



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
PROGRAMA DE MAESTRÍA Y DOCTORADO EN INGENIERÍA
INGENIERÍA ELÉCTRICA – PROCESAMIENTO DIGITAL DE
SEÑALES**

***Proyecto correspondiente al
Examen General de Conocimientos***

***Alumno: Hernández Delgado Armando Salomón
Revisor: Dr. Fernando Arámbula Cosío***

Fecha de entrega 17 julio de 2013.

Contenido

PLANTEAMIENTO DEL PROBLEMA	1
1. INTRODUCCIÓN.	2
1.1 Segmentación de imágenes.	2
1.2 Preprocesamiento.	3
1.2.1 Filtros lineales espaciales.	3
1.2.2 Filtros pasobajas (reducción de ruido/suavizado de imagen).	4
Filtro de media.	4
Filtro de mediana.	5
Filtro Gaussiano.	6
Filtro Laplaciano de Gaussiana (LoG).	10
1.3 Enfoques de segmentación.	14
1.3.1 Umbralización.	15
1.3.2 Crecimiento de regiones.	16
1.3.3 Agrupamiento por K-medias.	16
1.3.4 Enfoque estadístico.	17
Clasificador Bayesiano.	18
2. METODOLOGÍA.	21
3. DESARROLLO.	23
4. RESULTADOS.	29
5. CONCLUSIONES.	40
6. BIBLIOGRAFÍA.	41

Planteamiento del problema

Programar un clasificador de Bayes en MATLAB, para segmentación de los cartílagos de la rodilla en imágenes de RMN.

Entrenar un clasificador de pixeles cartílago/fondo basado en 4 funciones discriminantes de Bayes: $Y_{\text{cartílago}}$, Y_{hueso} , $Y_{\text{músculo}}$, Y_{tendones} . Utilizar como características los valores (x, y, g) de cada pixel de cada una de las 4 clases. Donde x, y son las coordenadas del pixel y g es el nivel de gris del mismo.

Entrenar con el método dejar-uno-fuera (LEAVE ONE OUT)

Utilizar las 9 imágenes proporcionadas como sigue:

1. Identificar una escala apropiada y filtrar todas las imágenes con un filtro Gaussiano.
2. Clasificador de Bayes. Utilizar una imagen para validación y las otras 8 para entrenamiento, repetir el proceso 9 veces con imágenes diferentes de entrenamiento/validación.

1. INTRODUCCIÓN.

1.1 Segmentación de imágenes.

La segmentación de imágenes es una técnica de procesamiento que se refiere a la extracción de información útil de una escena para facilitar su observación y análisis. El objetivo principal del proceso de segmentación es particionar una imagen en diversas regiones también llamadas clases o subimágenes, que son homogéneas con respecto a una o más características o atributos.

La segmentación es una herramienta importante en procesamiento de imágenes médicas y ha sido útil en varias aplicaciones, tales como: cuantificación de lesiones en esclerosis múltiple, planeación y simulación de cirugías, medición de volumen de tumores, clasificación automática de células en la sangre, estudios de desarrollo cerebral, detección de microcalcificación en mastografías, detección de tumores, etc. También es importante para extracción de atributos en una imagen, medición y despliegue; en otras aplicaciones es útil para diferenciar regiones anatómicas, órganos, músculos y sistema circulatorio de regiones con alguna patología como tumores, deformidades o lesiones. En algunos otros estudios el objetivo es dividir la imagen completa en secciones representativas o morfológicas, y en otros se debe extraer por ejemplo una sección específica [2].

Se enfrentan tres problemas principales en la segmentación de imágenes médicas vinculados a su adquisición y a la fuente que la genera:

- Las imágenes contienen gran cantidad de ruido, lo cual dificulta la clasificación de píxeles por intensidad.
- La intensidad en los tejidos y órganos, no es uniforme.
- La intensidad de un pixel está indirectamente relacionado con el tejido de interés pero también con otros que se desean suprimir.

Tomando en cuenta estos retos, se han empleado diversas técnicas de segmentación y combinación de éstas, sin embargo no hay una técnica estándar que produzca resultados satisfactorios para todas las aplicaciones. La definición del objetivo de la segmentación varía de acuerdo al objetivo del estudio o aplicación y al tipo de imagen.

1.2 Preprocesamiento.

Con frecuencia, las imágenes médicas contienen bajo contraste, contornos borrosos. También es común que los modos del histograma correspondientes a los diferentes tipos de regiones en la imagen se traslapen, haciendo que la segmentación por umbralización no sea una opción. La aplicación de técnicas de preprocesamiento puede ayudar a diferenciar los modos en el histograma y homogeneizar regiones con características similares con la finalidad de mejorar el desempeño de una segmentación posterior.

1.2.1 Filtros lineales espaciales.

Se denomina operaciones por vecindad a aquellas técnicas de procesamiento en las que el valor resultante de un pixel de coordenadas (x_0, y_0) , el cual llamaremos el pixel referencia, es función del valor original del pixel en ese punto así como de los valores de los pixeles vecinos. La forma en la que los valores se combinan para producir el resultado puede variar significativamente entre algoritmos. Muchos algoritmos trabajan en forma lineal y usan la convolución en dos dimensiones, mientras que otros procesan los valores de una forma no lineal. Al tipo de operaciones que se realizan directamente en los pixeles de una imagen se denominan filtros espaciales, para diferenciarlos de los filtros en el dominio de la frecuencia.

La mecánica del filtrado lineal espacial consiste en mover el centro de la máscara filtro (kernel o ventana) de punto a punto en la imagen, realizando en cada punto la suma de productos de los coeficientes del kernel con los pixeles vecinos, contenidos en el área cubierta por el kernel del filtro, como se muestra gráficamente en la figura 1.

Existen dos conceptos estrechamente relacionados al realizar filtrado lineal espacial: la correlación y la convolución. La correlación es el proceso de realizar la operación anteriormente descrita con el kernel y la imagen. La convolución es operacionalmente el mismo proceso excepto que el kernel es reflejado en ambas direcciones, horizontal y vertical.

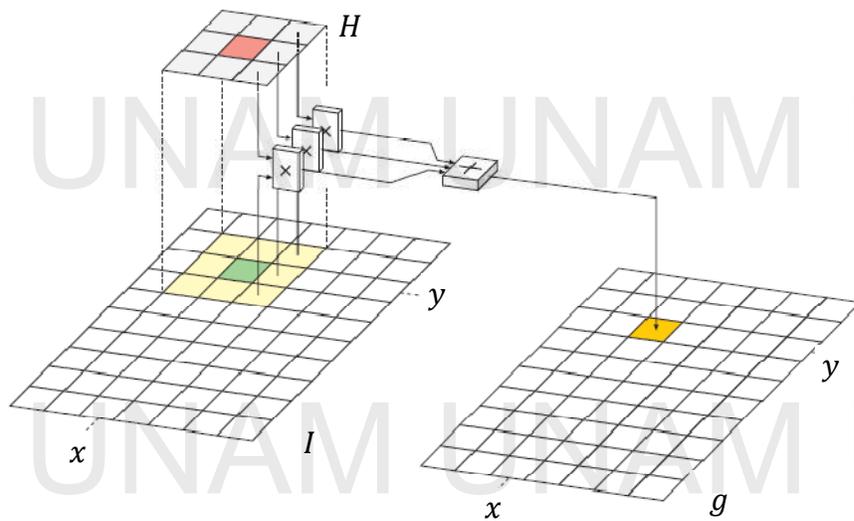


Figura 1. Proceso de obtención de los pixeles procesados de la imagen g , por medio de la suma de productos de los valores de pixeles de la imagen original I y los coeficientes del kernel H . Imagen adaptada de [5].

1.2.2 Filtros pasobajas (reducción de ruido/suavizado de imagen).

Uno de los usos principales del filtrado en imágenes es para eliminar ruido de la imagen; homogeneizar regiones afectadas por ruido, suavizando el gradiente de intensidad de la imagen, que puede ser sometida a un proceso posterior de procesamiento mejorando el desempeño de éste. A continuación se describen los filtros más comunes, es una lista no exhaustiva pero representante.

Filtro de media.

Una técnica de filtro pasobajas o suavizado de imagen es empleando un filtro promediador (**filtro de media**), o un filtro de mediana. El filtro de media reemplaza el valor de cada pixel por el promedio de los valores de intensidad de los pixeles vecinos. Este filtro es razonablemente efectivo para eliminar ruido Gaussiano pero con el costo de perder detalles finos, es decir, atenuando las altas frecuencias. Es evidente también que este filtro no es efectivo para remover ruido de tipo 'salt & pepper'. En resumen, las desventajas del filtro de media son: poca robustez para eliminar ruido causado por valores atípicos (outliers), cuando actúa en bordes finos o bien definidos, los hará borrosos. En la figura 2 se muestra el efecto de un filtro de media o promediador.

Imagen original



Imagen filtrada

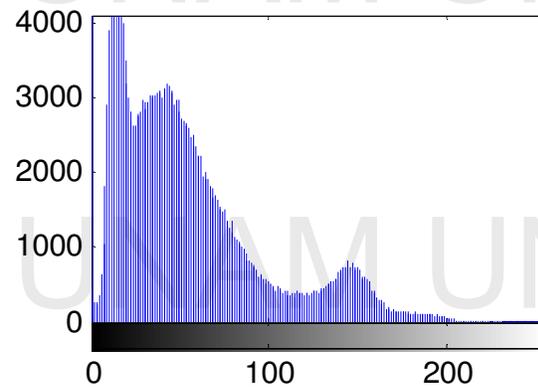
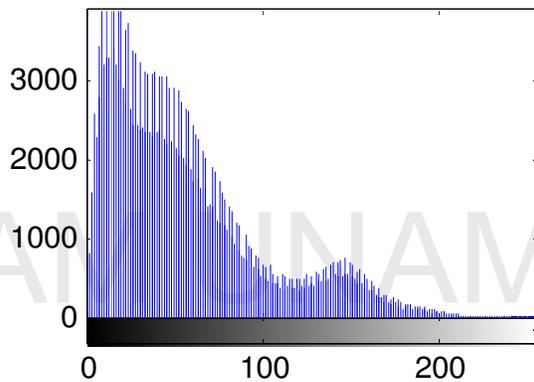


Figura 2. Imagen filtrada con filtro promediador de tamaño 3x3 y su histograma correspondiente.

Filtro de mediana.

En el filtro **de mediana**, el valor de cada pixel se reemplaza por la mediana del valor de intensidad de los pixeles vecinos. El filtro de mediana a diferencia del filtro de media no difumina los bordes de las regiones más grandes que el kernel del filtro, mientras que suaviza pequeñas variaciones de textura, preserva las altas frecuencias y elimina efectivamente el ruido impulsivo (como el tipo salt & pepper). El filtro de mediana agudiza las crestas en el histograma de la imagen permitiendo distinguir los modos y facilitando la elección de un umbral. El filtro de mediana es un tipo de filtro espacial no lineal [1]. En la figura 3 se muestra el efecto de aplicar un filtro de mediana

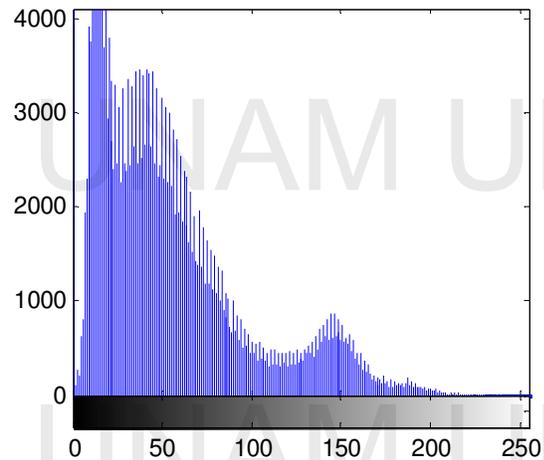


Figura 3. Imagen filtrada con filtro de mediana de tamaño 3x3

Filtro Gaussiano.

