de todos los pixels restantes de la imagen, es igual a la probabilidad de que el pixel i tome el nivel de gris f_i dados los valores de los niveles de gris f_{N_i} de la vecindad de pixels N_i . Para obtener la imagen segmentada es necesario optimizar la función de energía a posteriori, es decir, encontrar la configuración y que maximiza la distribución a posteriori $P(Y|X)$ o bien minimiza una función equivalente de energía:

$$U(Y|X) \approx U(X|Y) + U(Y)$$

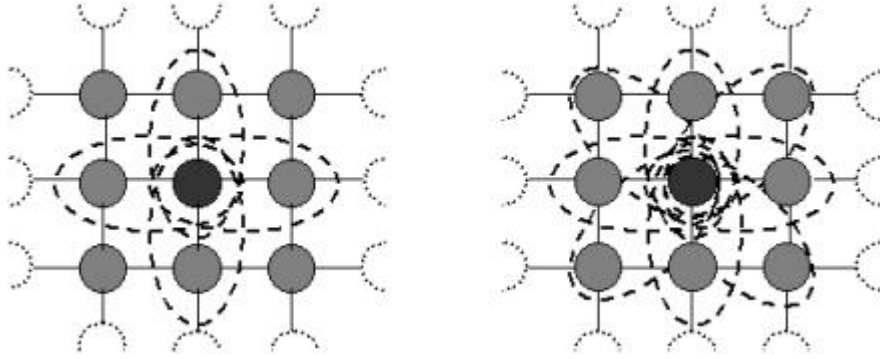


Figura 3.1: Análisis 4 y 8 conexidad

Por ejemplo en [17] se emplea la siguiente ecuación como función de energía, el vector de medias es denotado por μ_k y la matriz de covarianza por S_k ,

$$P(X_{ij} = q_k | Y_{ij}) \propto \frac{P(X_{ij} = q_k)}{|S_k|^{1/2}} \cdot \exp \left\{ -\frac{1}{2} (Y_{ij} - \mu_k)^T (S_k)^{-1} (Y_{ij} - \mu_k) - \sum_{b \in c} V_b(x_{ij})|_{x_{ij}=q_k} \right\}$$

Para esta tesis se utiliza un análisis de tipo contextual, la Fig. 3.1 muestra las agrupaciones de pixels en 4 y 8 conexidad.

Las funciones de energía conjunta $U(X|Y)$ están dadas por las funciones de potencial mostradas en la Fig. 3.2,

La energía a priori $U(Y)$ se calcula dependiendo si el análisis es en 8-conexidad o 4-conexidad. La función de energía a priori $U(Y)$ está definida por:

$$U(Y) = -\beta * n_i(Y)$$

donde $n_i(Y)$ es el número de *sites* en la vecindad N_i que tienen etiqueta Y y β es el coeficiente de homogeneidad.

Los Campos Aleatorios de Markov están ligados a un método específico, el recocido simulado [15], el cual permite resolver el problema de estimación en el sentido MAP, buscando la convergencia hacia el mínimo global de la función de energía a posteriori.

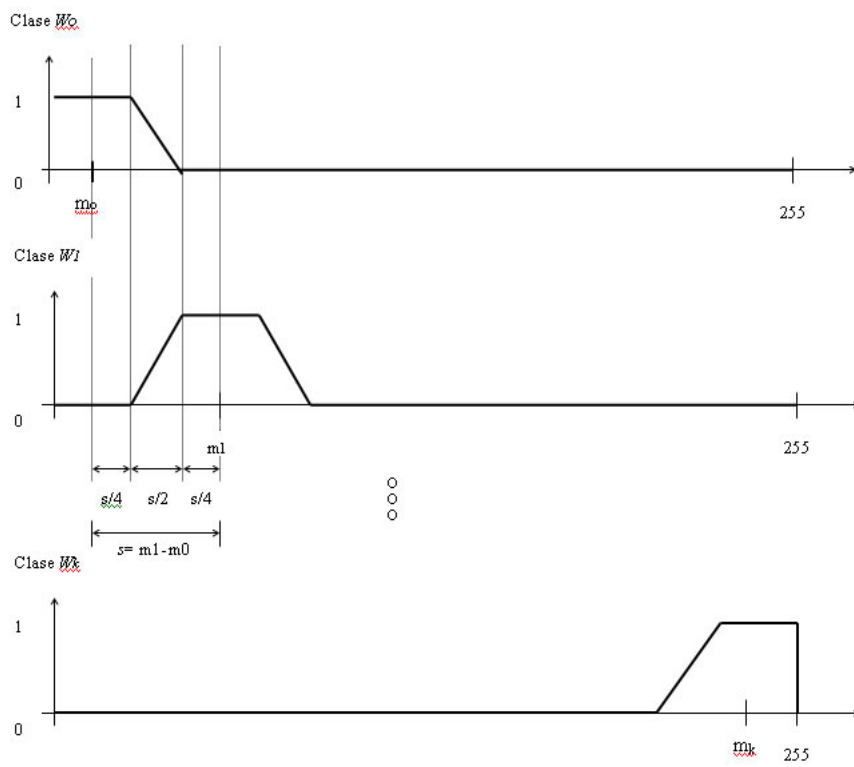


Figura 3.2: Funciones trapezoidales para k clases

Capítulo 4

Desarrollo y Resultados

Para probar el esquema propuesto se creó una imagen sintética generando 5 grupos de datos aleatorios gaussianos con medias $[0, 63, 127, 193, 255]$ y varianza $[0,01]$ de 256×256 pixels, la imagen y su respectivo histograma se muestran en las Figs. 4.1 y 4.2.

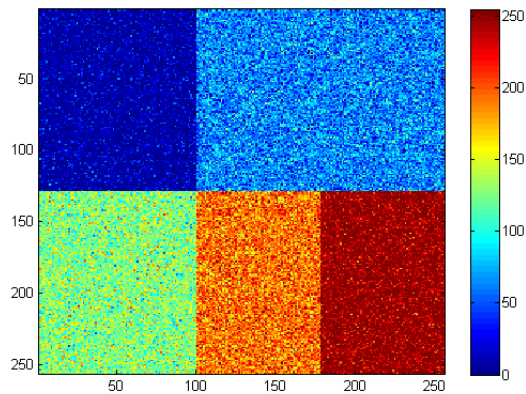


Figura 4.1: Imagen imgprueba

Los resultados de aplicar los métodos a comparar con el SEM se muestran a continuación, y como se puede observar en la Tabla 4.1 el esquema que mejor estima la media μ para la imagen generada es el SEM.

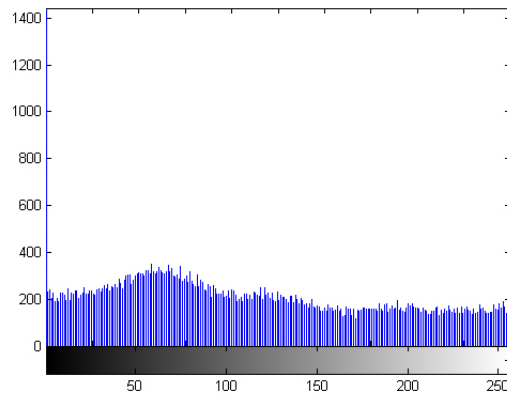


Figura 4.2: Histograma imgprueba

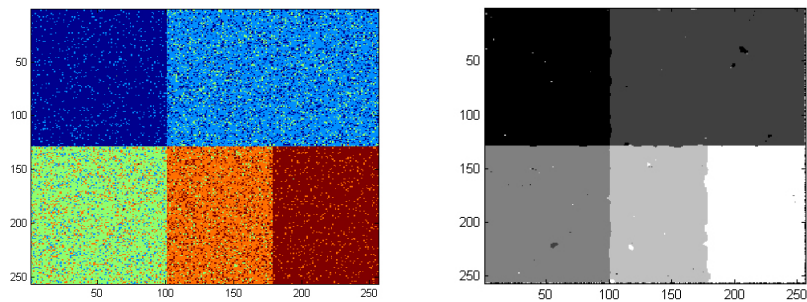


Figura 4.3: Resultados obtenidos por k-medias (izquierda) y Markov (derecha)

