



**BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA**

**FACULTAD DE CIENCIAS DE LA COMPUTACIÓN**

**“SISTEMA BASE PARA LA CLASIFICACIÓN DE  
COBERTURAS DE LA TIERRA EMPLEANDO  
IMÁGENES SATELITALES”**

**TESIS  
PROFESIONAL**

**QUE PARA OBTENER EL TÍTULO DE LICENCIADO EN**

**INGENIERÍA EN CIENCIAS DE LA COMPUTACIÓN**

**PRESENTA:**

**VÁZQUEZ ALDAMA OMAR OSCAR**

**ASESOR:**

**M.C. ERIKA DANAÉ LÓPEZ ESPINOZA**

**COASOR:**

**DR. MANUEL MARTÍN ORTIZ**

**H. PUEBLA DE ZARAGOZA, ABRIL DE 2008**



# Resumen

En este trabajo de tesis se presenta el diseño y la implementación de un sistema base para la clasificación de coberturas terrestres utilizando imágenes satelitales proporcionadas por el satélite SPOT 5.

Para realizar la clasificación de dichas coberturas se utiliza el reconocimiento de patrones mediante un enfoque estadístico. Los clasificadores implementados fueron Distancia Mínima a la Media, Distancia Mahalanobis y Máxima Probabilidad, e ISO-DATA. Las imágenes temáticas generadas por los clasificadores son evaluados cuantitativamente para obtener el porcentaje de clasificación.

El sistema también proporciona información de referencia geográfica en las imágenes temáticas generadas, así como información que permite evaluar los resultados de la clasificación mediante la estimación de exactitud de los clasificadores o mediante la información de referencia ya establecida. Además, permite la comparación de manera visual.

El desarrollo de este sistema pretende ser la base para la implementación de futuros clasificadores, los cuales puedan obtener mejores resultados con respecto a los clasificadores estadísticos implementados en esta tesis.

# Índice general

<b>Índice de Figuras</b>	<b>IV</b>
<b>Índice de Tablas</b>	<b>VI</b>
<b>1. Introducción</b>	<b>1</b>
<b>2. Percepción Remota</b>	<b>4</b>
2.1. Introducción . . . . .	4
2.2. Energía y Espectro Electromagnético . . . . .	6
2.3. Imágenes Satelitales . . . . .	10
2.3.1. Características Generales . . . . .	10
2.3.2. Satélites SPOT . . . . .	11
2.4. Reconocimiento de Patrones . . . . .	14
2.4.1. Introducción . . . . .	14
2.4.2. Clasificación Supervisada . . . . .	16
2.4.3. Clasificación No Supervisada . . . . .	18
2.5. Métodos de Evaluación . . . . .	21
2.5.1. Matriz de Confusión . . . . .	21
<b>3. Análisis y Diseño del Sistema</b>	<b>25</b>
3.1. Ingeniería de Software . . . . .	25
3.1.1. Descripción del Problema . . . . .	25
3.1.2. Casos de Uso . . . . .	26
3.1.3. Especificación de Casos de Uso . . . . .	28
3.1.4. Escenarios . . . . .	29
3.1.5. Diagramas de Secuencia . . . . .	37

3.1.6. Diagramas de Colaboración . . . . .	49
<b>4. Desarrollo del Sistema</b>	<b>62</b>
<b>5. Resultados Experimentales</b>	<b>72</b>
5.1. Origen de los datos experimentales . . . . .	72
5.2. Resultados experimentales obtenidos . . . . .	75
<b>6. Conclusiones y Trabajo a Futuro</b>	<b>87</b>
<b>A. Especificación de casos de uso</b>	<b>89</b>
<b>B. Descripción de las clases</b>	<b>101</b>
<b>Referencias</b>	<b>111</b>

# Índice de Figuras

2.1. Proceso de Percepción Remota (imagen tomada de [1]). . . . .	6
2.2. Esquema de una onda electromagnética (imagen tomada de [2]). . . . .	7
2.3. Regiones del espectro electromagnético. . . . .	9
2.4. Proceso de clasificación supervisada en percepción remota. . . . .	16
2.5. Ejemplo de validación cruzada con $k = 4$ . . . . .	23
2.6. Ejemplo de matriz de confusión . . . . .	24
3.1. Diagrama de Casos de Uso para el sistema de Clasificación. . . . .	28
3.2. Obtención de Datos de entrenamiento nuevos de forma manual. . . . .	38
3.3. Error validacion de Clases y Muestras. . . . .	39
3.4. Error en lectura de imágenes de entrada. . . . .	40
3.5. Obtención de datos de entrenamiento nuevos en forma automatica. . . . .	41
3.6. Obtención de datos de entrenamiento almacenados. . . . .	42
3.7. Clasificador de Distancia Mínima a la Media. . . . .	43
3.8. Clasificador de Distancia Mahalanobis. . . . .	44
3.9. Clasificador de Máxima Probabilidad. . . . .	45
3.10. Clasificador ISODATA. . . . .	46
3.11. Error al ingresar datos ISODATA. . . . .	47
3.12. Matriz de confusión. . . . .	48
3.13. Error en obtención de datos de la matriz de confusión . . . . .	49
3.14. Obtención de datos de entrenamiento nuevos en forma manual. . . . .	50
3.15. Error validacion de Clases y Muestras. . . . .	51
3.16. Error en lectura de imagenes de entrada. . . . .	52
3.17. Obtención de datos de entrenamiento nuevos en forma automatica. . . . .	53
3.18. Obtención de datos de entrenamiento almacenados. . . . .	54