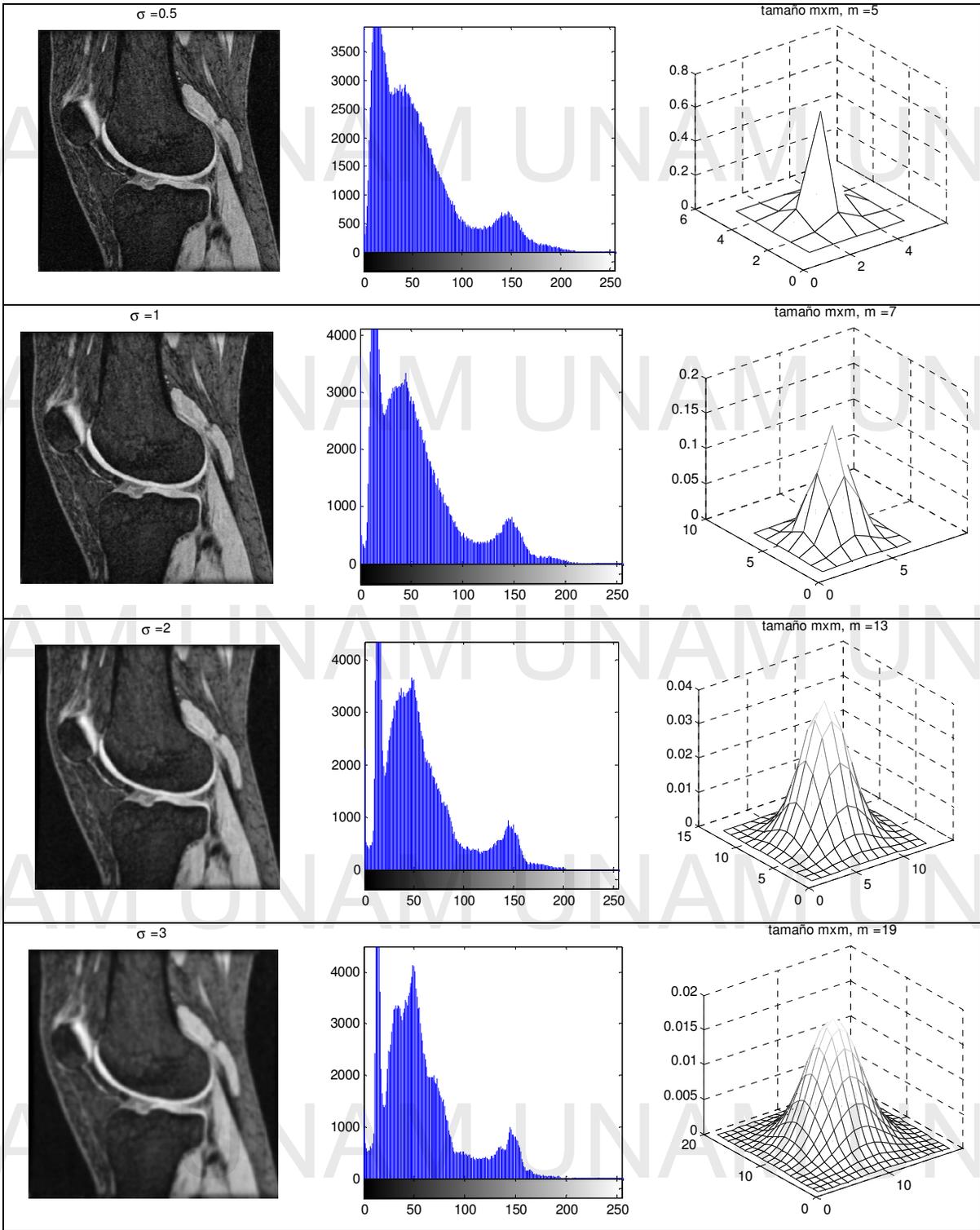
El filtro **Gaussiano** es un filtro suavizador de imagen; los valores de los coeficientes $[x, y]$ del kernel $g(x, y)$ están basados en una función Gaussiana

$$g(x, y) = e^{-\frac{(x^2 + y^2)}{2\sigma^2}} \quad (\text{ec. 1})$$

Donde σ es el parámetro de dispersión (desviación estándar) que define el grado de suavizado. Un σ grande implica un filtro Gaussiano amplio y por tanto un nivel grande de suavizado. El kernel de un filtro Gaussiano con un valor σ relativamente grande es un ejemplo de filtro paso bajas en el que el contenido de alta frecuencia espacial de una imagen (por ejemplo bordes nítidos) se suprime. En la figura 4 se muestra el resultado de filtrar una imagen de trabajo con un kernel Gaussiano con diferentes valores de σ y de diferente tamaño. Para establecer el tamaño del kernel, se establece una relación entre σ y el tamaño del kernel $m \times m$ de la siguiente forma

$$m = [\sigma \times 3] \times 2 + 1$$

Donde $[a]$ significa el redondeo del número a al entero cercano más grande.



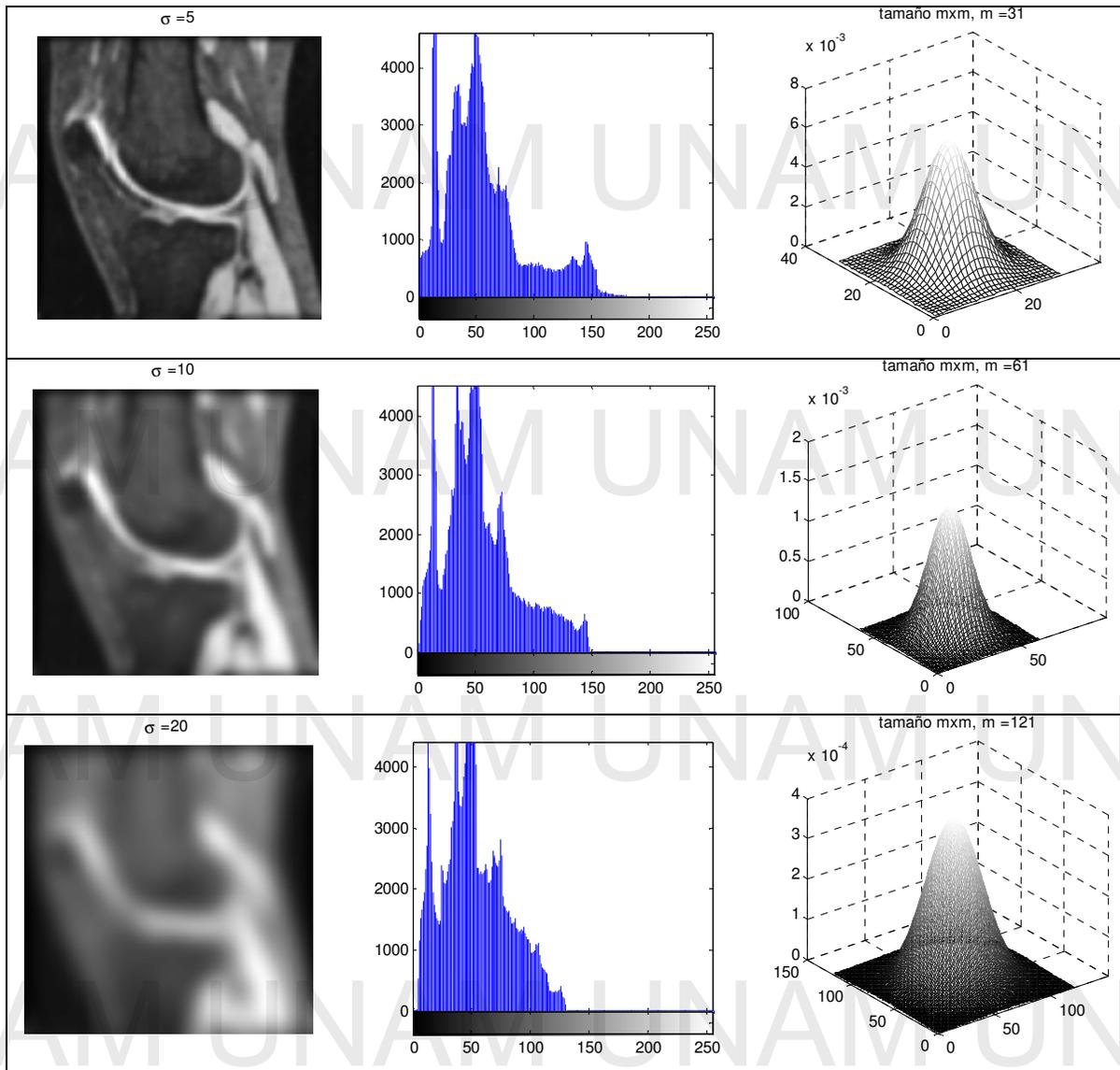


Figura 4. Imagen filtrada con kernel Gaussiano de diferentes tamaños $m \times m$, y su histograma correspondiente.

El operador Laplaciano ∇^2 es un operador de una función de dos dimensiones $f(x, y)$ y se define como la suma de las segundas derivadas parciales sobre las dos direcciones x e y :

$$\nabla^2 f(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

Y la forma discreta del filtro es:

$$\nabla^2 f(x, y) = f(x + 1, y) + f(x - 1, y) + f(x, y + 1) + f(x, y - 1)$$

La segunda derivada de una función mide su curvatura local. El efecto de aplicarlo a una imagen es que los bordes se encuentran en los cruces por cero de la segunda derivada, como se muestra en la figura 5.

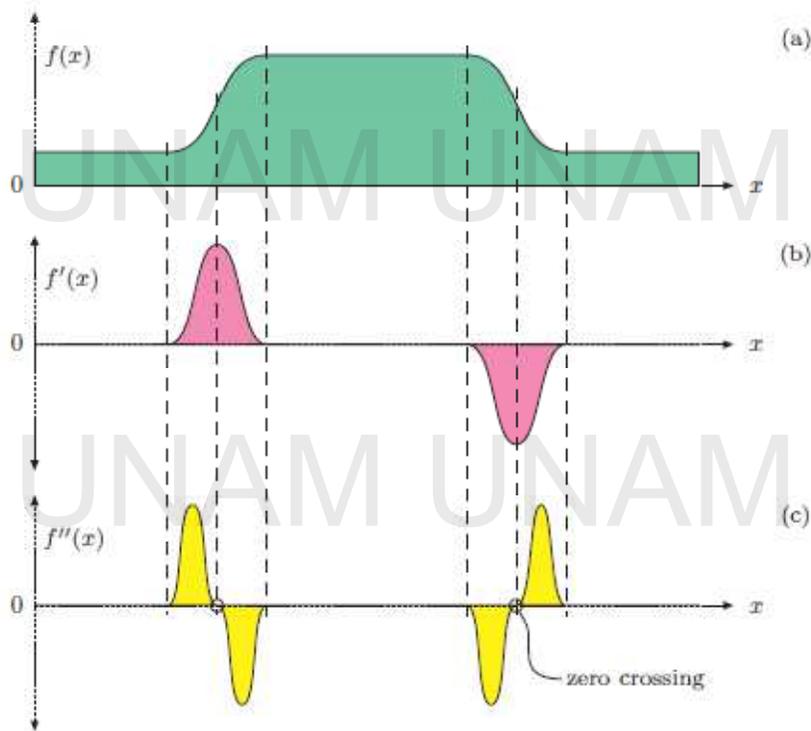


Figura 5. Principio de detección de bordes por medio de la segunda derivada. Se muestra una función en una dimensión. a) Función original, b) primera derivada, c) segunda derivada, mostrando que los puntos significativos de bordes de la función se encuentran en los cruces por cero de la segunda derivada. Imagen tomada de [5]

El filtro laplaciano por lo tanto detecta bordes en la imagen. El filtro responde solo a los cambios finos en la imagen, por ejemplo las regiones de la imagen donde el gradiente de la intensidad es significativo, por lo que también es muy sensible al ruido, por lo que el efecto de sensibilidad al ruido puede reducirse suavizando la imagen. Una forma de realizar el suavizado de la imagen y a la vez detectar los bordes con un solo operador es combinando el filtro laplaciano y el filtro pasobajas.

Filtro Laplaciano de Gaussiana (LoG).

Dado que la segunda derivada por lo general tiende a amplificar el ruido en la imagen, es común aplicar algún tipo de suavizado previo con filtros paso bajas. Un operador lineal que combina tanto el operador de segunda derivada como de filtro pasobajas es el Laplaciano de Gaussiana.