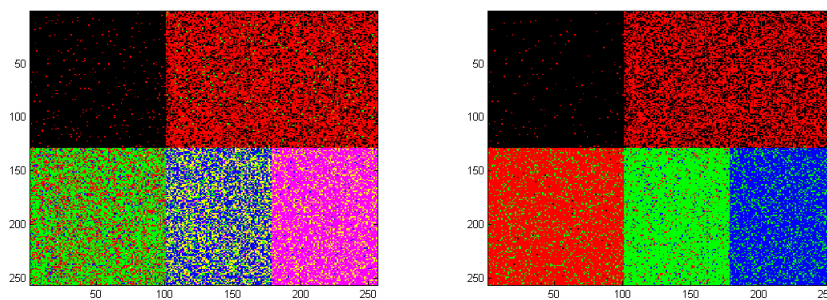


Figura 4.4: Resultados obtenidos por EM (izquierda) y SEM (derecha)

Tabla 4.1: Medias estimadas para la imagen de prueba

<i>clase</i>	1	2	3	4	5
Original	0	63	127	193	255
k-medias	11	66	125	187	245
EM	0	72	92	198	255
SEM	0	64	127	191	255

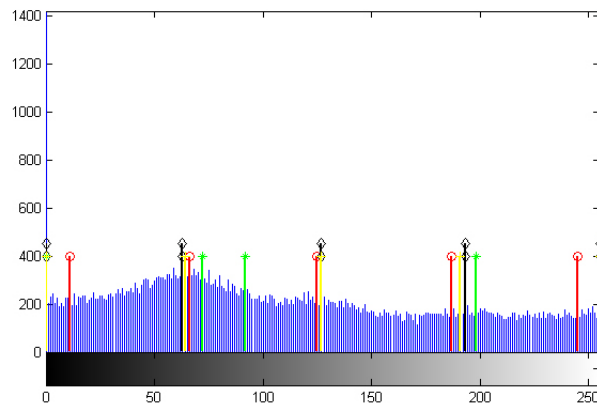


Figura 4.5: Medias sobre el histograma de la imagen de pruebas k-medias (rojo), EM (verde) y SEM (amarillo), Original (negro)

Para llevar a cabo las pruebas del esquema propuesto en imágenes reales, se eligieron tres imágenes, dos pancromáticas bosqueT Fig. 4.6, bosque2 Fig. 4.7 y una SAR iceband Fig. 4.8 con las características mostradas en la Tabla 4.2. Las primeras dos imágenes presentan una sección del bosque de Tlalpan y de la Colonia Miguel Hidalgo, Tlalpan, en el valle de México respectivamente, la imagen iceband presenta una región de Groenlandia en donde se pueden ver capas de hielo y canales de agua. A estas imágenes se les aplicaron los métodos desarrollados en el capítulo 3.

A continuación se presentan algunos de los resultados obtenidos al aplicar la metodología propuesta para realizar la segmentación en regiones urbanas y no urbanas de las imágenes de muestra de una zona al sur de la Ciudad de México Figs. 4.6 y 4.7 y la segmentación del hielo y las zonas de agua en la imagen de Groenlandia Fig. 4.8. Los métodos que se comparan con el EM y SEM son k-medias y Recocido Simulado.

Tabla 4.2: Imágenes de prueba

<i>Imagen</i>	<i>Tam.</i>	<i>Gris min.</i>	<i>Gris max.</i>	<i>Origen</i>
bosqueT	512x512	58	255	México
bosque2	512x512	46	252	México
iceband	435x342	4	255	Groenlandia



Figura 4.6: Imagen bosqueT

La imagen de la Fig. 4.8 fué proporcionada por el Dr. Flavio Parmiggiani del Istituto di Scienze Dell'Atmosfera e del Clima, Consiglio Nazionale delle Ricerche, Bologna Italia. Muestra una escena de Groenlandia en donde se pueden observar algunas zonas de hielo así como canales de agua.



Figura 4.7: Imagen bosque2

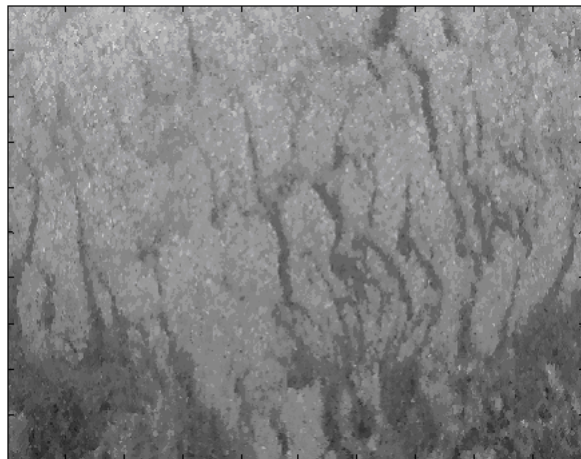


Figura 4.8: Imagen iceband

Los respectivos histogramas de las imágenes de prueba se presentan a continuación Figs. 4.9 y 4.10 en estos podemos observar que para bosqueT y bosque2 estos presentan un contraste muy pobre y que los niveles de gris bajos no aparecen en las imágenes.

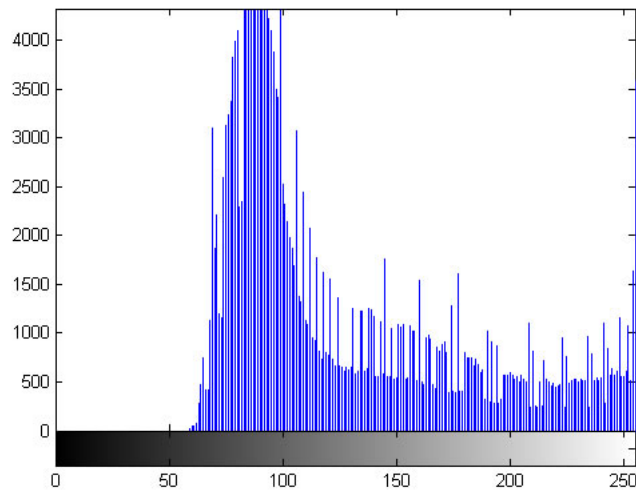


Figura 4.9: Histograma de la imagen bosqueT

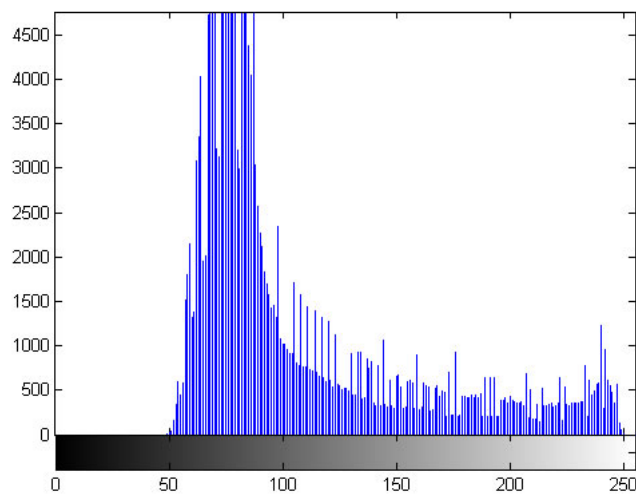


Figura 4.10: Histograma de la imagen bosque2

Para la imagen de prueba iceband podemos observar en el histograma de la Fig. 4.11 que la mayoría de la información esta contenida entre los niveles 50 y 200 por lo que los parámetros como μ deben estar dentro de este rango.

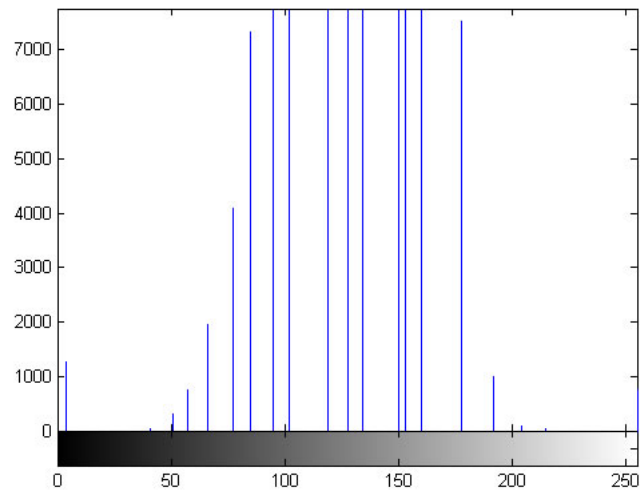


Figura 4.11: Histograma de la imagen iceband

4.1. Pruebas k-medias

Como se vió en el capítulo 3, para llevar a cabo la segmentación por el método clásico k-medias, es necesario inicializar el algoritmo con el número de clases a ser buscadas . Basándonos en los resultados para los métodos propuestos se decidió por suponer $k = 4$ para bosqueT, $k = 3$ para bosque2 y para la imagen iceband. Las imágenes de inicio para este método se presentan en la Fig. 4.12.