3.19. Clasificador de Distancia Mínima a la Media. . . . .	55
3.20. Clasificador de Distancia Mahalanobis. . . . .	56
3.21. Clasificador de Maxima Probabilidad. . . . .	57
3.22. Clasificador ISODATA. . . . .	58
3.23. Error al ingresar datos ISODATA. . . . .	59
3.24. Matriz de confusión. . . . .	60
3.25. Error en obtención de datos de la matriz de confusión. . . . .	61
4.1. Diagrama de Clases del sistema . . . . .	63
4.2. Pantalla de inicio del sistema. . . . .	64
4.3. Mensaje de error cuando un valor no se encuentra entre el rango [1,10].	65
4.4. Explorador de archivos. . . . .	65
4.5. Mensaje de error al ingresar las rutas de los archivos. . . . .	66
4.6. Inicio del sistema. . . . .	66
4.7. Mensaje que indica al usuario ingresar el nombre de la clase. . . . .	67
4.8. Mensaje que indica al usuario ingresar los datos de entrenamiento. . . .	67
4.9. Mensaje que indica al usuario que se va a iniciar la clasificación. . . . .	68
4.10. Se muestra imagen temática obtenida. . . . .	68
4.11. Se muestra la matriz de confusión. . . . .	69
4.12. Ejecución de todos los clasificadores supervisados con los mismos datos de entrenamiento. . . . .	70
4.13. Solicitud de datos en el clasificador ISODATA. . . . .	71
4.14. Clasificador ISODATA. . . . .	71
5.1. Imagen Satelital (corte de tamaño 2194 x 1931). . . . .	73
5.2. Imagen Multiespectral Sintética. . . . .	74
5.3. Esquema de Clasificación . . . . .	75
5.4. Mapas temáticos generados por cada clasificador para la imagen de 2194 x 1931. . . . .	76
5.5. Mapas temáticos generados por cada clasificador para imagen satelital 150 x 150. . . . .	79
5.6. Mapas temáticos generados por cada clasificador para IMS 150 x 150. . .	82
5.7. Validación de resultados entre clasificadores e imagen de referencia. . .	84

# Índice de Tablas

2.1. Características de los satélites SPOT. . . . .	12
2.2. Aplicaciones del sensor HRV. . . . .	13
5.1. Resultados del clasificador DMM para imagen de 2194 x 1931. . . . .	77
5.2. Resultados del clasificador DMH para imagen de 2194 x 1931. . . . .	77
5.3. Resultados del clasificador MP para imagen de 2194 x 1931. . . . .	78
5.4. Comparativo entre clasificadores de imagen satelital de 2194 x 1931. . .	78
5.5. Resultados del clasificador DMM para imagen satelital 150 x 150. . . .	79
5.6. Resultados del clasificador DMH para imagen satelital 150 x 150. . . .	80
5.7. Resultados del clasificador MP para imagen satelital 150 x 150. . . . .	80
5.8. Comparativo entre clasificadores de imagen satelital 150 x 150. . . . .	81
5.9. Resultados del Clasificador DMM para IMS 150 x 150. . . . .	82
5.10. Resultados del Clasificador DMH para IMS 150 x 150. . . . .	83
5.11. Resultados del Clasificador MP para IMS 150 x 150 . . . . .	83
5.12. Cuadro comparativo entre clasificadores de IMS de 150x 150 utilizando validación cruzada. . . . .	84
5.13. Resultados del Clasificador DMM evaluada con imagen de referencia. . .	85
5.14. Resultados del Clasificador DMH evaluada con imagen de referencia. . .	85
5.15. Resultados del Clasificador MP evaluada con imagen de referencia. . . .	86
5.16. Comparativo entre clasificadores de IMS de 150x 150 utilizando com- paración con imagen de referencia. . . . .	86

# Capítulo 1

## Introducción

Las imágenes satelitales son la principal fuente de información espacial, siendo tema de investigación de una amplia gama de aplicaciones. En particular, ha recibido una considerable atención durante los últimos diez años la clasificación de coberturas terrestres a partir de datos remotamente sensados, debido a que la información obtenida sirve como entrada para actualizar Sistemas de Información Geográfica (SIG o en inglés GIS).

Con el desarrollo de la ciencia y la tecnología, el procesamiento de imágenes satelitales se ha convertido en un área de investigación muy importante para el desarrollo de programas sustentables que recuperen los recursos naturales. La clasificación de coberturas presentes en una escena, tiene como objetivo, obtener una nueva imagen en la cual cada uno de los píxeles se encuentre etiquetado en una clase de interés. La clasificación de imágenes está relacionada con la obtención de un mapa temático en el que se categorizan los tipos de coberturas.

Existen muchas aplicaciones que se pueden desarrollar a partir de los mapas temáticos obtenidos, entre ellas encontramos la cartografía, evaluación de impacto ambiental de recursos naturales, medición de superficies, etc. El objetivo de analizar y procesar imágenes satelitales es obtener información valiosa y novedosa para los especialistas del área.

La clasificación de imágenes realiza a cabo a partir de un clasificador y una medida cuantitativa que lo ayuda a encontrar la semejanza entre los píxeles de una imagen. Entre los principales enfoques empleados en la clasificación están: el supervisado y el no supervisado.