Aplicando el operador Laplaciano a la función (ec.1) se obtiene el operador laplaciano de Gaussiana [6]:

$$\nabla^2 g(x, y) = \left[\frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} \right] e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Como ejemplo del resultado de aplicar un filtro Laplaciano de Gaussiana a una imagen, se empleará una imagen de trabajo con diferentes valores de σ . EL resultado se muestra en la figura 6.

Filtrado Gaussiano, $\sigma = 3$



Figura 6a. Imagen con los bordes sobrepuestos obtenidos con un filtro LoG de $\sigma = 3$

Filtrado Gaussiano, $\sigma = 4$



Figura 6b. Imagen con los bordes sobrepuestos obtenidos con un filtro LoG de $\sigma = 4$

Filtrado Gaussiano, $\sigma = 5$

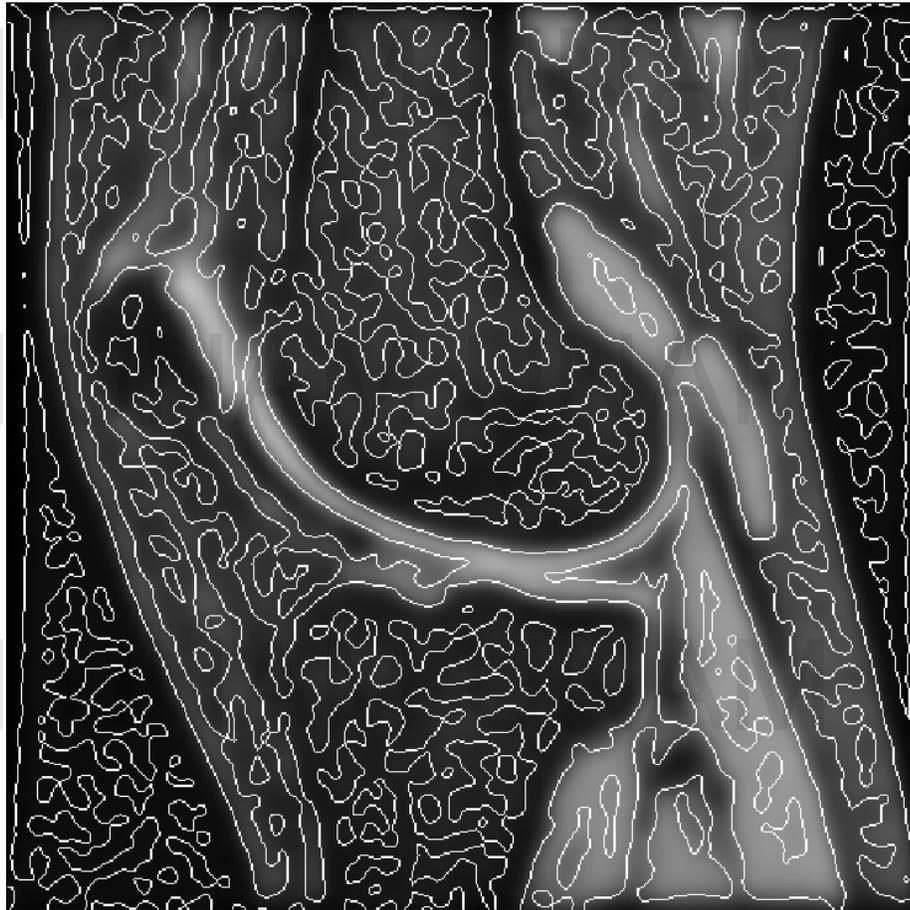


Figura 6c. Imagen con los bordes sobrepuestos obtenidos con un filtro LoG de $\sigma = 5$.

Como se puede apreciar en las figuras, el resultado de aplicar el filtro LoG radica en que los bordes obtenidos son regiones cerradas, donde la intensidad del nivel de gris mantiene una misma dispersión. Estos bordes se obtienen a partir de realizar la operación Laplaciano a la imagen filtrada por el filtro Gaussiano. Por lo tanto, los bordes obtenidos separan las regiones con diferente dispersión del nivel de gris. Para este trabajo se desea que la región del cartílago sea una región cerrada y con niveles de gris lo más homogénea posible.

El suavizado de las imágenes con las que se trabajan se realiza con el filtro Gaussiano. Para definir el nivel de dispersión σ sin perder los bordes del cartílago, se emplea el filtro LoG.

1.3 Enfoques de segmentación.

Los primeros enfoques de segmentación de imágenes digitales se basaron en los niveles de gris de los píxeles que forman la imagen. Estos niveles de gris son una traducción digital de un flujo energético recibido por el sensor para una determinada banda del espectro. Debido a esto, una segmentación basada exclusivamente en esta característica puede denominarse propiamente segmentación espectral. Aún cuando esta información es valiosa no siempre es capaz de discriminar entre clases, ya que en una escena puede haber varias clases con un comportamiento espectral similar. En este caso una opción ampliamente usada es la de incorporar a la segmentación el contexto espacial en que aparece cada píxel.

Tradicionalmente se han dividido los métodos de segmentación en supervisado y no supervisado de acuerdo a cómo son obtenidos los atributos de homogeneidad entre los píxeles. El método supervisado parte de un conocimiento previo de la imagen, de la cual se seleccionan muestras para cada una de las clases. Por otra parte el método no supervisado primero procede a una búsqueda automática de grupos de valores de píxel homogéneos dentro de la imagen.

De este modo el método supervisado pretende definir clases informacionales, mientras que el no supervisado tiende a identificar las clases espectrales presentes en la imagen. Ninguno de los dos métodos proporciona una solución inmediata a todos los problemas de segmentación. Por una parte el método supervisado podría forzar al algoritmo a discriminar categorías que no tengan un claro significado espectral, mientras que el método no supervisado puede proporcionar resultados de difícil interpretación, poco relacionados con las necesidades finales del usuario [4].

Ahora bien, la clasificación de las técnicas de segmentación varía de una bibliografía a otra. Tomando como referencia [2], las técnicas frecuentemente empleadas se clasifican en dos amplias categorías: (1) Las técnicas de segmentación de regiones, en las que se buscan regiones que satisfacen un

criterio de homogeneidad predefinido y, (2) técnicas de segmentación basadas en bordes, en las que se buscan bordes entre regiones con características diferentes. A continuación se describen algunas técnicas comunes empleadas en la segmentación de imágenes.

1.3.1 Umbralización.

Un método común de segmentación por regiones es la umbralización, en esta técnica se selecciona un umbral T , se realiza una comparación de nivel de intensidad píxel a píxel con el umbral, y se divide en grupos de píxeles con valores de intensidad menores, y mayores o iguales al umbral.

La Umbralización Global se basa en la suposición de que la imagen tiene un histograma bimodal, es decir, los píxeles del objeto de interés y el fondo tienen niveles de gris fácilmente discernibles, por lo que el objeto puede ser extraído con la simple operación de comparar los valores de gris de los píxeles con el valor umbral T que separa ambos modos.

Otras técnicas para seleccionar el umbral global consisten en optimizar un parámetro, por ejemplo, minimizar la probabilidad de error de clasificar un píxel como perteneciente al objeto, cuando en realidad pertenece al fondo, o el caso contrario, clasificar un píxel como perteneciente al fondo cuando en realidad pertenece al objeto.

La umbralización global es computacionalmente simple y rápida, funciona en un nivel aceptable en imágenes con regiones de intensidad uniforme y un fondo contrastante, sin embargo tiene pobre desempeño en imágenes con poco contraste entre el objeto y el fondo o si el fondo tiene una intensidad variable en toda la imagen.

Este enfoque no toma en cuenta la información de la vecindad del píxel bajo prueba, lo que puede llevar a problemas de sobresegmentación, resultando en imágenes segmentadas con una alta incidencia de píxeles aislados, causado posiblemente, por la presencia de ruido en la imagen, por lo que se suele suavizar antes la imagen para reducir los efectos de píxeles aislados, pero con el problema subyacente de que los filtros suavizadores difuminan los bordes, provocando en la segmentación error en la delimitación del objeto [2].

Hasta aquí se ha considerado a la Umbralización solo con un umbral, el cual tiene un funcionamiento aceptable siempre y cuando la distribución del histograma sea bimodal, por lo que el umbral debe ser colocado entre ambos modos.

Sin embargo, hay una gran cantidad de imágenes que no tienen un comportamiento análogo, por lo que se tiene que recurrir a una umbralización tipo multinivel, donde la cantidad de umbrales dependerá del comportamiento del histograma. Este último caso acarrea un problema considerable para el usuario, ya que la selección de varios umbrales de forma manual se puede convertir en una tarea ardua y compleja.

1.3.2 Crecimiento de regiones.

Mientras la umbralización actúa individualmente en cada pixel, el crecimiento de regiones lo hace en grupos de pixeles con intensidades similares. El crecimiento de regiones comienza con un pixel o un grupo de pixeles llamados semillas que pertenecen a la estructura de interés. Las semillas pueden ser elegidas por un usuario o con un método automático. En el siguiente paso se examinan los pixeles vecinos individualmente y se añaden a la región en crecimiento solo si son suficientemente similares, basándose en un criterio de similaridad. El procedimiento continúa hasta que no hay más pixeles por añadir.

La ventaja de crecimiento por regiones es que es capaz de segmentar correctamente regiones que tienen las mismas propiedades y están separadas espacialmente, así como generar regiones conectadas.

El resultado depende fuertemente de la selección del criterio de similaridad. Otro problema es que diferentes puntos de inicio podrían no crecer y terminar en regiones idénticas [2].

1.3.3 Agrupamiento por K-medias.

El agrupamiento por K-medias [7] es un método no supervisado de clasificación. Busca el agrupamiento natural en el conjunto de datos (pixeles) empleando una medida de similaridad o de diferencia. Dos estrategias de implementación de K-medias son: por error cuadrático y por mezcla de modelos. La suma del cuadrado de la distancia euclidiana entre los elementos de un grupo y el centroide del grupo

es llamada variación intra-grupo. Se usa como criterio la minimización de la suma de las variaciones:

$$Q = \sum_{j=1}^K \sum_{i=1}^n \|v_i^j - \mu_j\|^2$$

Donde v_i^j es el elemento i de la clase j y μ_j el centroide de la clase j definido como la media de v_i perteneciente a la clase K_j .

El algoritmo para segmentación de una imagen sigue los siguientes pasos:

1. Proporcionar el número de clases K y los centroides μ_j
2. Asignar cada pixel al grupo cuyo centroide es más cercano
3. Después de que todos los pixeles fueron asignados recalcular los centroides con los nuevos grupos.
4. Repetir los pasos 2 y 3 hasta que no haya una diferencia significativa en los centroides.

1.3.4 Enfoque estadístico.

Se puede minimizar el número de pixeles erróneamente clasificados si se tiene algún conocimiento a priori sobre la distribución de los niveles de gris que componen tanto al objeto de interés y el fondo o aquello que no interesa [3]. De esta forma, al reducir el error estadístico de clasificar erróneamente los pixeles del objeto y de su entorno empleando un criterio de minimización, se obtiene un detector óptimo.

Patrones y patrones clase.

Un patrón es un arreglo de descriptores o atributos. Una clase patrón es un conjunto de patrones que comparten algunas propiedades comunes. Las clases patrón se denotan como $\omega_1, \omega_2, \dots, \omega_W$, donde W es el número de clases. El reconocimiento de patrones involucra diversas técnicas para asignar un patrón a su respectiva clase de forma automática y con mínima intervención humana.

Un enfoque de clasificación de patrones empleada para segmentación es el probabilístico. La clasificación se basa en teoría de probabilidad y con el principio

de escoger la opción más probable o con el menor riesgo de clasificar un patrón incorrectamente.

Clasificador Bayesiano.

La clasificación consiste en la tarea de etiquetar vectores de atributos (muestras) de k diferentes clases. Por convención, el vector de atributos se denota por $x = [x_1, x_2, \dots, x_n]^T$ donde n es la dimensión o dimensionalidad del vector de atributos, es decir, un vector de atributos contiene d elementos que caracterizan a la muestra bajo observación, permitiendo que sea posible agrupar muestras con atributos similares y separar aquellas que no lo son.