Al realizar las pruebas en las imágenes bosqueT y bosque2 se encontró que por el método k-medias la división de clases se realiza de forma correcta para elementos como techos de casas o bodegas, pero para regiones como caminos, calles y zonas verdes no es suficiente la mezcla de clases como se observa en las Figs. 4.13 y 4.14 en la zona inferior donde hay árboles y pastos. Para la imagen bosqueT obtenemos los datos de la Tabla 4.3 al inicializar con 4 clases y al alcanzar 7 iteraciones, para la imagen bosque2 al inicializar con 3 clases y después de 5 iteraciones se obtienen los valores de la Tabla 4.4.

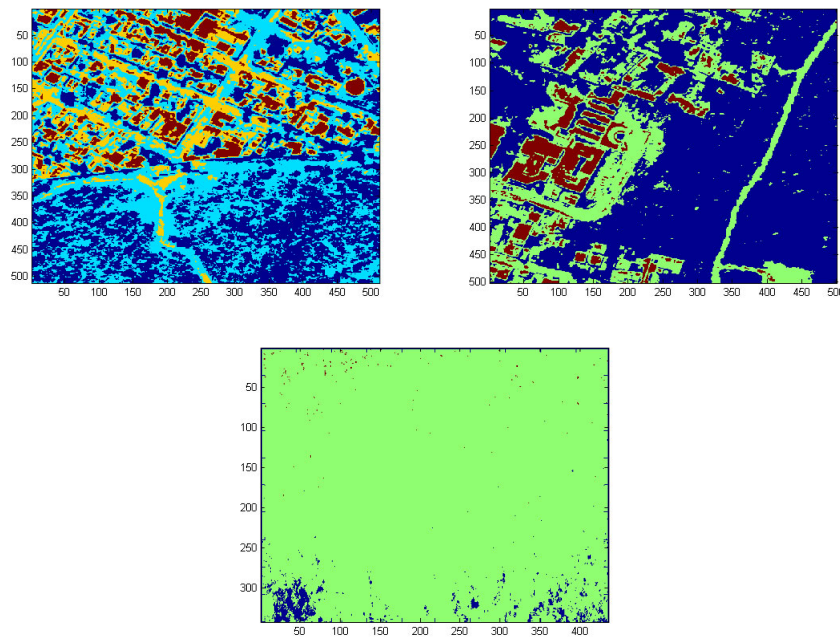


Figura 4.12: Imágenes de inicialización para k-medias

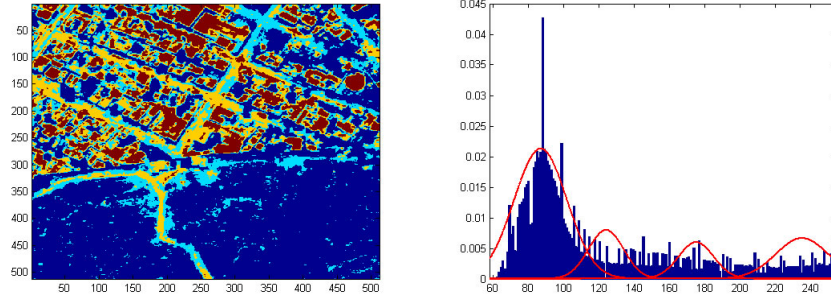


Figura 4.13: Imagen e histograma de bosqueT segmentada por k-medias

Tabla 4.3: Medias por k-medias para la imagen bosqueT

<i>iter.</i>	<i>m1</i>	<i>m2</i>	<i>m3</i>	<i>m4</i>
1	58	123	189	255
2	81	112	186	242
3	84	116	178	239
4	86	120	178	237
5	86	123	175	235
6	87	124	175	235
7	87	124	175	235
	bosque	caminos	caminos	techos

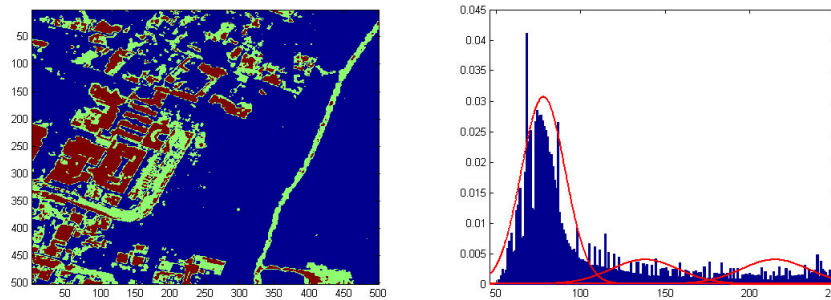


Figura 4.14: Imagen e histograma de bosque2 segmentada por k-medias

Tabla 4.4: Medias por k-medias para la imagen bosque2

<i>iter.</i>	<i>m1</i>	<i>m2</i>	<i>m3</i>
1	46	149	252
2	76	138	227
3	78	139	218
4	78	138	215
5	78	138	215
	bosque	caminos	techos

En el resultado de aplicar el método k-medias sobre la imagen iceband podemos observar que los canales de agua son bien detectados, también se puede apreciar que la clase hielo esta dividida en 2 subclases Fig. 4.15, los valores de las medias para cada clase, después de 4 iteraciones, se presentan en la Tabla 4.5.

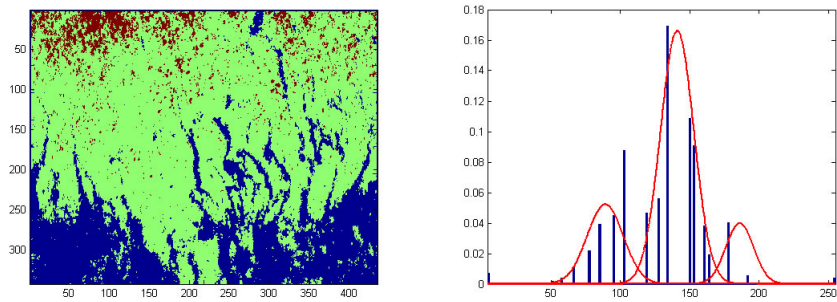


Figura 4.15: Imagen e histograma de iceband segmentada por k-medias

Tabla 4.5: Medias por k-medias para la imagen iceband

<i>iter.</i>	<i>m1</i>	<i>m2</i>	<i>m3</i>
1	4	129	255
2	45	132	246
3	72	136	218
4	89	141	186
	agua	hielo	hielo

4.2. Pruebas Recocido Simulado

Para llevar a cabo las pruebas por recocido simulado es necesario inicializar el algoritmo utilizando regiones tipo para cada clase Figs. 4.16, 4.18 y 4.20, y así obtener una aproximación a la media que caracteriza a cada una de las clases para ser utilizada en las funciones trapezoidales.

Además del paso antes mencionado, es necesario que el algoritmo itere un gran número de veces, en algunos casos más de 100 veces para obtener una clasificación más uniforme.

En las pruebas realizadas por este método, utilizando 8-conexidad, $k=3$ y dejando correr 50 iteraciones se obtuvieron los siguientes resultados,

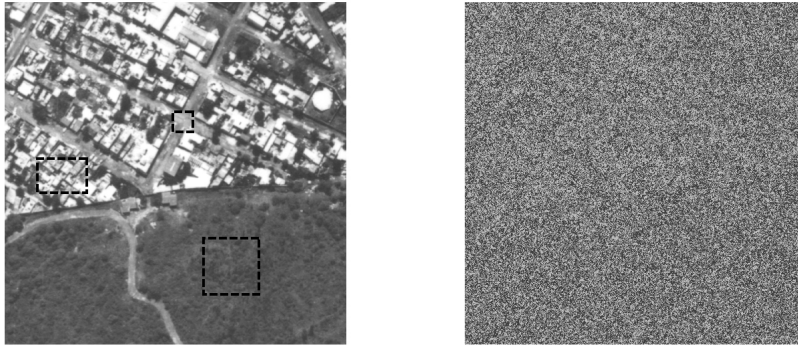


Figura 4.16: Ventanas de entrenamiento e inicialización bosqueT

Como se puede apreciar en las Figs. 4.17, 4.19 y 4.21 los resultados de la segmentación arrojan buenos resultados para elementos como áreas verdes aunque se pierden en gran medida los contornos. Este análisis presentó el inconveniente de no conservar estructuras finas.