La clasificación no supervisada es una técnica de agrupamiento, la cual se enfoca a encontrar clases espectrales que están presentes en la imagen. No implica ningún conocimiento en el área de estudio, por lo que la interpretación humana se centra más en la interpretación de los resultados. Por otro lado, la clasificación supervisada parte de un cierto conocimiento de la zona de estudio. Esta familiaridad con la zona de estudio, permite al interpretador delimitar sobre la imagen áreas de entrenamiento que se consideran lo suficientemente representativas de las coberturas a clasificar. A partir de estas muestras de entrenamiento, el algoritmo determina la asignación de cada píxel a la clase que más se parece.

En la actualidad existe una gran variedad de software comercial relacionado con el análisis y procesamiento de imágenes satelitales. Estas herramientas tienen entre sus múltiples aplicaciones algoritmos para clasificación de coberturas, sin embargo, sólo proporcionan métodos de clasificación estadística, los cuales resultan algunas veces limitados para resolver algunos problemas reales. La mayoría de estos productos son demasiado caros, costando alrededor de 20,000 euros.

Se han empleado y propuesto diversos algoritmos para clasificación de imágenes satelitales. Entre ellos encontramos algoritmos genéticos [3], de redes neuronales [4], lógicos combinatorios [5], modelos probabilistas [6][7] y algoritmos estadísticos [8]. Todos estos trabajos reportan mejores desempeños de clasificación en comparación con los algoritmos que tienen las herramientas comerciales.

Por tal motivo, el objetivo principal de este trabajo es establecer las bases para que en un futuro se desarrolle una herramienta que conjunte los diversos enfoques de clasificación, considerando como punto de partida los clasificadores estadísticos que forman parte del software comercial ERDAS, ya que estos clasificadores son los comúnmente utilizados en la literatura y en las herramientas comerciales para la clasificación de coberturas. Para el presente trabajo se eligieron los clasificadores supervisados de Distancia Mínima a la Media, el de Distancia Mahalanobis y el de Máxima Probabilidad y el clasificador no supervisado ISODATA. Por lo cual, en este trabajo de tesis se plantea como objetivo general:

- *Desarrollar un sistema base de percepción remota que conjunte algoritmos estadísticos de clasificación, así como métodos de evaluación de coberturas terrestres sin perder la referencia geográfica de las imágenes satelitales.*

Los objetivos específicos a cumplir son los siguientes:

- *Realizar el análisis y diseño del sistema con base a los requerimientos definidos (Ingeniería de Software) .*
- *Analizar y programar algoritmos de clasificación supervisada más empleados para la generación de mapas temáticos.*
- *Analizar y programar algoritmos de clasificación no supervisada más empleados para la generación de mapas temáticos.*
- *Analizar y programar métodos de evaluación de clasificadores supervisados.*
- *Programar la función que mantenga durante todo el proceso de clasificación la referencia geográfica de las imágenes satelitales.*

La organización de la tesis esta conformada de la siguiente manera:

Capítulo 2 aborda los fundamentos teóricos que nos permiten el conocimiento del panorama general del tema de estudio. Esta compuesto por temas relacionados como percepción remota, imágenes satelitales, reconocimiento de patrones y métodos de evaluación.

Capítulo 3 aborda todo el proceso de análisis y diseño del sistema, desarrollando toda la ingeniería de software necesaria para la implementación del sistema de una manera eficiente. También se muestran los diagramas de casos de uso y su descripción, los diferentes escenarios, diagramas de secuencia y de colaboración.

Capítulo 4 explica el desarrollo y la implementación del sistema, el lenguaje de programación utilizado y la descripción de cada una de las clases y funciones implementadas.

En el capítulo 5 se discuten y muestran los resultados obtenidos de las pruebas realizadas con los diferentes clasificadores implementados en este trabajo de tesis.

Capítulo 6 se muestran las conclusiones y trabajo futuro.

Por ultimo se muestran dos anexos en donde se explica de manera detallada la especificación de los casos de uso para el desarrollo del sistema y la descripción de las clases implementadas.

# Capítulo 2

## Percepción Remota

### 2.1. Introducción

La palabra **Percepción Remota** ó **Teledetección** es la traducción al español de la expresión inglesa **Remote Sensing**, ideada a principios de los años 60's para designar cualquier método de observación remota de la superficie terrestre, aunque se aplicó principalmente a la fotografía aérea.

La historia de la Percepción Remota comenzó hace unos 600 millones de años, cuando alguna forma inferior de vida animal diferenció algunas de sus células, volviéndolas fotosensibles. Durante millones de años dicho elemento fotosensible evolucionó convirtiéndose en un poderoso y sofisticado sensor, el ojo humano. Este tuvo un imitador mecánico, la cámara fotográfica, que hizo su aparición hace algo más de un siglo y que fue mejorada durante la década de los 30's para ser aplicada a la fotografía aérea. La Segunda Guerra Mundial dio un gran impulso a la fotografía aérea así como a otras formas de Percepción Remota. Sin embargo, el "salto cuántico", en esta disciplina se produjo en la década de los 60's cuando las plataformas satelitales reemplazaron a las aéreas y los sensores electrónicos multiespectrales, acoplados a computadoras, reemplazaron las cámaras fotográficas.