La probabilidad de que un vector x pertenezca a la clase ω_k es $P(\omega_k|x)$ y se hace referencia a esta probabilidad como la probabilidad a posteriori, una vez que se conoce un conjunto de datos de referencia correspondiente a cada clase, llamado conjunto de entrenamiento denotado por $p(x|\omega_k)$ y corresponde a la función de densidad de probabilidad en el espacio de atributos, de la clase ω_k . La relación entre estas dos probabilidades se establece con la regla de Bayes:

$$P(\omega_k|x) = \frac{p(x|\omega_k) P(\omega_k)}{p(x)}$$

En donde $P(\omega_k)$ es la probabilidad a priori de la clase ω_k . Si esta probabilidad se desconoce se pueden estimar a partir del conjunto de datos de ejemplo o entrenamiento. El divisor:

$$p(x) = \sum_{i=1}^k p(x|\omega_i) P(\omega_i)$$

Corresponde a la probabilidad total de la variable multidimensional x , el cual asegura que con cada nueva muestra observada, las probabilidades a posterior sean escaladas con tal de que su sumatoria sea uno.

Elegir la clase con la máxima probabilidad a posteriori una vez observada una nueva muestra x , se minimiza la probabilidad de error. Uno de los problemas con este clasificador es el desconocimiento de la función de densidad de probabilidad

condicional $p(\mathbf{x}|\omega_k)$ la cual describe el tipo de distribución que tienen los vectores de atributos n -dimensionales en el espacio de atributos, tan solo para una clase en particular. Esta distribución puede ser estimada por diversos métodos. El método empleado en este trabajo es con la mezcla de funciones de probabilidad Gaussianas.

Las distribuciones Gaussianas usualmente son buenas aproximaciones para modelar la forma de las clases en un espacio de atributos adecuado, en el sentido de que los vectores de atributos de cada clase generan clases mutuamente excluyentes. En algunos casos, los vectores que provienen de diferente clase muestran un grado de traslape en el espacio de atributos. Por tal motivo, no siempre se puede garantizar que la clasificación estará libre de errores empleando distribuciones Gaussianas.

El clasificador bayesiano que asigna una pérdida homogénea a los errores de clasificación, emplea funciones de decisión de la forma:

$$d_j(\mathbf{x}) = p(\mathbf{x}|\omega_j) P(\omega_j)$$

Para $j = 1, 2, \dots, W$ clases y vectores \mathbf{x} de dimensión n . El proceso de clasificación consiste en calcular W funciones de decisión y luego, asignar al patrón la clase cuya función de decisión produjo el valor máximo. En caso de valores iguales, la asignación se resuelve escogiendo al azar alguno de esos valores.

La función de densidad Gaussiana para una variable (vector) \mathbf{x} n -dimensional, digamos para la j -ésima clase patrón es:

$$p(\mathbf{x}|\omega_j) = \frac{1}{(2\pi)^{n/2} |\mathbf{C}_j|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_j)^T \mathbf{C}_j^{-1}(\mathbf{x}-\boldsymbol{\mu}_j)}$$

Donde $\boldsymbol{\mu}_j$ y \mathbf{C}_j son correspondientemente la media de dimensión n y la matriz de covarianza de dimensión $n \times n$. El término $|\mathbf{C}_j|$ corresponde al determinante de \mathbf{C}_j . Dado que la función logaritmo natural es una función monótonicamente creciente, escoger la máxima función $d_j(\mathbf{x})$ para clasificar un patrón equivale a escoger la

máxima función $\ln [d_j(\mathbf{x})]$ de tal forma que se pueden redefinir las funciones de decisión de la siguiente forma:

$$d_j(\mathbf{x}) = \ln [p(\mathbf{x}|\omega_j) P(\omega_j)]$$
$$d_j(\mathbf{x}) = \ln [p(\mathbf{x}|\omega_j)] + \ln [P(\omega_j)]$$

Sustituyendo $p(\mathbf{x}|\omega_j)$ en la ecuación anterior por la expresión del modelo Gaussiano se obtiene la ecuación:

$$d_j(\mathbf{x}) = \ln P(\omega_j) - \frac{n}{2} \ln 2\pi - \frac{1}{2} \ln |\mathbf{C}_j| - \frac{1}{2} [(\mathbf{x} - \boldsymbol{\mu}_j)^T \mathbf{C}_j^{-1} (\mathbf{x} - \boldsymbol{\mu}_j)]$$

Para $j = 1, 2, \dots, W$. El término $\frac{n}{2} \ln 2\pi$ es constante en cada función y puede por lo tanto descartarse en el cálculo, con lo que finalmente se obtienen las funciones de decisión:

$$d_j(\mathbf{x}) = \ln P(\omega_j) - \frac{1}{2} \ln |\mathbf{C}_j| - \frac{1}{2} [(\mathbf{x} - \boldsymbol{\mu}_j)^T \mathbf{C}_j^{-1} (\mathbf{x} - \boldsymbol{\mu}_j)]$$

Los parámetros $\boldsymbol{\mu}_j$ y \mathbf{C}_j definen completamente la distribución Gaussiana y frecuentemente son desconocidos. En esta situación son estimados a partir de un conjunto de datos de ejemplo correspondientes a cada clase. Estos datos deben corresponder fielmente a las clases que representan (datos sin ruido). Las probabilidades a priori $P(\omega_j)$ de cada clase se calculan como la relación del número de píxeles por clase entre el total de los píxeles empleados para el entrenamiento.

La estructura del clasificador de Bayes empleando las funciones de decisión anteriores se muestra en la figura 7.

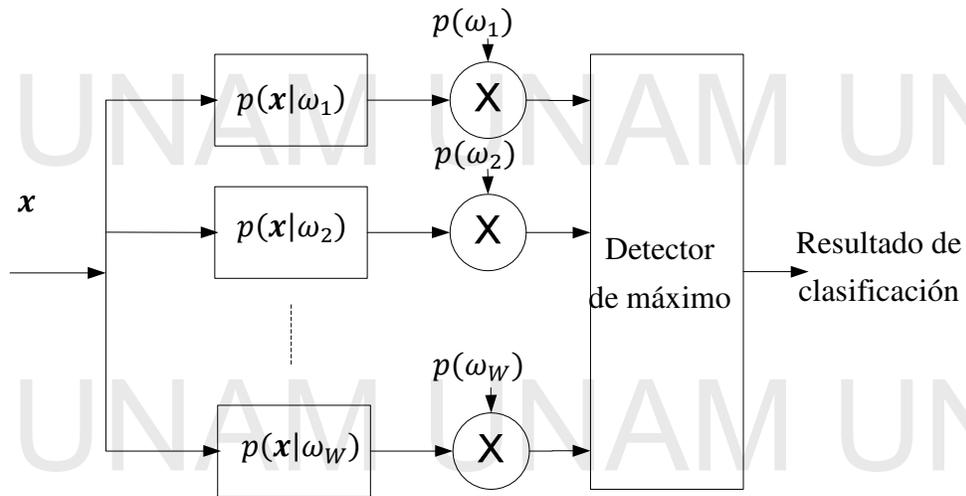


Figura 7. Clasificador de Bayes empleando funciones de decisión de costo uniforme (clasificación MAP)

2. METODOLOGÍA.

Al emplear el enfoque de un clasificador bayesiano, se debe contar con un conjunto de datos de entrenamiento y un conjunto de datos de prueba.

Se dispone de nueve imágenes de RMN de la rodilla. Todas las imágenes se filtran con un filtro Gaussiano. Para definir el nivel de dispersión σ sin perder los bordes del cartílago, se emplea el filtro LoG hasta encontrar un nivel 'adecuado'.

El filtrado de las imágenes se realiza con el archivo llamado *filok.m* donde se especifica el nombre de la imagen que deberá estar en el mismo directorio donde se encuentra este archivo. Las imágenes filtradas se almacenan en otra carpeta del mismo directorio. Para todas las imágenes se escogió un valor de $\sigma = 3$ ya que se percibe que el cartílago es una región cerrada, sin agregar más regiones que no lo son, pero en muchos casos, regiones que no pertenecen al cartílago con niveles similares de gris son incluidas y no se puede evitar esto, incluso con un valor de σ mucho más pequeño.

El procedimiento de entrenamiento-prueba se realiza con el enfoque dejar uno fuera. Para esto se emplearon ocho imágenes para entrenamiento y una para prueba.

Se realizaron nueve experimentos, cada vez cambiando el conjunto de entrenamiento y la imagen de validación, así como seleccionando de nuevo las regiones de entrenamiento para la estimación de los parámetros. Los conjuntos de se muestran en la tabla II.

Tabla II.

No. Experimento	Imágenes de entrenamiento	Imagen de prueba
1	1-8	9
2	1-7,9	8
3	1-6,8,9	7
4	1-5,7-9	6
5	1-4,6-9	5
6	1-3,5-9	4
7	1,2,4-9	3
8	1,3-9	2
9	2-9	1

Se deben generar cuatro funciones de decisión para las regiones de interés: Ycartílago, Yhueso, Ymúsculo, Ytendón. En la figura 8 se muestran las regiones de interés a segmentar.

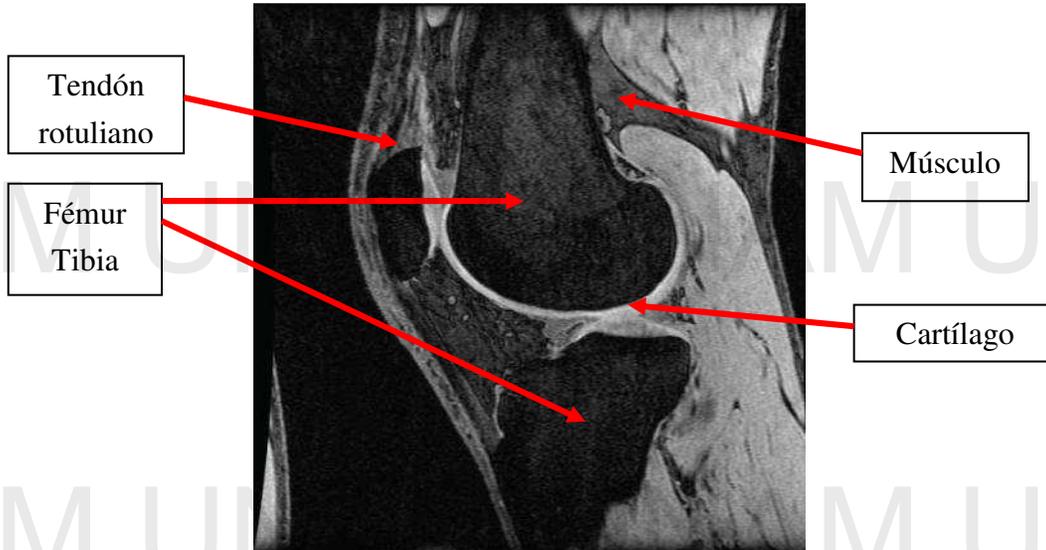


Figura 8. Anatomía de la rodilla, regiones de interés para segmentación.

La obtención de parámetros de modelado de la distribución de cada región de interés, se realiza seleccionando regiones representativas en cada imagen y se almacenan tanto los niveles de gris como las coordenadas de cada pixel, generando los vectores de atributos $[x \ y \ g]$. Una vez generados estos vectores de

atributos de cada región, se calcula el vector de medias (valores medios de cada conjunto de vectores atributo, para cada clase). Ya que se maneja una distribución con tres variables de interés, el vector de atributos es de dimensionalidad tres y generará una matriz de covarianza de 3 x 3. Una vez obtenidos los vectores de medias y la matriz de covarianza de cada clase se generan las funciones de decisión, en las cuales se evaluará cada pixel de la imagen de prueba.

3. DESARROLLO.

El programa en MatLab desarrollado permite seleccionar el número de imágenes de entrenamiento por medio de un cuadro de ingreso de este valor como se presenta en la figura 9.

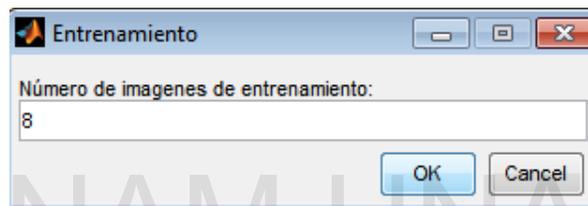


Figura 9. Ingreso del número de imágenes de entrenamiento.