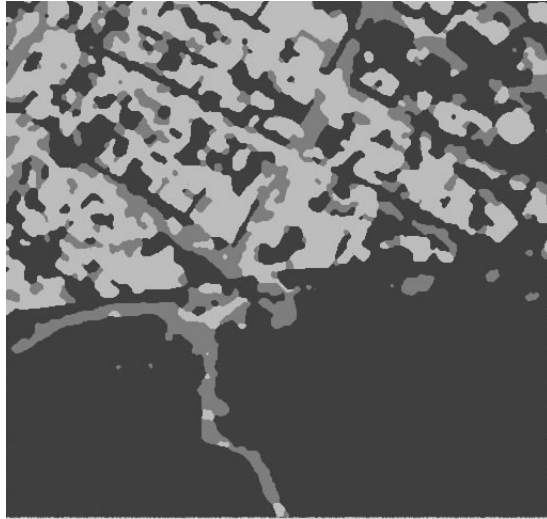


Figura 4.17: Imagen bosqueT segmentada por Markov

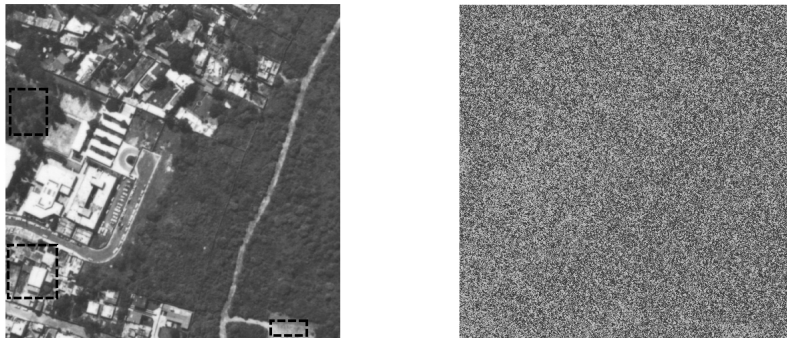


Figura 4.18: Ventanas de entrenamiento e inicialización bosque2

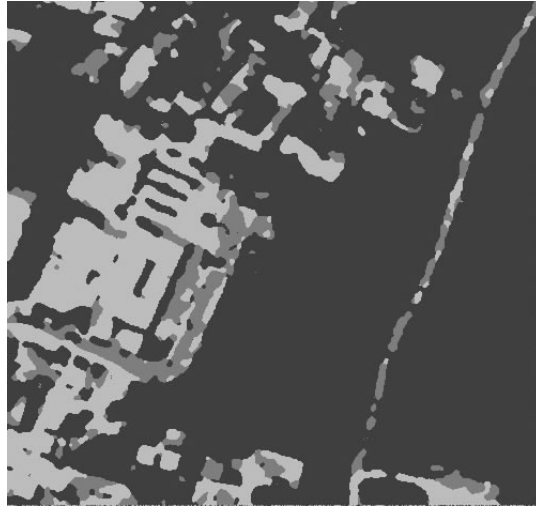


Figura 4.19: Imagen bosque2 segmentada por Markov

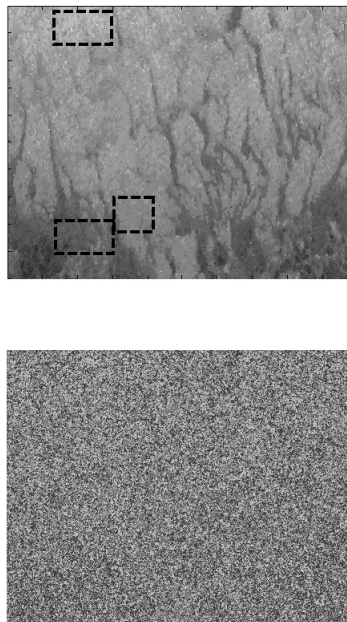


Figura 4.20: Ventanas de entrenamiento e inicialización iceband

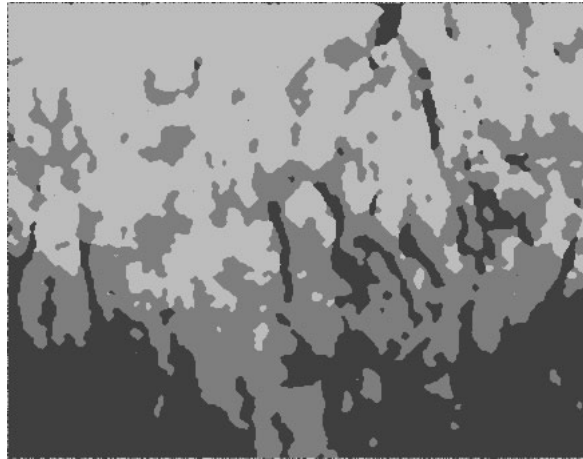


Figura 4.21: Imagen iceband segmentada por Markov

4.3. Pruebas EM

A diferencia del análisis por k-medias y recocido simulado, el análisis por el método Expectation Maximization no requiere de ventanas de entrenamiento ni definición del número de clases, sólo es necesario fijar un número alto de clases por ejemplo $k=10$, mediante el método propuesto se realiza la mezcla de las clases que están más cerca una de otra. Las imágenes de inicio para este método se presentan en la Fig. 4.22. Para las pruebas llevadas a cabo mediante este método, se encontró que para la imagen bosqueT después de 22 iteraciones el algoritmo converge y encuentra 4 clases Tabla 4.6, como se puede observar en la Fig. 4.23.

La definición de áreas verdes (negro) y techos (azul) se realiza de forma correcta además de que se preservan las estructuras a diferencia del análisis por recocido simulado.

Tabla 4.6: Medias por EM para la imagen bosqueT

	$m1$	$m2$	$m3$	$m4$	$m5$	$m6$	$m7$	$m8$	$m9$	$m10$
ini	58	80	101	124	145	167	189	211	233	255
fin			91		145		183			255
			Ver		cam		cam			tech

Para la imagen bosque2 fueron necesarias 35 iteraciones para que el algoritmo se detenga y obtener las medias finales para las 3 clases encontradas Tabla 4.7, áreas verdes (negro), caminos y calles (rojo) y techos (verde) Fig. 4.24.