El desarrollo de la tecnología espacial permitió contar con plataformas de observación satelital, dando lugar a una variante de la Percepción Remota: la Percepción Remota Satelital. El concepto de Percepción Remota involucra no sólo los procesos y aparatos que permiten obtener una imagen remota de la superficie terrestre, sino también su posterior procesamiento para la obtención de información relevante.

Podemos definir la Percepción Remota como:

*“La ciencia de obtener información acerca de la superficie de la tierra sin entrar en contacto con ella. Esto se realiza aplicando un conjunto de técnicas, aparatos y procedimientos que permiten obtener y analizar las imágenes de la superficie de la tierra desde sensores ubicados remotamente”[1][9].*

A partir de la definición podemos encontrar que existe un objeto, área o fenómeno de interés a investigar. Generalmente los científicos están interesados en estudiar los fenómenos naturales y los recursos de la superficie terrestre, como son los cultivos. También, se tiene un dispositivo que está a una distancia del objeto de interés y que permite obtener datos del mismo a través de sensores que capturan la interacción entre el objeto y la radiación electromagnética. De esta forma se captura la relación espacial y espectral de los objetos y materiales observados a distancia. Además, se realiza un análisis de los datos para obtener información, la cual involucra tareas de identificación y categorización de dichos datos.

Como se dijo anteriormente el proceso de Percepción Remota considera la interacción entre la radiación electromagnética y los objetos de interés, involucrando los siguientes elementos [1](Fig. 2.1):

- A. **Fuente de energía o iluminación:** El primer requerimiento en Percepción Remota es disponer de una fuente de energía que ilumine o provea energía electromagnética al objeto de interés.
- B. **Radiación y la atmósfera:** Ya que la energía viaja desde la fuente al objeto, entrará en contacto e interactuará con la atmósfera. Esta interacción tiene lugar una segunda vez cuando la energía viaja desde el objeto al sensor.
- C. **Interacción con el objeto:** La energía interactúa con el objeto dependiendo de las propiedades de éste y de la radiación incidente.
- D. **Detección de energía por el sensor:** Necesitamos un sensor remoto que capture y grabe la radiación electromagnética reflejada o emitida por el objeto y la atmósfera.
- E. **Transmisión, recepción y procesamiento.** La energía capturada por el sensor debe ser transmitida, normalmente en forma electrónica, a una estación de recepción y procesamiento donde los datos son convertidos a imágenes digitales.

- F. **Análisis e interpretación:** La imagen procesada se interpreta, visualmente y/o digitalmente, para extraer información acerca del objeto que fue capturado por el sensor.
- G. **Aplicación:** El paso final en el proceso de Percepción Remota se alcanza cuando aplicamos la información extraída de las imágenes del objeto para un mejor conocimiento del mismo, revelando nueva información o ayudándonos a resolver un problema en particular.

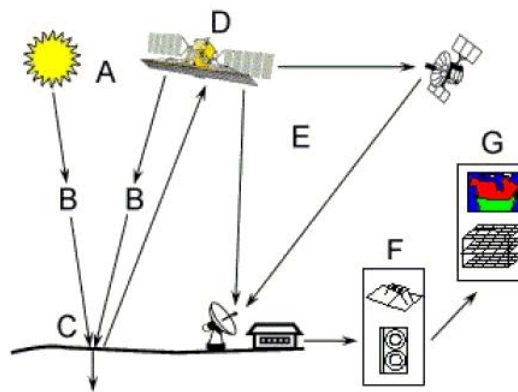


Figura 2.1: Proceso de Percepción Remota (imagen tomada de [1]).

## 2.2. Energía y Espectro Electromagnético

**Radiación:** Es la energía emitida, transmitida y propagada en forma de ondas a través de un medio (Fig 2.2).

**Radiación electromagnética:** Es la energía propagada a través del espacio o de medios naturales en forma de una interacción que avanza entre campos eléctricos y magnéticos. A diferencia de otros tipos de onda, como el sonido, que necesitan un medio natural para propagarse, la radiación electromagnética se puede propagar en el vacío.

Las propiedades de la radiación electromagnética están explicadas por dos teorías contrapuestas: la primera es aquella que la concibe como un haz ondulatorio (Huy-

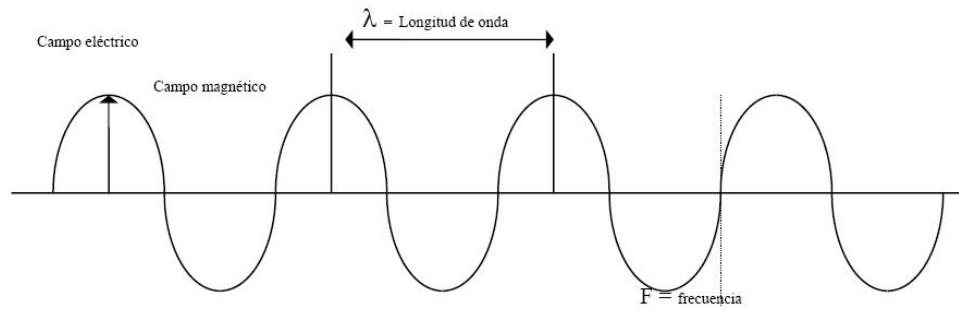


Figura 2.2: Esquema de una onda electromagnética (imagen tomada de [2]).

Maxwell), donde la energía electromagnética se transmite de un lugar a otro siguiendo un modelo armónico y continuo, a la velocidad de la luz y conteniendo dos campos de fuerza ortogonales entre sí (eléctrico y magnético). Las características de este flujo energético pueden describirse por dos elementos:

$\lambda$  = Longitud de onda (distancia entre dos picos sucesivos de una onda).