Una vez ingresado esta información, se abrirá una ventana para seleccionar la primera imagen de entrenamiento. Después se irá indicando cuál región debe seleccionarse y el usuario deberá seleccionar una región cerrada representativa de la región indicada. Este proceso se muestra en la figura 10.

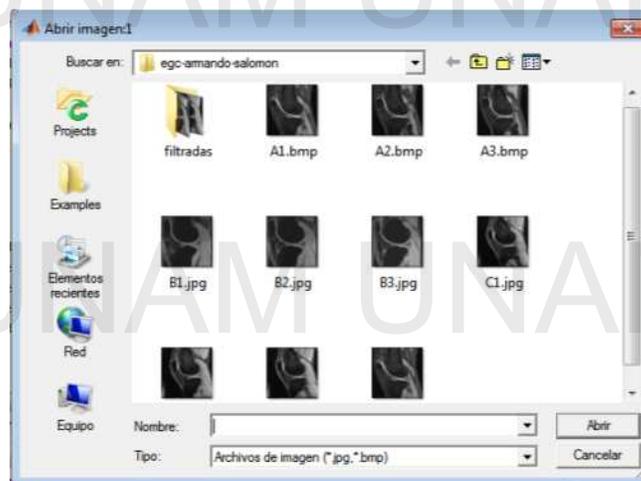


Figura 10. Selección de imagen de entrenamiento y selección de regiones.

Al inicio se decidió realizar para algunas regiones más de una selección de una región cerrada, por ejemplo, para la clase hueso se pide una región de fémur y aparte, una región de tibia, estas dos regiones se almacenan como una sola perteneciente a la clase hueso. Se hizo lo mismo para tendón y cartílago. Ya que esto tomaba más tiempo y que no se observó un mejor desempeño que al usar solo una región por clase, se terminó por seleccionar solo una zona por clase. En la figura 11 se muestran un ejemplo de las regiones seleccionadas para una sola imagen.

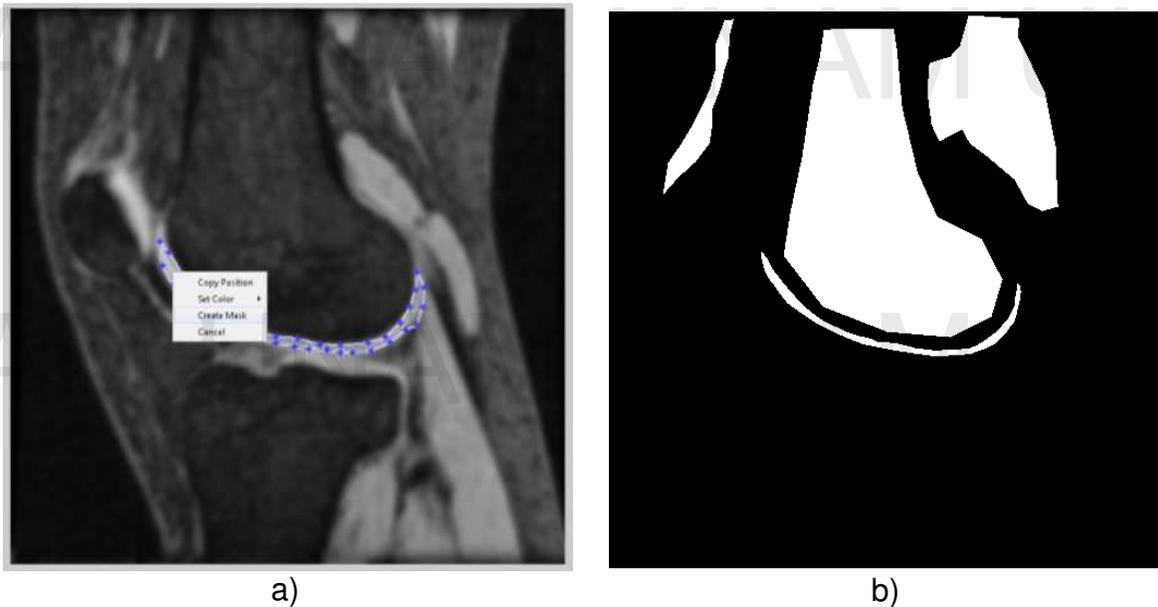


Figura 11. Selección de las regiones de interés como una máscara binaria donde las regiones seleccionadas tienen el valor 1 y el fondo valor 0. a) Selección interactiva de las regiones, b) máscaras obtenidas en una realización del experimento.

En la figura 12 se muestran los histogramas de la intensidad de gris correspondiente a las cuatro regiones seleccionadas de una imagen, en donde se puede apreciar que para cada región se puede identificar la distribución del nivel de gris aproximada a una Gaussiana, sin embargo, si estas regiones se fusionan en una sola imagen, esta mezcla de distribuciones deja de ser evidente, como se muestra en la figura 13.

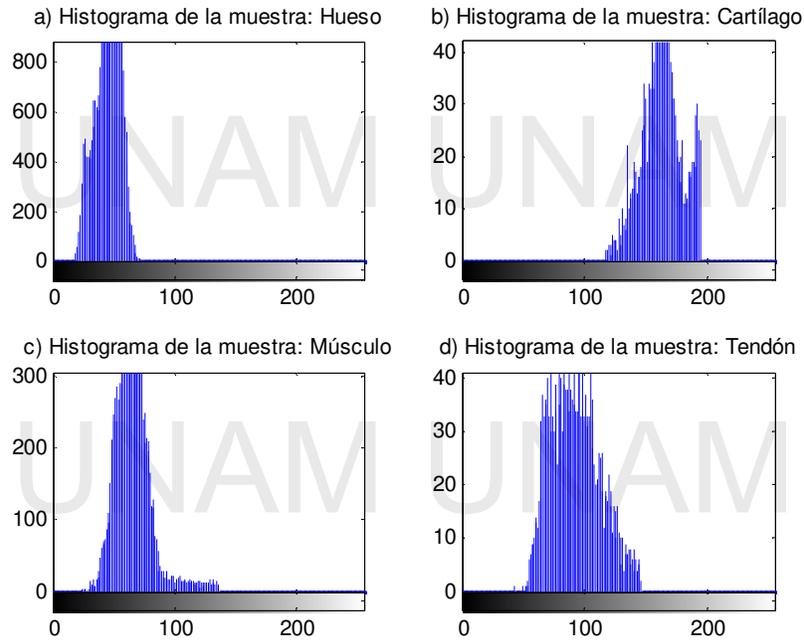


Figura 12. Histograma de las diferentes regiones seleccionadas para una imagen, a) cartílago, b) hueso, c) músculo, d) tendón.

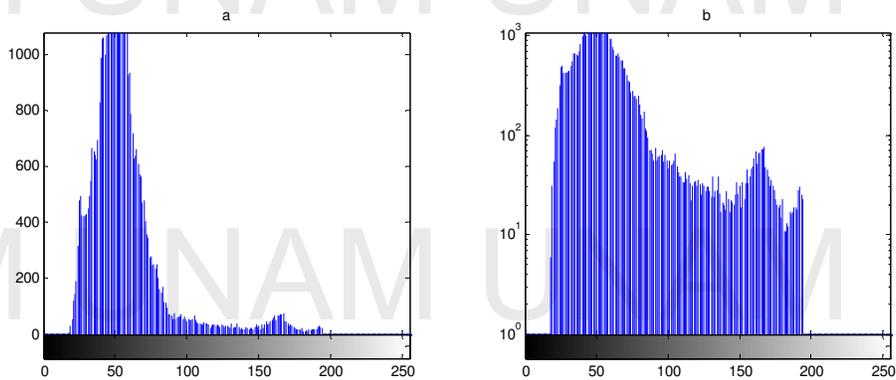


Figura 13. Histograma de un vector de entrenamiento mezclado, conteniendo las muestras de las cuatro regiones. a) Escala vertical lineal, b) escala vertical log.

Además de la intensidad de gris de los píxeles, se toman las coordenadas de estos píxeles, con lo cual se tiene como información adicional la relación espacial y que puede servir como información relevante para discriminar los píxeles de otras regiones con niveles de gris similares.

Las distribuciones conjuntas se pueden graficar con el comando de MatLab `plotmatrix(vector)` el cual grafica por ejemplo, la distribución de las variables

conjuntas nivel de gris y coordenada x , nivel de gris y coordenada y , o coordenada x y coordenada y . Se puede observar que la distribución de las coordenadas x, y genera la forma de la región a la que pertenecen, por ejemplo, en la figura 14. Se muestra las distribuciones conjuntas x, y , nivel de gris.

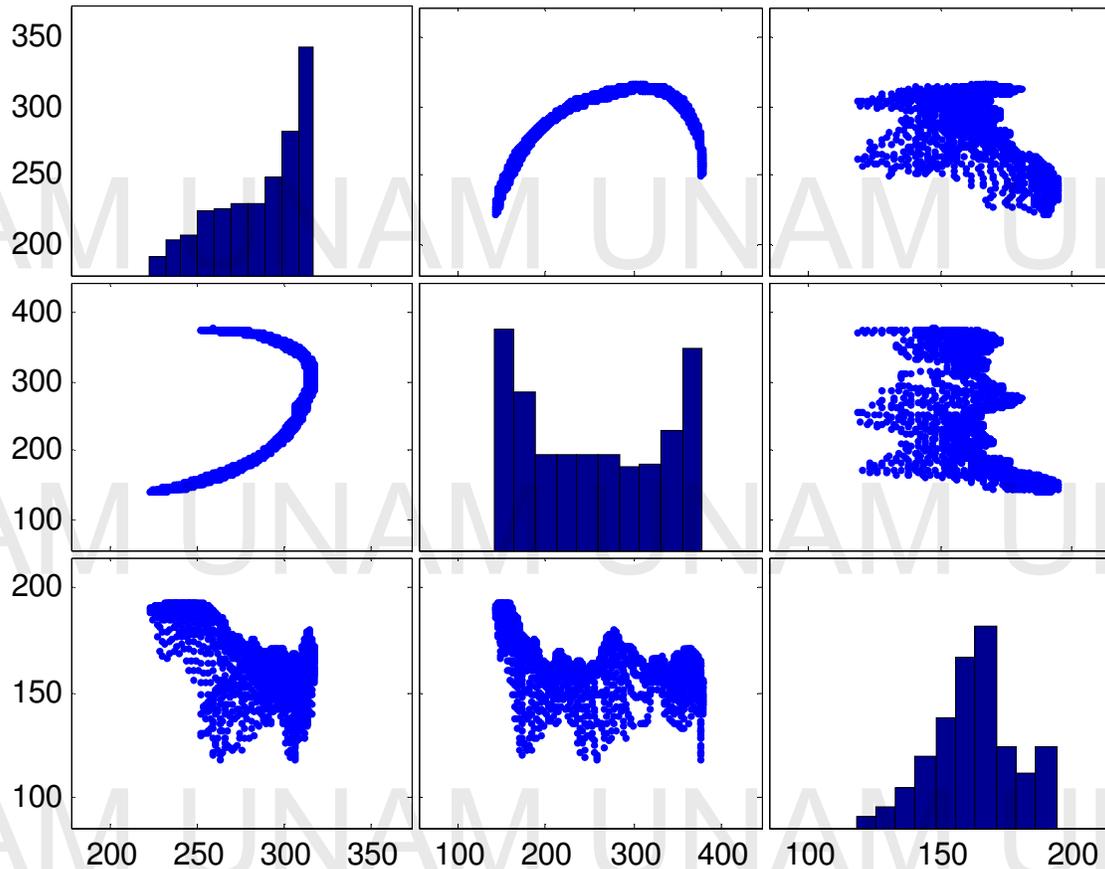


Figura 14. Distribución en pares de las variables para la región de cartílago. El primer elemento de la diagonal corresponde al histograma de la coordenada (y), el segundo a la coordenada (x) y el tercero al nivel de gris.

En un experimento, las muestras de las regiones se tomaron tratando de seguir el contorno del objeto de interés. Una vez formado el vector de atributos para generar los modelos Gaussianos, se antoja observar cual es la distribución de las variables para todas las regiones, para lo cual se obtiene la figura 15, en la cual se puede observar a lo más, tres regiones más o menos separadas si se emplea como discriminante solo el nivel de gris. Se espera que añadiendo información con la relación espacial que guardan los pixeles de una misma región, se logre una mejor clasificación para separar las cuatro regiones.