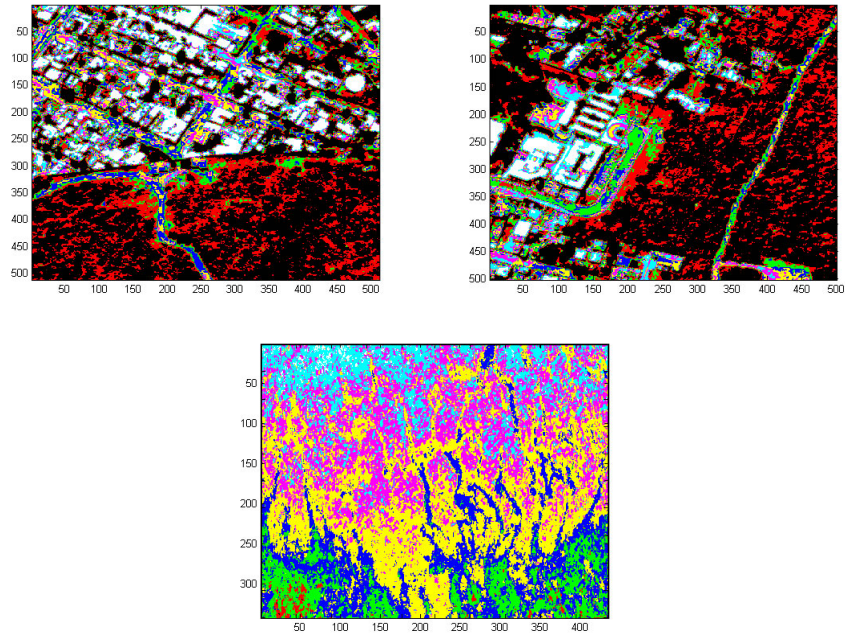


Figura 4.22: Imágenes de inicialización para EM

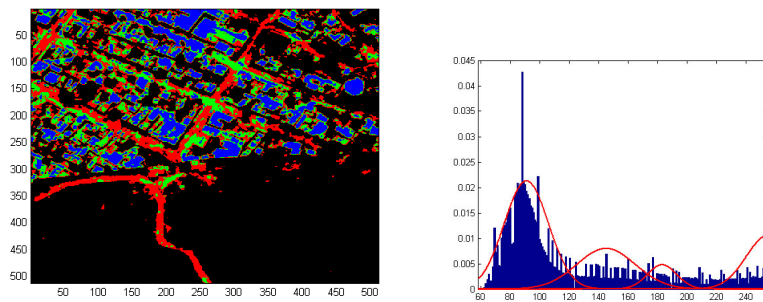


Figura 4.23: Imagen e histograma de bosqueT segmentada por EM

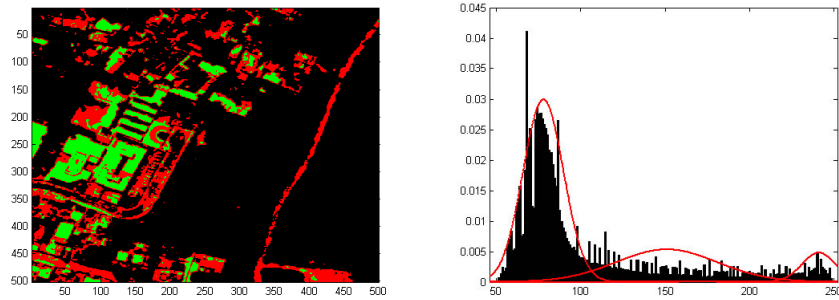


Figura 4.24: Imagen e histograma de bosque2 segmentada por EM

Tabla 4.7: Medias por EM para la imagen bosque2

	$m1$	$m2$	$m3$	$m4$	$m5$	$m6$	$m7$	$m8$	$m9$	$m10$
ini	46	69	91	114	137	160	183	206	229	252
fin		78				151			241	
		Ver			cam					tech

Para la segmentación de iceband, fueron necesarias 10 iteraciones para cumplir la condición $e < 0,001$ con lo que se llegan a discriminar 4 clases Fig. 4.25 y Tabla 4.8, para este caso la zona de agua y canales fue dividida en 2 clases (negro y rojo), la clase hielo fué dividida en 2 subclases (verde, azul).

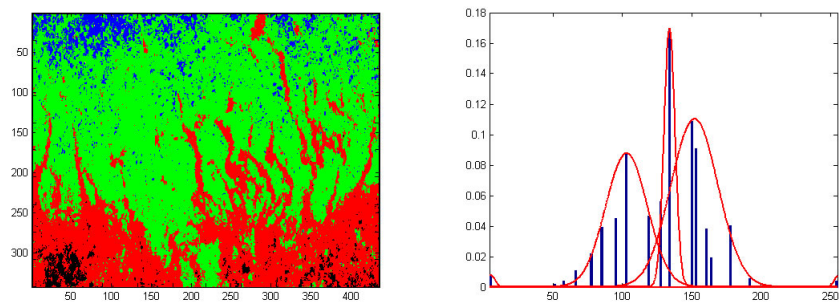


Figura 4.25: Imagen e histograma de iceband segmentada por EM

Tabla 4.8: Medias por EM para la imagen iceband

<i>iter.</i>	<i>m1</i>	<i>m2</i>	<i>m3</i>	<i>m4</i>	<i>m5</i>	<i>m6</i>	<i>m7</i>	<i>m8</i>	<i>m9</i>	<i>m10</i>
1	4	42	69	93	120	145	165	187	226	255
2	4		76	96	125		157	180	212	
3	4		81	98	130	154	164	179		
4	4		85	99	134	158				
5	4			100	136	152				
6	4			101	136	150				
7	4			102	135	150				
8	4			103	134	151				
9	4			103	134	151				
10	4			103	134	152				

4.4. Pruebas SEM

Al correr el algoritmo SEM para la imagen bosqueT obtenemos los siguientes datos al inicializar con 10 clases y después de 5 iteraciones quedan 3 clases como se muestra en la Fig. 4.26 y Tabla 4.9.

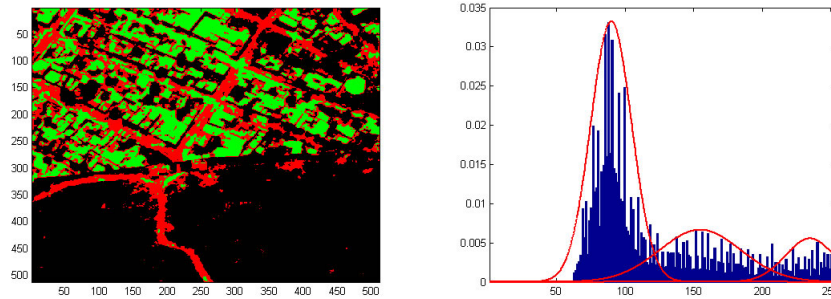


Figura 4.26: Imagen e histograma de bosqueT segmentada por SEM

Tabla 4.9: Medias por SEM para la imagen bosqueT

	<i>m1</i>	<i>m2</i>	<i>m3</i>	<i>m4</i>	<i>m5</i>	<i>m6</i>	<i>m7</i>	<i>m8</i>	<i>m9</i>	<i>m10</i>
ini	59	80	101	123	145	167	189	211	233	255
fin		90			155				235	

Al correr el algoritmo SEM para la imagen bosque2 obtenemos los si-

güentes datos al inicializar con 3 clases y después de 5 iteraciones se obtiene la Fig. 4.27 y los datos de la Tabla 4.10.

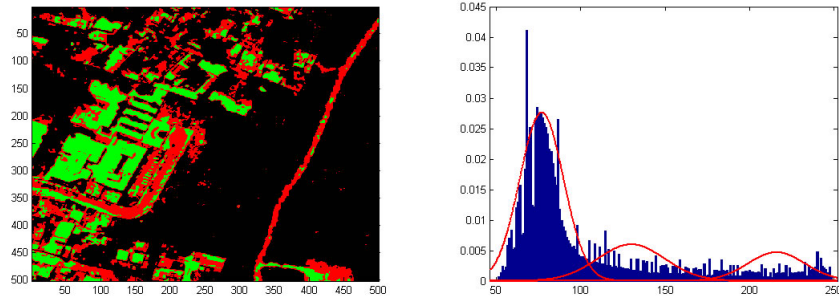


Figura 4.27: Imagen e histograma de bosque2 segmentada por SEM

Tabla 4.10: Medias por SEM para la imagen bosque2

	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
ini	46	69	91	114	137	160	183	206	229	252
fin		77			130			216		

Al correr el algoritmo SEM para la imagen iceband obtenemos los datos de la Tabla 4.11 al inicializar con 10 clases y después de 27 iteraciones queda la imagen de la Fig. 4.28.

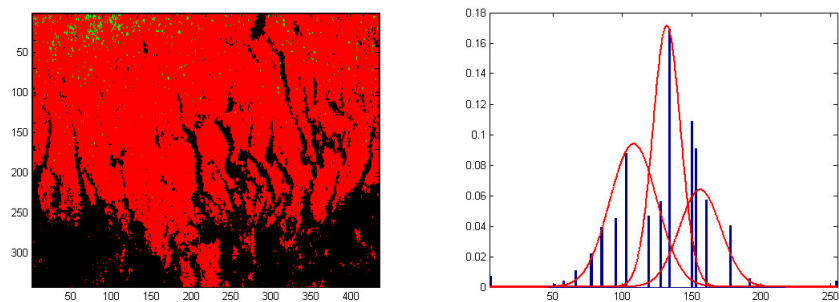


Figura 4.28: Imagen e histograma de iceband segmentada por SEM

Tabla 4.11: Medias por SEM para la imagen iceband

	$m1$	$m2$	$m3$	$m4$	$m5$	$m6$	$m7$	$m8$	$m9$	$m10$
ini	4	32	60	87	115	143	171	199	227	255
fin					109	133	157			

4.5. Comparación de Resultados

Como se puede apreciar en la Fig. 4.29, los elementos como techos son bien segmentados por medio del método k-medias, aunque los caminos, calles y áreas verdes no son segmentados de forma correcta. Por otro lado mediante el método de Markov podemos observar una mejor distinción de áreas verdes y caminos, aunque en este resultado las estructuras finas se pierden.