$F$  = Frecuencia (número de ciclos pasando por un punto fijo en una unidad de tiempo).

$c$  = velocidad de la luz

$$c = F(\lambda) \quad (2.1)$$

lo que significa a mayor longitud de onda, menor frecuencia y viceversa.

La segunda teoría explica la radiación electromagnética considerándola como una sucesión de unidades discretas de energía, fotones o cuantos, con masa igual a cero (Planck, Einstein). Esta nos permite calcular la cantidad de energía calculada por un fotón, siempre que se conozca su frecuencia:

$Q$  = Energía radiante de un fotón (Julios)

$F$  = Frecuencia

$h$  = Constante de Planck ( $6.6 \times 10^{-34}$  julios/seg.)

$$Q = h(F) \quad (2.2)$$

sustituyendo frecuencia:

$$F = c/\lambda \quad (2.3)$$

quedaría:

$$Q = h(c/\lambda) \quad (2.4)$$

lo que significa, que a mayor longitud de onda o menor frecuencia el contenido energético será menor y viceversa.

**Espectro electromagnético:** Se denomina espectro electromagnético a la radiación electromagnética que emite (espectro de emisión) o absorbe (espectro de absorción) una sustancia. El espectro electromagnético incluye longitudes de onda que se extienden desde angstroms hasta kilómetros y las frecuencias fluctúan entre  $10^4$  y  $10^{20}$  correspondiendo las frecuencias mas bajas a las longitudes de onda mayores y las frecuencias altas a las longitudes de onda menores.

Las regiones identificadas en este espectro electromagnético son las siguientes (Fig 2.3):

- **Rayos Gamma**  $< 0.03$  nm: La radiación que llega es completamente absorbida por la atmósfera y no es aprovechada por los sensores remotos.
- **Rayos -X** 0.03 a 3.0 nm: Son rayos producidos por los electrones más fuertes y son completamente absorbidos por la atmosfera. No se emplean en sensores remotos.
- **Ultravioleta** 0.03 a 0.4  $\mu\text{m}$ : Los rayos que llegan estan en longitudes de onda muy bajas (0.3 $\mu\text{m}$ ), por lo cual son completamente absorbidos por el ozono de la atmósfera.
- **Luz visible** 0.4 a 0.7  $\mu\text{m}$ : Es detectable con película y fotodetectores, sensible al ojo humano, coincide con la máxima radiación de la tierra (0.5 $\mu\text{m}$ ).

- **Infrarroja** 0.7 a 100  $\mu\text{m}$ : La interacción con la materia varía según la longitud de onda. Se encuentran 3 subdivisiones: IR-cercano, IR-lejano ó térmico y la radiación infrarroja.
- **Microondas**: Estas longitudes de onda pueden penetrar nubes y niebla. Las imágenes se adquieren en forma pasiva o activa. El rango de microondas se extiende desde los 1000 $\mu\text{m}$  hasta los 0.3 m. En él se pueden localizar las ondas de radar y otros sistemas de comunicación como el UHF (del inglés: Ultra High Frequency).
- **Radio**: Corresponde a ondas de televisión, radio (cortas y largas) y redes eléctricas de corriente alterna, su rango va desde los 0.3 m hasta algunos km. Estás son las ondas que al reflejarse en la ionosfera permiten establecer comunicación a grandes distancias. No tiene aplicación en percepción remota.