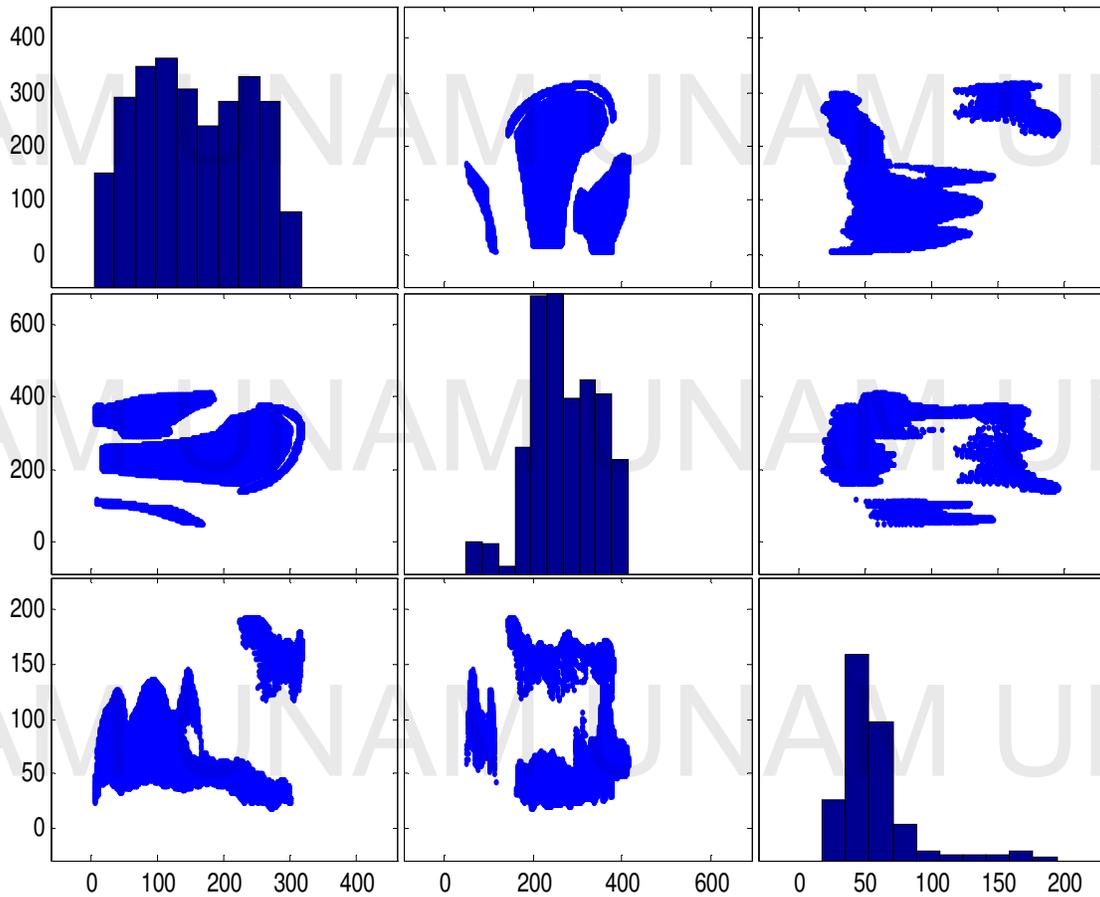


Figura 15. Distribución de pares de las variables de un vector de entrenamiento para una sola imagen.

Una vez que se seleccionaron todas las regiones indicadas, si hay más imágenes por agregar para el entrenamiento, se abrirá de nuevo el cuadro de la figura 10 para seleccionar una nueva imagen de entrenamiento y realizar el mismo proceso de selección de regiones. Una vez que se termina con las imágenes de entrenamiento, el programa genera automáticamente las funciones discriminantes y finalmente solicita la imagen que se empleará para realiza una clasificación. Una vez seleccionada la imagen a segmentar se generará un arreglo llamado 'R' del mismo tamaño que las imágenes con las que se trabajó (512x512), en donde el valor que contiene cada celda corresponde a la clase a la cual pertenece el pixel con esa misma ubicación. Los valores y las clases correspondientes se muestran en la tabla II.

Tabla II. Numero de clase y región correspondiente.

Numero de clase	Región
1	Cartílago
2	Hueso
3	Músculo
4	Tendón

El programa desarrollado emplea una estructura para cada clase, la cual contiene todos los parámetros del modelo Gaussiano como son la media del vector de atributos, matriz de covarianza, su inversa y su determinante, así como la probabilidad a priori obtenida con las muestras seleccionadas. El listado parcial 1 muestra este proceso.

```
% crea clases y campos de cada clase
NoClases = 4; % Numero de regiones o clases
clase(1).nombre = 'Cartílago'; % crea cada clase
clase(2).nombre = 'Hueso';
clase(3).nombre = 'Músculo';
clase(4).nombre = 'Tendón';
for i = 1:NoClases;
    clase(i).VeFeat = []; % vector de atributos de cada clase
    clase(i).VeMedias = []; % vector de medias de cada clase
    clase(i).CovMatrix = zeros(3,3); % Matriz de Covarianza de cada clase
    clase(i).InvCovMat = zeros(3,3); % Inversa de las M de C
    clase(i).DetCovMatrix = []; % Determinantes de las M de Cov
    clase(i).P = []; % vector de probabilidades a priori
end

%% estimacion de parámetros para cada clase o región
TotalPixeles = 0;
for i = 1:NoClases
    clase(i).VeMedias = [mean(clase(i).VeFeat(:,1))
    mean(clase(i).VeFeat(:,2)) mean(clase(i).VeFeat(:,3))]'; %#ok<SAGROW>
    clase(i).CovMatrix = cov(clase(i).VeFeat); %matriz de Covarianza
    clase(i).InvCovMat = inv(clase(i).CovMatrix); %Inversa matriz Cov
    clase(i).DetCovMatrix = det(clase(i).CovMatrix);% Determinante Cov
    [Reng Col] = size(clase(i).VeFeat); % total de pixeles
    TotalPixeles = TotalPixeles + Reng; % de entrenamiento
end
```

Listado 1. Generación de clases y parámetros de los modelos Gaussianos. Los campos `clase(i).VeFeat` se obtiene después de haber seleccionado las regiones de todas las imágenes de entrenamiento.

El proceso para clasificar cada pixel de la imagen para validación emplea un barrido secuencial renglón-columna; de cada pixel se extrae el número de renglón, columna y nivel de gris. Estos tres valores son el vector de características bajo

prueba y en el programa se almacena temporalmente en la variable $vt = [\text{ren col } g]^T$, que es un vector de 3 renglones y 1 columna. Con este vector se evalúa cada una de las funciones de decisión y se clasifica el pixel de acuerdo a la función que genere la probabilidad MAP. Este proceso se muestra en el listado 2.

```

for i = 1:4                                % para cada función de decisión
    M = (vt - clase(i).VeMedias)' ...      % distancia de Mahalanobis
        * clase(i).InvCovMat ...
        * (vt-clase(i).VeMedias);
    fd(i) = -0.5 * M - 0.5 ...             % evalúa funciones
            * log(clase(i).DetCovMatrix)...
            + log(clase(i).P);
end
sel(j, k, 2) = find(fd == max(fd));      % halla clase de MAP e indexa pixel

```

Listado 2. Evaluación de las funciones de decisión y clasificación de pixeles.

4. RESULTADOS.

Al ejecutar el archivo *prog4struc.m* se realizan los procesos anteriormente descritos y se espera mientras se genera la segmentación de la imagen de prueba (figura 16).

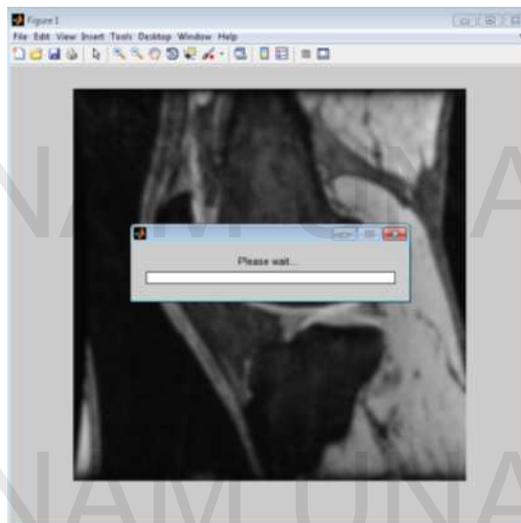


Figura 16. Mensaje de clasificación en proceso.

De los experimentos realizados se obtuvieron los siguientes resultados mostrados en las figuras 17 a 25, donde se muestran cuatro niveles de gris para cada región clasificada y la imagen seleccionada para prueba (imagen filtrada), con el borde en blanco de la región clasificada como cartílago.

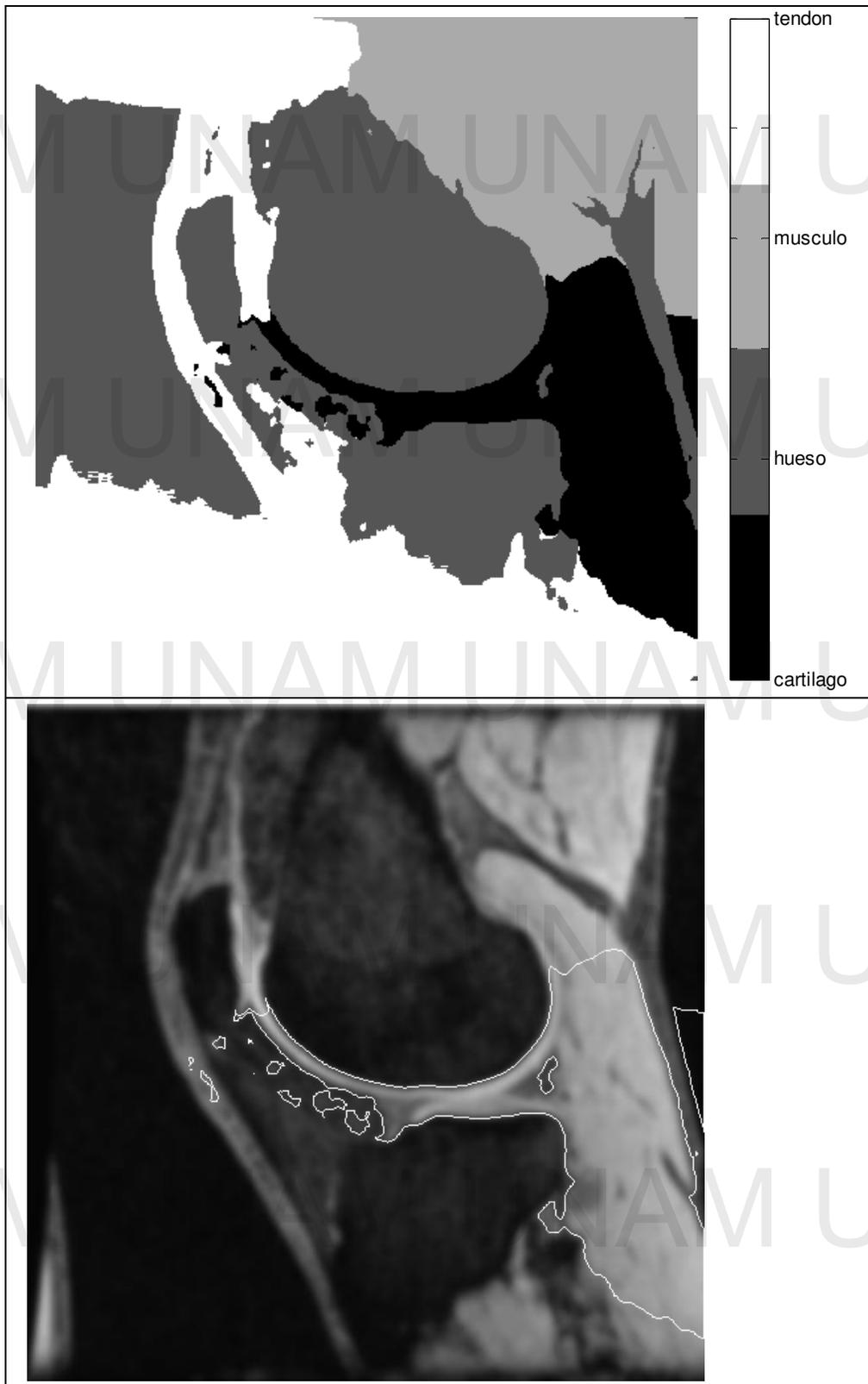


Figura 17. Experimento 1, segmentación imagen 9.