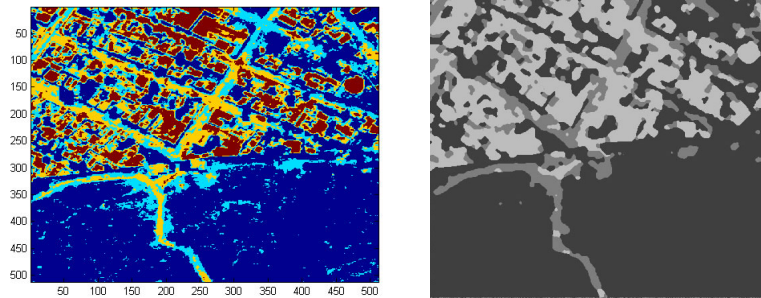


Figura 4.29: Resultados obtenidos por k-medias (izquierda) y Markov (derecha)

A diferencia de los resultados mostrados en la Fig. 4.29, en la Fig. 4.30 podemos observar una buena definición de elementos urbanos así como la obtención de tres clases solamente, caminos(rojo), casas, bodegas y edificios (verde) y áreas verdes (negro).

En las Tablas 4.12, 4.13 y 4.14 se muestran las medias estimadas por los diferentes métodos estudiados. Se puede observar que para la imagen de prueba bosqueT el número de clases encontrado por SEM a diferencia de EM, segmenta la imagen en las regiones en las que se supone que esta puede separarse.

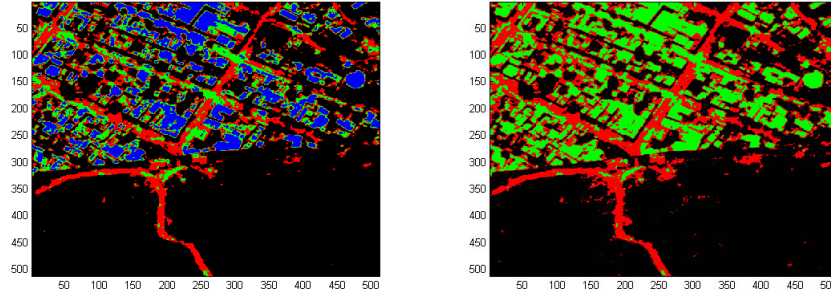


Figura 4.30: Resultados obtenidos por EM (izquierda) y SEM (derecha)

Tabla 4.12: Medias estimadas para la imagen bosqueT

<i>clase</i>	1	2	3	4
k-medias	87	124	175	235
EM	91	145	183	255
SEM	90	155		235

Tabla 4.13: Medias estimadas para la imagen bosque2

<i>clase</i>	1	2	3
k-medias	78	138	215
EM	78	151	241
SEM	77	130	216

Tabla 4.14: Medias estimadas para la imagen iceband

<i>clase</i>	1	2	3	4
k-medias	89	141	186	
EM	4	103	134	152
SEM	109	133	157	

Capítulo 5

Conclusiones

Siendo México uno de los países más diversos del mundo, alberga entre el 60 % y 70 % de la biodiversidad total del planeta y partiendo de la necesidad de llevar a cabo un desarrollo sustentable de nuestros recursos, es importante conocer la distribución espacial de la vegetación, las modificaciones de la tierra, etc. También, se debe contar con información sobre el crecimiento demográfico, el uso y distribución de las tierras, etc.

En la actualidad se cuenta con una herramienta muy poderosa llamada Percepción Remota, que proporciona información para poder tomar mejores decisiones con respecto al crecimiento demográfico, distribución de áreas verdes, carreteras, etc.

La materia prima de la Percepción Remota es el conjunto de imágenes de las áreas de interés, éstas pueden obtenerse mediante diferentes tipos de instrumentos con características muy diferentes unas de otras. Las imágenes utilizadas durante el desarrollo de esta tesis son de radar y pancromáticas de fotografía aérea.

La percepción remota es un dominio de gran actividad de investigación, esta área nos permite un gran campo de desarrollo de procesos y algoritmos para realizar la segmentación de imágenes.

Un apartado dentro de la segmentación de imágenes, es el desarrollo de algoritmos, técnicas y procesos para realizar una segmentación de tipo supervisada y no supervisada. Este tipo de segmentación es una herramienta muy poderosa en la clasificación de imágenes, no sólo de fotografía aérea, si no también en la de imágenes biomédicas, en la visión de robots, etc.

En algoritmos aplicados en la segmentación supervisada se debe de especificar el número de clases (k-medias), o se deben proporcionar las ventanas de entrenamiento para obtener las medias (segmentación Bayesiana, Markov) para después llevar a cabo la segmentación. En este tipo de algoritmos, los resultados dependen fuertemente de los datos de entrenamiento y del número de clases en las que se decida hacer la segmentación, es decir el resultado depende del usuario, de la forma en que introduzca la información

y de cómo analice los resultados.

El esquema propuesto para alcanzar el objetivo de este tesis incorpora una etapa de estimación estadística a fin de realizar procesos de segmentación más automáticos y robustos. Se realizaron experimentos con los métodos de maximización de la esperanza (EM y SEM), k-medias y un metodo de analisis contextual via la formulacion de campos aleatorios de Markov para definir un esquema automático de segmentación de imágenes con aplicaciones concretas en el área de la percepción remota.

El problema de estimación de parámetros se aborda desde los principios de información y probabilidad, la diferencia entre estimación MAP y ML se resume en términos de conocimiento a priori de los parámetros a estimar.

Se encontró que mediante la utilización de métodos como SEM y EM se puede realizar la definición de los parametros estadísticos necesarios para formular las funciones de densidad de probabilidad para discriminar de forma adecuada las regiones urbanas y no urbanas. Además de proporcionar una herramienta para encontrar de forma automática el número de regiones para segmentar una imagen.

Lo anterior a diferencia de los métodos supervisados en donde se debe estudiar la imagen o tener conocimiento previo de las regiones de interés, probó ser de gran utilidad en imágenes pancromáticas de fotografía aérea así como en imágenes de radar como se puede apreciar en los resultados del apartado 4.4 y 4.5.

El objetivo de este trabajo es proponer una metodología para segmentar imágenes pancromáticas y de radar de regiones urbanas y no urbanas, y con los resultados obtenidos podemos concluir que el objetivo esta cubierto.

Apéndice A

Códigos

En este apéndice, se presentan los códigos que se han utilizado en la elaboración de este trabajo. El objetivo es mostrar la parte fundamental sobre la que se sostienen todas las consideraciones de los resultados obtenidos y analizados. Para la programación de los métodos se utilizó Matlab Ver. 7.1 y fueron probados en una computadora Pentium IV con 1 GB de memoria Ram y 80 GB de disco duro.

Se presentan códigos para clasificar basados en:

- EM
- k-medias
- Recocido simulado, criterio de Metrópolis y Modelo de Potts para 8 conexidad

A.1. EM

```
%%%%%%%%%%%% Find mean and variance by EM %%%%%%%%%%%%%%
% Purpose:
%   Find the mean, variance, probability of gaussian distributions by EM
%   algorithm
%
%
% Input Variables:
%   c      classes number
%   iter   maximum iteration number
%   ni     image name
%   iv     initial variance
%   ipk    initial probability for P(k), uniform probability
%   img3d  A matrix image of zeroes
%   pkxi3d A matrix of zeroes to store the probability of P(k|xi)
```



```

%      pxik3d  A matrix of zeroes to store the probability of P(xi|k)
%      m3d,nmd3, v3d, nv3d, pk3 and npk3 matrices of zeroes
%
% Returned Results:
%      mm      A matrix with all the means
%      vm      A matrix with all the variances
%      pkm     A matrix with all the probabilities for P(k)
%
% Processing Flow:
%      1. Load input image
%      2. Initialize auxiliary matrices
%      3. E step
%      4. M step
%
% Restrictions/Notes:
%      Non
%
% Functions called:
%      1. linspace:  for initialize the mean vector
%      2. zeros:     of initialize the 3d matrices
%      3. exp:       exponential function
%      4. sum:       for add the columns and rows of 3d matrices
%      5. sqrt:     for square root operation
%      6. min,max:  to obtain the min and max gray level in the image
%
% Author:      David Morales
% Date:       June 2007
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all;
% Read the image into array "img", and get its 2D dimensions,
% Note that the image resides in the directory of this .m file

img=double(imread('pruebas','jpg'));
mapa=[0 0 0 ; 1 0 0 ; 0 1 0 ; 0 0 1 ; 1 1 0 ; 1 0 1 ; 0 1 1 ; 1 1 1];
imagesc(img)
[x,y]=size(img);
n=x*y;
% here we set the number of classes
c=5;
% Next step fills the 3d image cube
img3d=zeros(x,y,c);
for j=1:c
    img3d(:,:,j)=img;