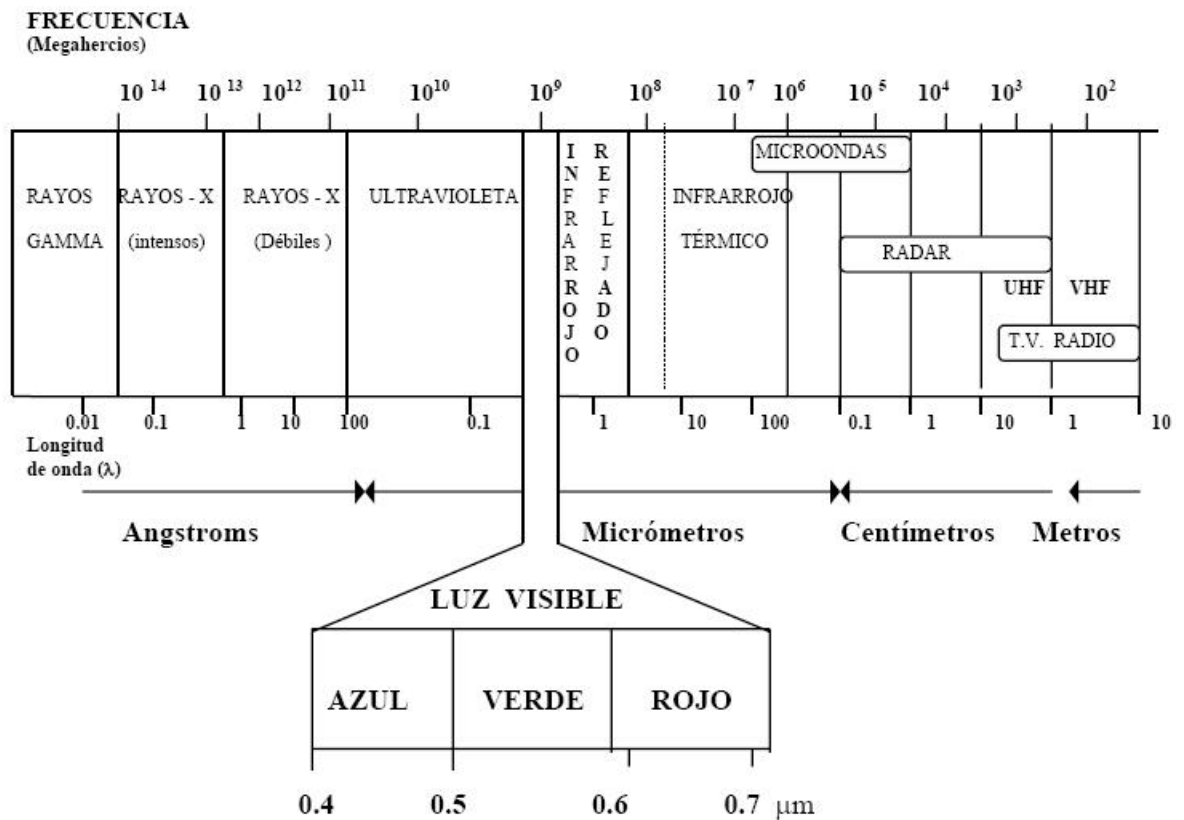


Figura 2.3: Regiones del espectro electromagnético.

## 2.3. Imágenes Satelitales

### 2.3.1. Características Generales

Una **imagen satelital** se puede definir como la representación visual de información capturada por un sensor montado en un satélite artificial. Estos sensores registran información de la superficie de la tierra, que procesada convenientemente entrega información valiosa sobre las características de la zona capturada. Cada parte del espectro que un sensor registra se llama **banda** o canal. Una banda es un rango de frecuencias detectadas del espectro electromagnético.

Un **sensor** es un dispositivo que detecta manifestaciones de cualidades o fenómenos físicos.

La forma típica de clasificar a los sensores remotos es en pasivos y activos. **Pasivos** son aquellos que se limitan a recibir la energía proveniente de un emisor exterior a ellos y **activos** cuando son capaces de emitir su propio haz de energía, siendo éstos los más flexibles, pues no dependen tanto de las condiciones exteriores.

La **resolución de un sistema sensor** es la habilidad de producir una imagen nítida y bien definida, implicando al menos cuatro manifestaciones:

- **Resolución espacial:** Designa al objeto mas pequeño que puede ser distinguido sobre una imagen.
- **Resolución espectral:** Indica el número y anchura de las bandas espectrales que puede detectar un sensor. Un sensor será mejor mientras más bandas proporcione y mientras más estrechas sean éstas. Una resolución gruesa indica que el sensor capta una amplia gama de frecuencias por cada banda. Una resolución fina indica que una banda del satélite es sensible sólo a una sección muy angosta del espectro.
- **Resolución Radiométrica:** Hace mención a la sensibilidad del sensor, esto es, a la capacidad de detectar variaciones en la radiancia espectral que recibe.
- **Resolución Temporal:** Es el intervalo de tiempo o lapso en días que un sensor tarda en observar una misma área del terreno.

Una **imagen multiespectral** se produce por la combinación de imágenes que registran diferentes longitudes de onda o bandas espectrales.

### 2.3.2. Satélites SPOT

El sistema SPOT (del francés: Systeme Polyvalent D' Observation de la Terre) que significa satélite para la observación de la tierra, fue concebido por el CNES (Centre National d'Etudes Spatiales) y realizado por Francia en colaboración con Suecia y Bélgica. Está formado por una constelación de satélites en órbita, así como por infraestructuras terrestres de control, de programación, de producción y de distribución de imágenes. Ha tenido 5 generaciones, SPOT 1,2,3,4 y 5, los cuales fueron lanzados respectivamente el 22 de febrero de 1986, el 22 de enero de 1990, el 26 de septiembre de 1993, el 24 de marzo de 1998 y el 4 de mayo del 2002. SPOT 1 y SPOT 3 salieron de servicio en el año de 1990 y el 14 de noviembre de 1996 respectivamente.

La plataforma se encuentra ubicada a 832 km. de altura, tiene una orbita heliosin-crónica y un periodo orbital de aproximadamente 101 min. Cada imagen cubre un área de 60 x 60 km con una resolución temporal de 26 días. Transporta dos sensores pasivos gemelos llamados HRV1 y HRV2 (alta resolución visible) y dado que el espejo receptor de la energía reflejada, no oscila, éste puede orientarse directamente bajo la trayectoria (visión nadiral) o a un lado de ésta (observación oblicua), lo cual permite tener oportunidades de visitar más frecuentemente un mismo objeto de estudio y tener estereoscopia.

SPOT 5[10] posee dos sensores de alta resolución del visible, HRG (del ingles: High Resolution Geometric), que ofrece la resolución de 2.5 m y alta resolución estereoscópica, HRS (del ingles: High Resolution Stereoscopic). Proporcionando una excelente precisión de localización <15 m a 90 % y en altimétrica <10 m a 90 %, sin punto de apoyo. Además, junto a los anteriores sensores, se encuentran a bordo otros instrumentos como el VEGETATION -2, que ofrece imagenes de la vegetación global con 1 km. de resolución y el DORIS, un sistema de posicionamiento.