Figura 18. Experimento 2, segmentación imagen 8

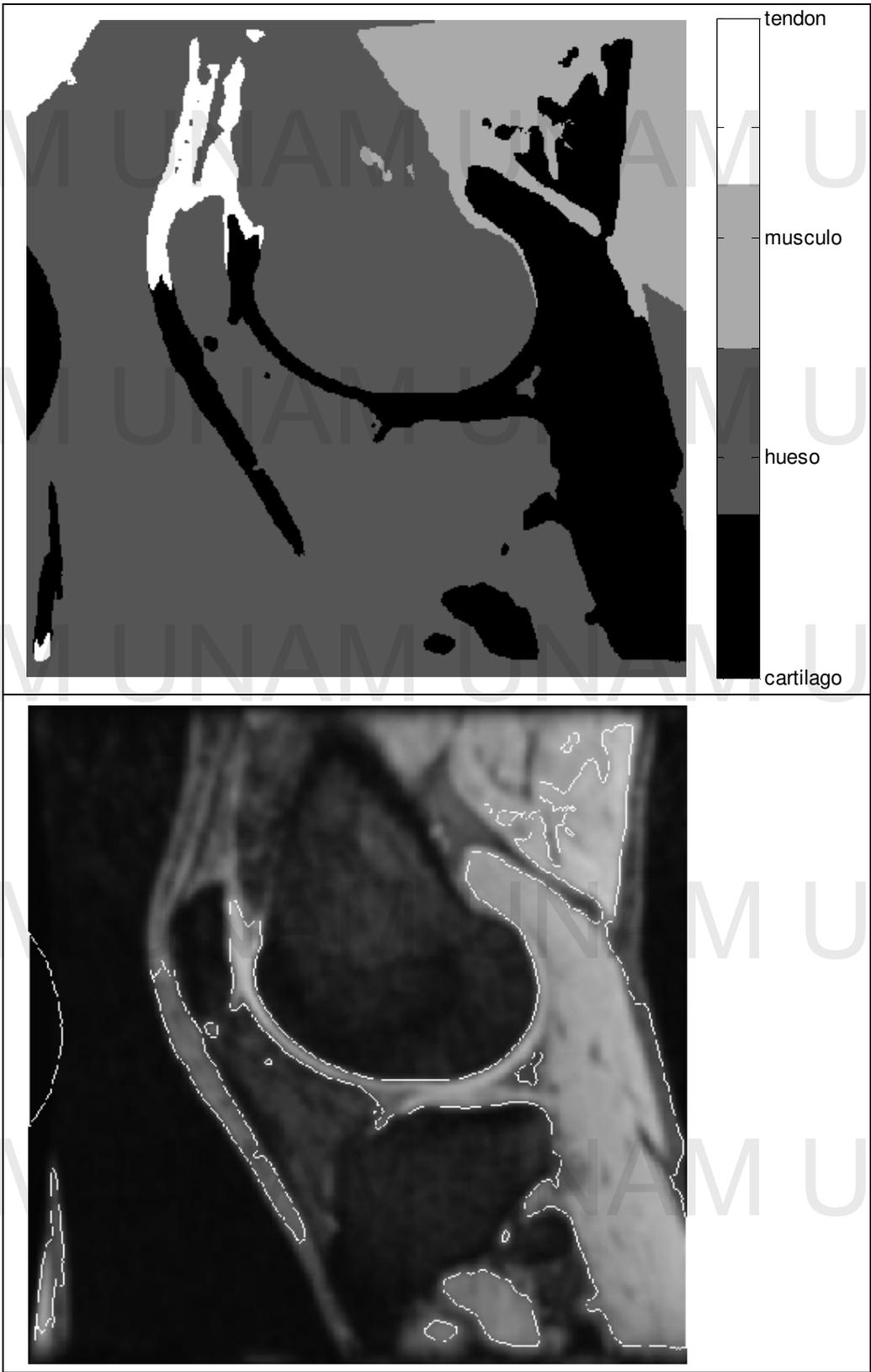


Figura 19. Experimento 3, segmentación imagen 7

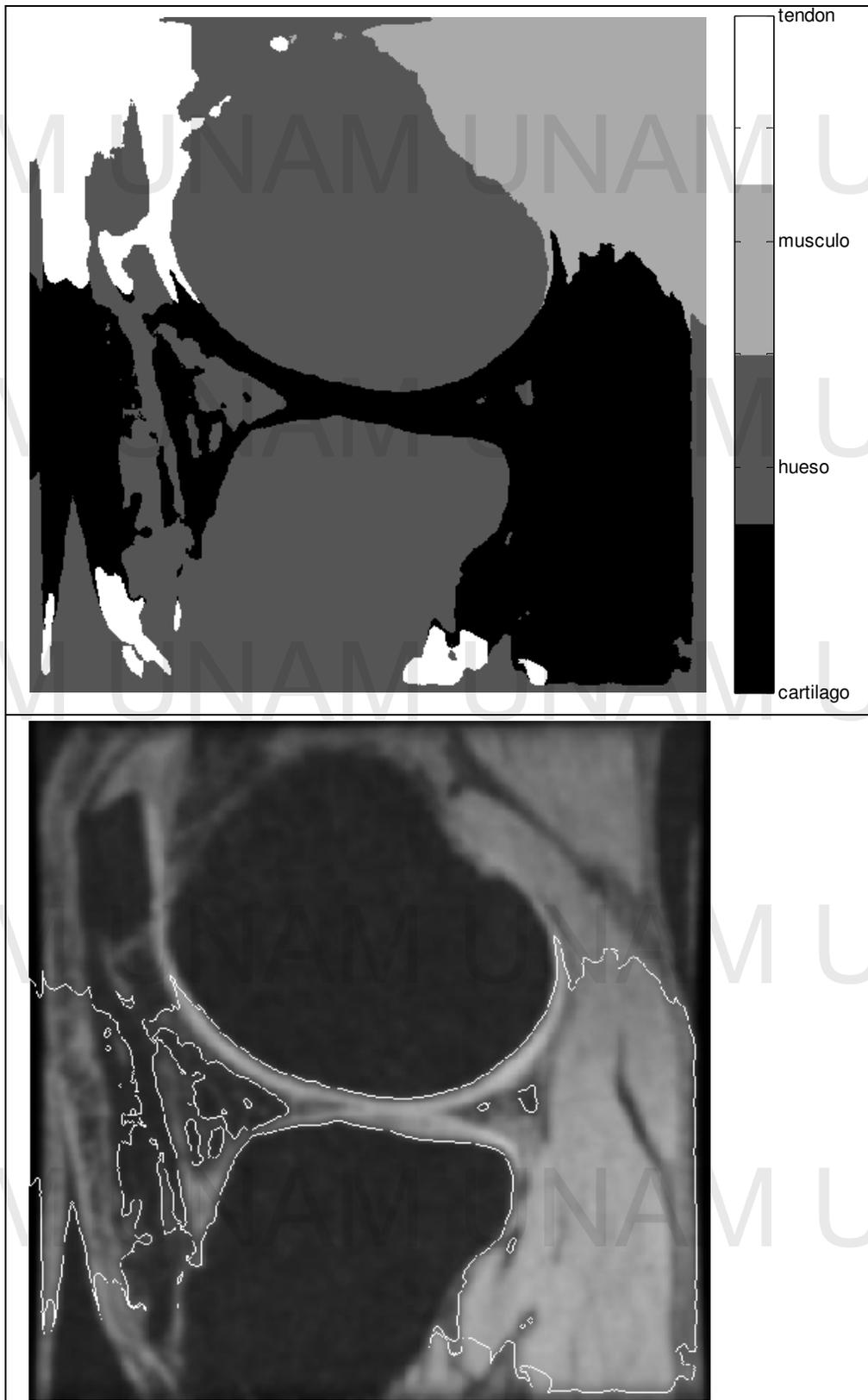


Figura 20. Experimento 4, segmentación imagen 6



Figura 21. Experimento 5, segmentación imagen 5.

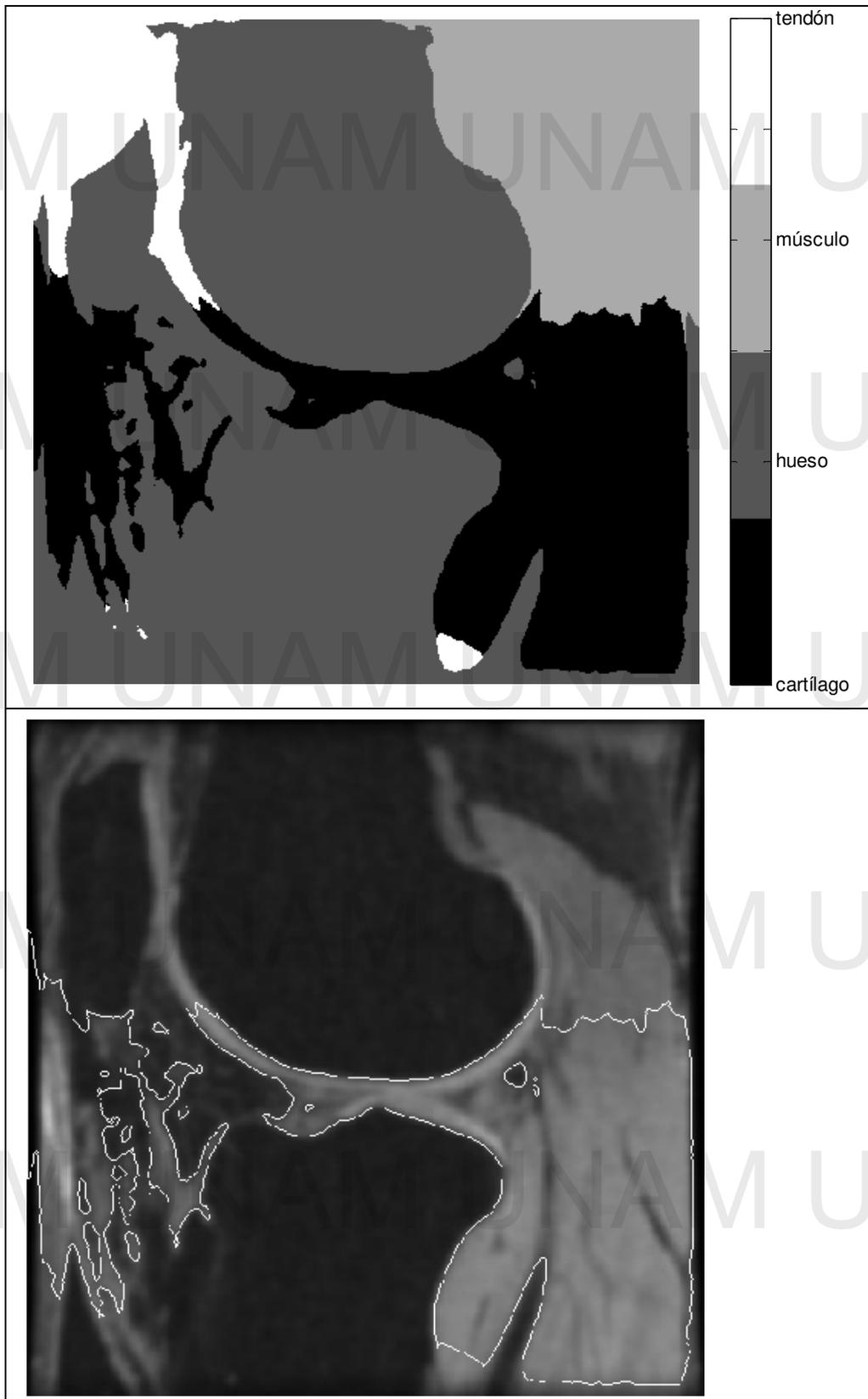


Figura 22. Experimento 6, segmentación imagen 4.