```

```

end
% linspace performs the initialization of the mean vector
m=linspace(min(min(img)),max(max(img)),c)
% making the variance vector and the P(k) initialization vector
v(1:c)=2;
pk(1:c)=1/c;

%Initialize the iteration steps
for i=1:50
    i
    % Stores in one matrix all the means calculated by the algorithm
    mm(i,:)=m;
% Initialize all 3d matrices used
    pkxi3d=zeros(x,y,c);
    pxik3d=zeros(x,y,c);
    m3d=zeros(x,y,c);
    nm3d=zeros(x,y,c);
    v3d=zeros(x,y,c);
    nv3d=zeros(x,y,c);
    pk3d=zeros(x,y,c);
    npk3d=zeros(x,y,c);
% Map the mean, variance and P(k) vector into 3d matrix.
    for j=1:c
        m3d(:,:,j)=m(j);
    end
    for j=1:c
        v3d(:,:,j)=v(j);
    end
    for j=1:c
        pk3d(:,:,j)=pk(j);
    end

% E - step
% Finds the P(xi|k) - gaussian distribution - for all classes
    pxik3d=(1/(sqrt(2*pi).*v3d)).*exp((-img3d-m3d).^2./(v3d.^2 *2));
% Find the P(k|xi)
    aux=(pxik3d.*pk3d);
    for j=1:c
        pkxi3d(:,:,j)=aux(:,:,j)./sum(aux,3);
    end

% M - step
    pk=sum(sum(pkxi3d))/n;
    nm=sum(sum(pkxi3d.*img3d))./sum(sum(pkxi3d));

```

```

    for j=1:c
        nm3d(:,:,j)=nm(j);
    end
    v=sqrt(sum(sum(pkxi3d.*((img3d-nm3d).^2))./sum(sum(pkxi3d))));
    aux2(1:c)=m;
    aux3(1:c)=nm;
    dist=abs(img3d-m3d);
    [minimo,res]=min(dist,[],3);
    colormap(mapa)
    figure,
    image(res)
    if sum(abs(aux3-aux2))/c<0.2 %decides if the algorithm ends
        'fin'
        mm(i+1,:)=nm;
        break
    else
        m=nm;
    end
end
end

```

% ***** END of *.m file *****

A.2. K-medias

```

%Calcula k-medias para un vector de tamaño
%n y con k, depende mucho de la inicialización de los centroides.
%al final entrega un *.avi con los cambios en la imagen y el histórico
%de medias.
%R - nombre de la imagen
%k número de clases
function Y=kmeans(R,k)
img0=imread(R,'jpg');
s1=size(img0,1);
s2=size(img0,2);
img(1,:)=double(img0(:));
[w,n]=size(img(1,:));
c_n(1:k,1:n)=0;
img(2,:)=0;
imin=min(min(img(1,:)));
imax=max(max(img(1,:)));
%Dependiendo de la inicialización, el algoritmo
%discrimina de una forma u otra.

```

```

c=((imax-imin)/k)*(1:k)'*ones(1,n)+imin;
c=round(c-((imax-imin)/(2*k)));

%-----INICIO AVI FILE
fig=figure;
set(fig,'DoubleBuffer','on');
set(gca,'xlim',[-3 13],'ylim',[-3 13],...
      'NextPlot','replace','Visible','off')
mov = avifile('k-means.avi','fps',1,'compression','none')
%-----FIN AVI FILE
bool=true;
aux=1;
while(bool)
    c_hist(1:k,aux)=c(1:k,1);
    for aux1=1:k
        dif(aux1,1:n)=abs(img(1,:)-c(aux1,1:n));
    end
    [a,b]=min(dif(:,:));
    img(2,:)=b;
    i_hist(aux,:)=b;
    imgF=reshape(i_hist(aux,:), s1, s2);
    imagesc(imgF);
    F = getframe(gca);
    mov = addframe(mov,F);
    for R = 1:k
        count=0;
        sum=0;
        for C = 1:n
            if img(2,C)==R
                count=count+1;
                sum=sum+img(1,C);
            end
        end
        c(R,1:n)=round(sum/count);
    end
    aux=aux+1
    c_hist(1:k,aux)=c(1:k,1);
    if c_hist(:,aux)==c_hist(:,aux-1)
        bool=false;
    end
end
Y=c_hist;
mov = close(mov);%*Cierra AVI FILE
%-----TERMINA kmeans-----

```

A.3. Recocido Simulado

```

function Y=markov8c(R,nclases,iter,clases)
imagen=imread(R,'jpg');
imagen=imagen(:,:,1);
itertext=0;

[tamanox,tamanoy]=size(imagen);
figure,imshow(imagen)

if nargin==3
    for i=1:nclases
        trainingzone=imcrop;
        clases(i)=mean(mean(trainingzone));
    end
end

Y=unidrnd(nclases,tamanox,tamanoy);
Y=Y+1;

clases=order(clases);

tinicial=clock;
tinicial(1,4)=60.*60.*tinicial(1,4);
tinicial(1,5)=60.*tinicial(1,5);
tinicial=tinicial(1,4)+tinicial(1,5)+tinicial(1,6);
clases=floor(clases);
intervalos=interval(clases);
subintervalos=floor((1/4)*intervalos);
energia_a_priori_clases=ones(nclases,255);
%-----CLASE UNO -----
for i=1:1:clases(1)+subintervalos(1)
    energia_a_priori_clases(1,i)=0;
end

for i=clases(1)+subintervalos(1):1:clases(1)+3*subintervalos(1)
    energia_a_priori_clases(1,i)=(i-(clases(1)+subintervalos(1)...
    ))/(2*subintervalos(1));
end

%-----CLASE 2 a N-1-----
for u=2:nclases-1
    for i=clases(u)-subintervalos(u-1):1:clases(u)+subintervalos(u)

```

```

        energia_a_priori_clases(u,i)=0;
    end
    for i=clases(u)-3*subintervalos(u-1):1:clases(u)-subintervalos(u-1)
        energia_a_priori_clases(u,i)=(i-(clases(u)-...
        subintervalos(u-1)))/(-2*subintervalos(u-1));
    end
    for i=clases(u)+subintervalos(u):1:clases(u)+3*subintervalos(u)
        energia_a_priori_clases(u,i)=(i-(clases(u)+...
        subintervalos(u)))/(2*subintervalos(u));
    end
end
end

%-----CLASE N-----
for i=clases(nclases)-subintervalos(nclases-1):1:255
    energia_a_priori_clases(nclases,i)=0;
end
for i=clases(nclases)-3*subintervalos(nclases-1):1:clases(nclases)...
-subintervalos(nclases-1)
    energia_a_priori_clases(nclases,i)=(i-(clases(nclases)-...
    subintervalos(nclases-1)))/(-2*subintervalos(nclases-1));
end
%-----FUNCIONES DE ENERGIA-----
Uxw=energia_a_priori_clases;

%T=.5;
%beta=0.1;
T=2.5;
beta=0.35;

iteraciones=iter;

%inicio de numero de iteraciones
for iter=1:1:iteraciones
    sizx=tamanox-1; sizy=tamanoy-1;
%inicio de barrido de la imagen
    for x1=2:1:sizx
        for y1=2:1:sizy
            %escoger un site de con ubicacion x1,y1
            % y su vecindad asociada
            ventana=Y(x1-1:x1+1,y1-1:y1+1);
            %sorteo de variable aleatoria [1,2]
            eleccion=unidrnd(nclases);
            eleccion=eleccion+1;
            actual =Y(x1,y1);
        end
    end
end
end

```

```

    prob_actual=0;
    prob_eleccion=0;
    n1=0;
    n2=0;