En la tabla 2.1 se presentan las características más importantes de las diferentes generaciones de satelites SPOT.

SATELITE	SENSOR	RESOLUC. ESPECTRAL ( $\mu\text{m}$ )	RESOLUC. ESPACIAL (m)	RESOLUC. RADIOM. (bits)	ANCHO DE BARRIDO (km)
SPOT 1,2,3	Pancromát.	0.51 - 0.73	10	6	60
	Multiespec.	0.50 - 0.59	20	8	60 Banda 1(verde)
		0.61 - 0.68	20	8	60 Banda 2(Rojo)
SPOT 4	Pancromát.	0.79 - 0.89	20	8	60 Banda 3(IR cercano)
		0.61 - 0.68	10	6	60
	Multiespec.	0.50 - 0.59	20	8	60 Banda 1(verde)
		0.61 - 0.68	20	8	60 Banda 2(Rojo)
		0.78 - 0.89	20	8	60 Banda 3(IR cercano)
SPOT 5	Pancromát.	1.58 - 1.75	20	8	60 Banda 4(IR medio)
		0.61 - 0.68	2.5	6	60
	Multiespec.	0.50 - 0.59	10	8	60 Banda 1(verde)
		0.61 - 0.68	10	8	60 Banda 2(Rojo)
		0.78 - 0.89	10	8	60 Banda 3(IR cercano)
		1.58 - 1.75	20	8	60 Banda 4(IR medio)

Tabla 2.1: Características de los satélites SPOT.

Las bandas espectrales de los sensores del programa SPOT tienen diversas aplicaciones en estudios de percepción remota. El sensor **HRV** presenta aplicaciones generales (tabla 2.2), las cuales nos proporcionan una orientación inicial para la selección de las mejores bandas que pueden ser utilizadas en un proyecto en específico.

CANAL	BANDA ESPECTRAL ( $\mu\text{m}$ )	Principales Aplicaciones
1	0.50 - 0.59	Reflectancia de vegetación verde sana Mapeo de aguas
2	0.61 - 0.68	Absorción de la clorofila Diferenciación de especies vegetales Diferenciación entre suelo y vegetación
3	0.79 - 0.89	Levantamiento de fitomasa Delineamiento de cuerpos de agua
Pancromática	0.48 - 0.73	Estudio de áreas urbanas

Tabla 2.2: Aplicaciones del sensor HRV.

El formato **Dimap** es el formato de los productos SPOT. Fue introducido a mediados del año 2002 al hacerse el lanzamiento del último satélite de la familia SPOT el SPOT 5.

El formato Dimap es un formato público de descripción de datos geográficos, dedicado a los datos de imágenes, también puede soportar datos vectoriales. Se compone de dos partes: la parte imagen y la parte descriptiva.

- **Parte imagen:** Se describe por defecto en formato GeoTiff, compuesto por:
  - **Una parte TIFF:** Formato de imagen más difundido en el mundo de la generación de imágenes. Es compatible con todos los softwares del mercado.
  - **Una parte Geo:** Compatible con todos los softwares de procesamiento de la información geográfica. Contiene en el archivo TIFF básico información de georreferencia del archivo de imagen (coordenadas de la esquina superior izquierda de la imagen, dimensión de los píxeles) y eventualmente describe la proyección cartográfica y el sistema geográfico asociados.
- **Parte Descriptiva:** Se escribe en XML. El XML, similar al HTML, posee una estructura mejor definida y ofrece la posibilidad de crear sus propias palabras clave así como sus valores asociados. Directamente legible por los exploradores de Internet del mercado, se le puede asociar una hoja de estilo XSL que clasifica y compagina en HTML las informaciones contenidas en el archivo XML.

## 2.4. Reconocimiento de Patrones

### 2.4.1. Introducción

El Reconocimiento de Patrones [11] es el área de investigación que estudia la operación y el diseño de los sistemas que reconocen patrones en datos. Incluye subdisciplinas como análisis discriminante, extracción de características, error de estimación, análisis de grupos (juntos algunas veces llamados reconocimiento de patrones estadístico), inferencia gramatical y el análisis sintáctico (algunas veces llamado reconocimiento de Patrones Sintáctico).

El objetivo principal es clasificar patrones con base en un conocimiento a priori o información estadística extraída de los patrones. Los patrones a clasificar suelen ser grupos de medidas u observaciones, definiendo puntos en un espacio multidimensional apropiado.

Un **patrón** es un conjunto de características específicas que describen y representan una entidad u objeto en un contexto dado.

Una **característica o atributo** es una propiedad importante que permite distinguir total o parcialmente un objeto de otro.

Los principales enfoques en el Reconocimiento de Patrones son los siguientes [12]:

- **Enfoque Estadístico:** Basado principalmente en la teoría de la probabilidad y estadística. Supone que se tiene un conjunto de medidas numéricas con distribuciones de probabilidad conocidas, a partir de las cuales se hace el reconocimiento.
- **Enfoque Sintactico-Estructural:** Se basa en la teoría de autómatas y lenguajes formales para hacer la clasificación. Se enfoca más en la estructura de las cosas a clasificar que en mediciones numéricas. Su objetivo es construir una gramática que describa la estructura del universo de los objetos.
- **Enfoque Lógico Combinatorio:** Se basa en la idea de que la modelación del problema debe ser lo mas cercana a la realidad del mismo. Tiene como principal característica el poder trabajar con variables de todo tipo, aunque sus algoritmos suelen ser de alta complejidad.
- **Enfoque Neuronal:** Se basa en modelos matemáticos de las neuronas biológicas, tratando de emular la forma en cómo interactúan nuestras neuronas, las cuales

deben de ser “entrenadas” para dar cierta respuesta cuando se le presenten determinados valores.