Figura 23. Experimento 7, segmentación imagen 3

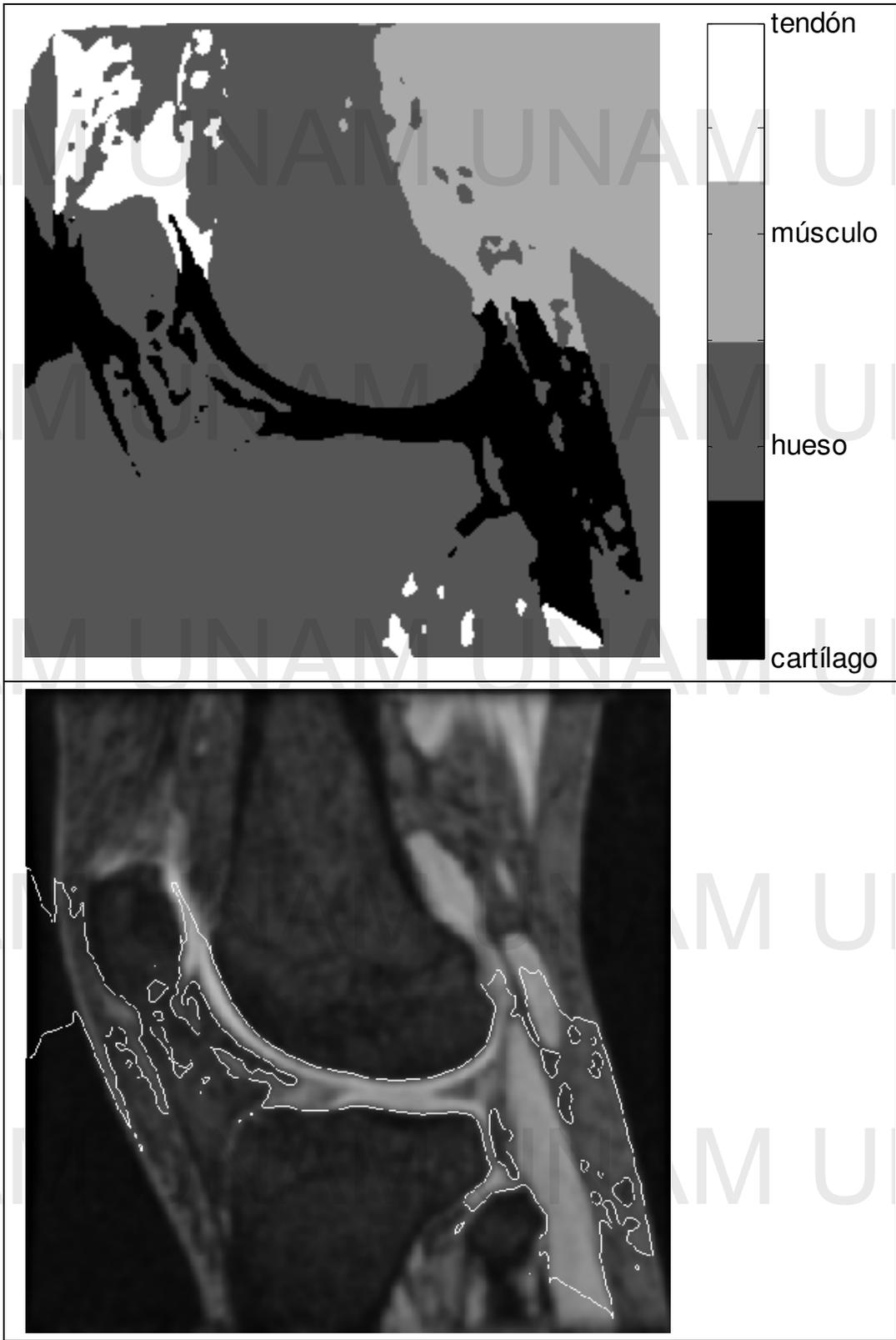


Figura 24. Experimento 8, segmentación imagen 2

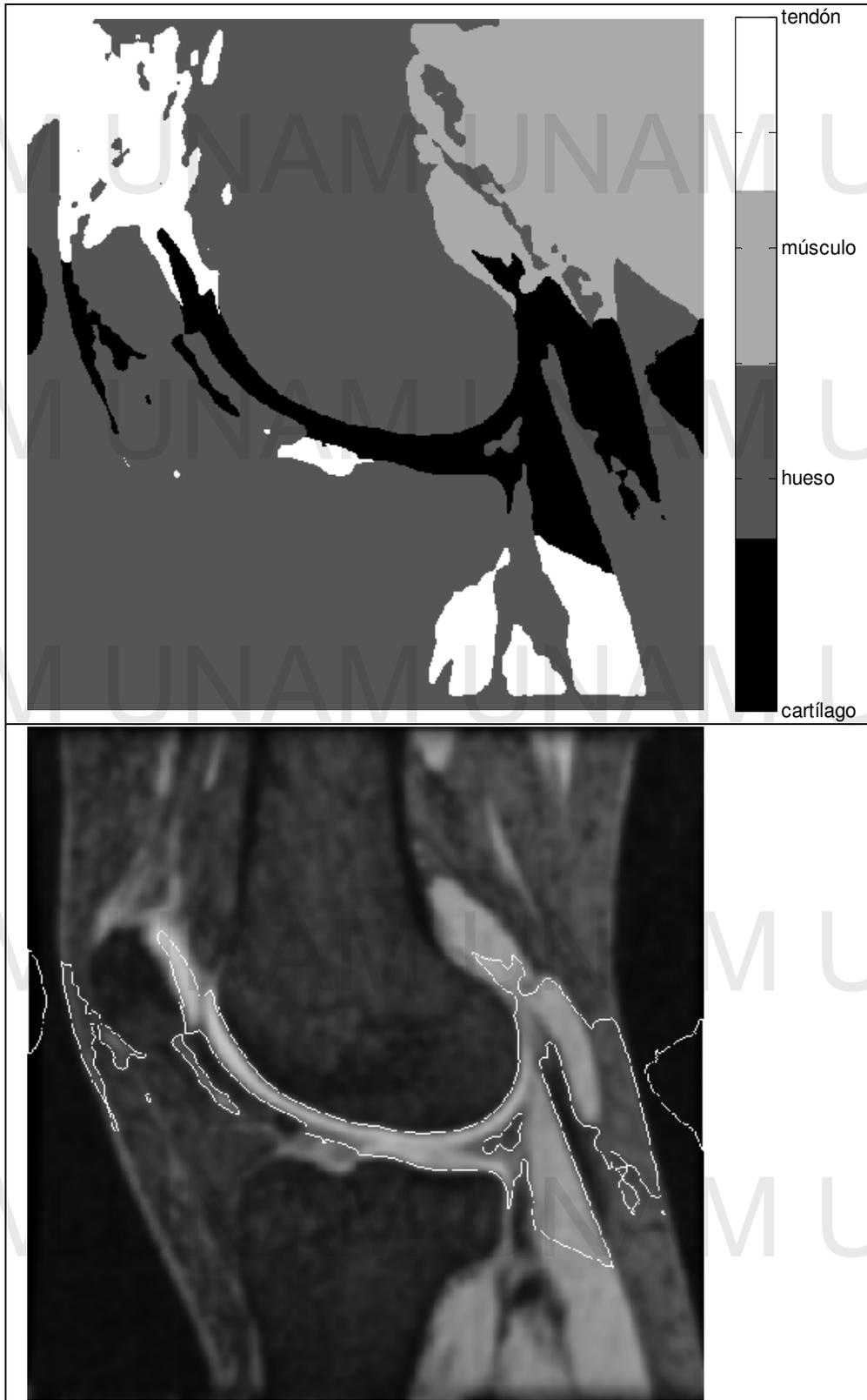


Figura 25. Experimento 9, segmentación imagen 1

Como se puede observar, no se tiene una buena segmentación del cartílago. En la mayoría de los casos, regiones con nivel de gris aparentemente mucho menor, fueron seleccionadas como parte de la región. Una posible causa de este resultado es la relación espacial de la zona de cartílago y la inclusión en las muestras tomadas, de pixeles con nivel de gris menor al de la mayoría de las imágenes, ya que no todas las imágenes tienen un mismo nivel de contraste entre el fondo y los objetos de interés.

Debido a esta diferencia en las imágenes, la dispersión en los niveles de gris aumenta para el modelo Gaussiano del cartílago.

Se considera que la inclusión de las coordenadas x, y de los pixeles de las zonas de interés son un buen discriminante, como lo sería de igual forma relacionar espacialmente cada una de estas zonas con cuadrantes bien definidos, además de ubicar la extremidad en una posición y relación estratégica de tamaño respecto del tamaño de la imagen, con el fin de proporcionar esta relación espacial.

Por ejemplo, en el experimento 4 (figura 20), regiones de tendón y partes del músculo que tienen nivel de gris similar pueden diferenciarse fácilmente ya que se encuentran a los extremos de la imagen. En este sentido, la coordenada x (vertical o columna) de estas regiones son buen discriminante, ya que mantienen poca correlación, a pesar de esto, por ejemplo la zona/clase tendón, puede mantener una relación alta en zonas con nivel de gris similar y una correlación alta o moderada en la coordenada x y poca o ninguna correlación horizontal, provocando que la asociación 'coordenada vertical-nivel de gris' tenga más peso y se asocien pixeles a esta clase, cuando no debería ser así, como se aprecia en la figura 20, donde la zona blanca que incluye el tendón, aparece también en la parte inferior izquierda y derecha.

Los vectores que provienen de diferente clase muestran un grado de traslape en el espacio de atributos, lo cual provoca errores en la clasificación.

5. CONCLUSIONES.

Se emplearon imágenes de una misma dimensión (512x512). No se experimentó con imágenes de otro tamaño.

La posición de la articulación relativa a la imagen no fue homogénea, ya que en algunas imágenes la rodilla ocupaba casi la totalidad del espacio disponible de la imagen, eliminando parte del fondo, mientras en otras, se ocupaba menor espacio permitiendo apreciar una zona de fondo. Además, la diferencia de contraste en las imágenes pudo mezclar aún más las regiones provocando un pobre desempeño del clasificador.

En este sentido, se puede mejorar el desempeño ya sea agregando más imágenes de entrenamiento o realizando ajustes a la toma de la imagen, como ubicación de la extremidad y contrastes homogéneos.

En los experimentos se observó que el desempeño para segmentar el cartílago dependía del tamaño de la muestra así como de los niveles de gris incluidos, procurando que este fuera lo más homogéneo posible, con lo cual se obtenía al menos, una segmentación de regiones con niveles de gris similar y cercanas al cartílago, descartando otras con similar nivel de gris ubicadas más lejos, como en el experimento 2 y 9 (figuras 18 y 25).

Finalmente, en este tipo de imágenes donde se captura gran cantidad de ruido y donde existen regiones anatómicas diferentes con niveles de gris similar y correlación espacial, o traslape en el espacio de atributos, un enfoque bayesiano no tiene buen desempeño según las pruebas realizadas.

6. BIBLIOGRAFÍA.

1. Solomon C, Breckon T. *Fundamentals of Digital Image Processing. A Practical Approach with Examples in Matlab*. John Wiley & Sons, Ltd, 2011.
2. Bankman IN (Editor). *Handbook of Medical Image Processing and Analysis*, 2nd Edition, Academic Press, USA, 2008.
3. Petrou M, Bosdogianni P. *Image Processing, the Fundamentals*. John Wiley & Sons, Ltd, 1999.
4. Barilla Perez ME, Franco Bello JA. *Fusión de datos y segmentación de imágenes de percepción remota*. Tesis, UNAM, Facultad de Ingeniería 2001.
5. Burger W, Burge MJ. *Principles of digital image processing Processing. Fundamental techniques*. Springer-Verlag London Ltd 2009.
6. Gonzalez RC., Woods RE, Eddins SL. *Digital Image Processing using Matlab*, 2nd Ed.
7. Omer Demirkaya et al. *Image Processing with MATLAB. Applications in Medicine and Biology*. CRC Press Taylor & Francis Group 2009.