```

```

% analisis de (((((((((((((((((8-conexidad))))))))))))))

```

```

%----- analisis para ACTUAL-----

```

```

    if actual==ventana(1,1)
        prob_actual=prob_actual-beta.*n1;
        n1=n1+1;
    end

    if actual==ventana(1,2)
        prob_actual=prob_actual-beta.*n1;
        n1=n1+1;
    end

    if actual==ventana(1,3)
        prob_actual=prob_actual-beta.*n1;
        n1=n1+1;
    end

    if actual==ventana(3,1)
        prob_actual=prob_actual-beta.*n1;
        n1=n1+1;
    end

    if actual==ventana(3,2)
        prob_actual=prob_actual-beta.*n1;
        n1=n1+1;
    end

    if actual==ventana(3,3)
        prob_actual=prob_actual-beta.*n1;
        n1=n1+1;
    end

    if actual==ventana(2,1)
        prob_actual=prob_actual-beta.*n1;
        n1=n1+1;
    end

```

```
    if actual==ventana(2,3)
        prob_actual=prob_actual-beta.*n1;
        n1=n1+1;
    end

%-----ANALISIS PARA ELECCION-----

    if eleccion==ventana(1,1)
        prob_eleccion=prob_eleccion-beta.*n2;
        n2=n2+1;
    end

    if eleccion==ventana(1,2)
        prob_eleccion=prob_eleccion-beta.*n2;
        n2=n2+1;
    end

    if eleccion==ventana(1,3)
        prob_eleccion=prob_eleccion-beta.*n2;
        n2=n2+1;
    end

    if eleccion==ventana(3,1)
        prob_eleccion=prob_eleccion-beta.*n2;
        n2=n2+1;
    end

    if eleccion==ventana(3,2)
        prob_eleccion=prob_eleccion-beta.*n2;
        n2=n2+1;
    end

    if eleccion==ventana(3,3)
        prob_eleccion=prob_eleccion-beta.*n2;
        n2=n2+1;
    end

    if eleccion==ventana(2,1)
        prob_eleccion=prob_eleccion-beta.*n2;
        n2=n2+1;
    end

    if eleccion==ventana(2,3)
```



```

        prob_eleccion=prob_eleccion-beta.*n2;
        n2=n2+1;
    end

%-----FIN DE ANALISIS-----

    %calculando la variacion de energia local
    %asociada a un site r

    N_gris=double(imagen(x1,y1));

    if N_gris==0
        N_gris=N_gris+1;
    end

    Energia_Actual=Uxw(actual,:);
    Energia_Eleccion=Uxw(eleccion,:);

    prob_actual=prob_actual+ Energia_Actual(1,N_gris);
    prob_eleccion=prob_eleccion+Energia_Eleccion(1,N_gris);

    %criterio de metropolis
    delta=prob_eleccion-prob_actual;
    if delta<=0
        Y(x1,y1)=eleccion;
    else
        p=exp(-delta/T);
        %sorteo variable aleatoria para el rango [0,1]
        epsilon=rand(1);
        if epsilon<p
            Y(x1,y1)=eleccion;
        end
    end

end

end %fin del
end %barrido de la imagen

prog_text = sprintf(repmat( '\b',1,(itertext>0)*...
    12+ceil(log10(itertext+1)) ));
itertext = itertext + 1;
fprintf( '%s%d iterations\n',prog_text,itertext );

```

```
T=0.95*T ;%decremento de temperatura
figure(iter+1)
imshow(gray2ind(Y));

end %fin de iteraciones

tfinal=clock;
tfinal(1,4)=60.*60.*tfinal(1,4);
tfinal(1,5)=60.*tfinal(1,5);
tfinal=tfinal(1,4)+tfinal(1,5)+tfinal(1,6);
tiempo=tfinal-tinicial
clases
Y=uint8(Y);
%-----TERMINA markov8c
```

Bibliografía

- [1] B. Sing-Tze, “Pattern recognition and image processing”, Marcel Dekker Inc., New York, 1992.
- [2] J.M.Laferté, F.Heitz & P.Pérez, “A multiresolution EM algorithm for unsupervised image classification”, Proceedings of the ICPR 96,pp.849-853,1996.
- [3] T.M.Pellizzeri, F. Dell’Acqua, P. Gamba, P. Lornbardo & D. Mazola “Multiband SAR classification using contextual analysis: annealing segmentation vs a neural kernel-based approach”, Geoscience and Remote Sensing Symposium IGARSS ’02. IEEE International, Volume:5, Pages:2593-2595, 2002.
- [4] J. R. Jensen “Introductory Digital Image Processing, A Remote Sensing Perspective ”, Prentice Hall, 1986.
- [5] N. Williams & R.A. Vaughan “Textural segmentation of SAR images”, Progress in Environmental Remote Sensing Research and Applications, Parlow, 1996.
- [6] J.S. Weszka, C.R. Dyen & A. Rosenfield “A comparative study of Texture Measures for Terrain Classification”, IEEE Transactions of Systems, Man and Cybernetics, vol. smc-6, No.4, 1976.
- [7] A. Leon García “Probability and Random Processes for Electrical Engineering”, Segunda Edición, Ed. Addison Wesley, 1994.
- [8] A. Papoulis, “Probability, Random Variables and Stochastic Processes”, Tercera Edición, MacGraw Hill, 1991.
- [9] D.D. Wackerly, “Estadística Matemática con Aplicaciones”, Sexta Edición, Thomson, 2002.
- [10] G. Celeux & J. Diebolt, “L’algorithme SEM: an algorithme d’apprentissage probabiliste pour la reconnaissance de mélanges de densités”, Revue de Statistique Appliquée, Vol. 34, no.2, 1986.

- [11] M.M. Dempster, N. M. Laird & D.B. Rubin, "Maximum Likelihood from incomplete data via the EM algorithm", *Journal of the Royal Statistical Society, Series B*, vol. 39, pp. 1-38, 1977.
- [12] N.J. Nilsson, "Introduction to Machine Learning", Department of Computer Science, Stanford University, 1996.
- [13] G.Celeux & J.Diebolt, "Comportement asymptotique d'un algorithme d'apprentissage probabilistic pour la reconnaissance de mélanges de densités", *Tech. Rep. INRIA no. 563*, 1986.
- [14] Z.S. Li, "Markov Random Field Modeling in Image Analysis", Ed. Springer Verlag, 2001.
- [15] K.A. Downsland & B. Adenso-Díaz, "Diseño de Heurísticas y Fundamentos de Recocido Simulado", *Revista Iberoamericana de Inteligencia Artificial*, No. 20, pág. 34-52, 2001.
- [16] J. Lira "La Percepción Remota - Nuestros ojos desde el espacio", *Fonde de Cultura Económica, La ciencia para todos No. 33*, pág. 106-147, 1997.
- [17] J. Vandeneede, D. Zhang, P. Wambacq & L. Van Gool "Supervised classification of remote sensing images with the aid of rebust statistics and the Gibbs distribution", *Katholieke Universiteit Leuven Technical Report KUL/ESAT/MI2/9421*, pág. 10-12, Noviembre 1994.
- [18] J. Pascale Masson & Wojciech Pieczynski "SEM Algorithm and Un-supervised Statistical Segmentation of Satellite Images", *IEEE Transaction on Geoscience and Remote Sensing*, Vol. 31, No.3 pág. 618-633, Mayo 1993.
- [19] J. Nicole Marhic, Pascale Masson & Wojciech Pieczynski "Melange de Lois et Segmentation non Supervisee de Donnees Spot", *Statistique et Analyse des Données* Vol. 16, No.2 pág. 59-79, 1991.
- [20] T. Baidyk, E. Kussul, O. Makeyev & A. Martín "Reconocimiento de texturas de superficies metálicas", *SOMI XX, Congreso de instrumentación*, Octubre 2005.
- [21] T. Baidyk, E. Kussul & O. Makeyev "Texture Recognition with Random Subspace Neural Classifier", *WSEAS Trans. on Circuits and Systems*, Issue 4, Vol. 4, Abril 2005.
- [22] P. Wadhams, F. Parmiggiani & G. De Carolis "The use of SAR to Measure Ocean Wave Dispersion in Frazil-Pancake Icefields", *American Meteorological Society, Journal of Physical Oceanography*, pág. 1721-1746, Junio 2002.