Las etapas para el desarrollo de un sistema de Reconocimiento de Patrones son las siguientes:

- **Análisis y delimitación del problema:** En esta etapa se lleva a cabo la identificación de variables, constantes y particularidades del problema, además del reconocimiento de la infraestructura necesaria para su solución. La selección de una muestra representativa, las propuestas de opciones de solución y los límites en tiempo y espacio de dicho problema.
- **Identificación de todos los atributos o características útiles:** Esta etapa se logra con la observación detenidamente de la muestra representativa y con ello la identificación de los atributos medibles y los atributos útiles para resolverlo.
- **Extracción de características:** En esta etapa se extrae la información que discrimina a las variables y se elimina la información redundante o irrelevante para maximizar los atributos discriminantes.
- **Selección de características:** En esta etapa se lleva a cabo la discriminación de características por su obtención, manipulación y representación, a través de: tablas de decisión, en las cuales se busca un subconjunto mínimo de variables que no introduzca una confusión entre las clases seleccionadas ó se crea un árbol de decisión y se selecciona un conjunto de variables que permita discriminar entre clases.
- **Clasificación:** En esta etapa se asignan los patrones de clases desconocidos a una clase apropiada. Es la etapa de decisión del sistema, en el cual se realizarán los ajustes por características faltantes.
- **Post-Clasificación:** Se realizan ajustes de acuerdo al contexto y de acuerdo a los costos. Se calcula el porcentaje de error.

Uno de los problemas del reconocimiento de patrones es la clasificación, la cual busca asignar a cada entidad u objeto una clase. En Percepción Remota la clasificación de coberturas de la tierra puede ser también de dos tipos: Supervisada y no Supervisada.

## 2.4.2. Clasificación Supervisada

La clasificación supervisada consiste en categorizar nuevos objetos basándose en la información de una muestra ya clasificada. Este subconjunto muestra es utilizado como los datos de entrenamiento para los algoritmos de clasificación [11].

Un esquema general del proceso de clasificación supervisada en percepción remota se muestra en la Fig. 2.4. En él, se observa que a partir de datos de entrenamiento son clasificadas nuevas muestras, con el objetivo de obtener la imagen temática. Una imagen temática, muestra las coberturas presentes en la imagen satelital.

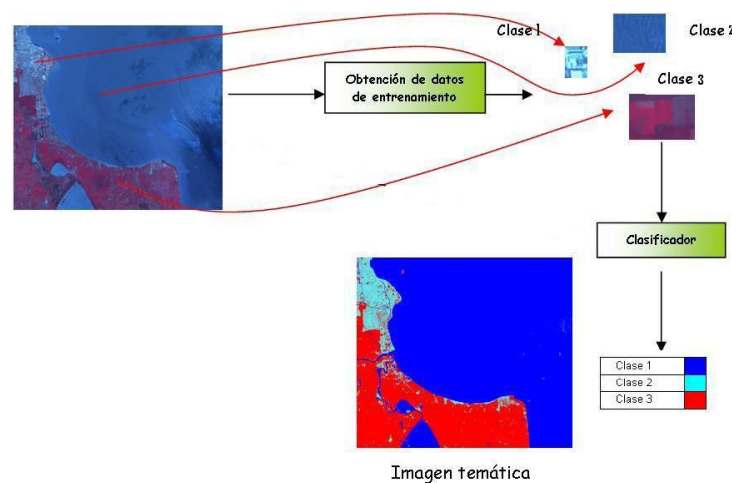


Figura 2.4: Proceso de clasificación supervisada en percepción remota.

Para lograr un mejor entendimiento sobre los métodos de clasificación definamos los siguientes conceptos:

**Vector de valores:** Sea  $x$  el vector que contiene los valores de brillo  $BV_{ijk}$  de los píxeles en la posición  $i,j$  de la banda  $k$ .

$$\begin{aligned}
 & BV_{ij1} \\
 & BV_{ij2} \\
 x = & BV_{ij3} \\
 & \vdots \\
 & BV_{ijk}
 \end{aligned} \tag{2.5}$$

**Vector de medias:** Sea  $M_c$ , el vector de medias, donde cada elemento  $\mu_{ck}$  representa el valor medio de los datos obtenidos para la clase  $c$  en la banda  $k$ .

$$M_c = \begin{matrix} \mu_{c1} \\ \mu_{c2} \\ \mu_{c3} \\ \vdots \\ \mu_{ck} \end{matrix} \quad (2.6)$$

**Matriz de covarianza:** La covarianza es la relación entre dos variables, si la covarianza es positiva las dos variables crecen al mismo tiempo, es decir, existe una relación directa entre las variables, si la covarianza es negativa quiere decir que cuando una variable crece la otra decrece, es decir, existe una relación inversa entre las variables. La matriz de covarianza es la relación que tiene una variable con todas las demás involucradas.

Sea  $V_{ckl} = V_c$  la matriz de covarianza para cada clase  $c$ , donde  $Cov_{ckl}$  es la covarianza de la clase  $c$  entre las bandas  $k$  y  $l$ .

$$V_c = V_{ckl} = \begin{matrix} Cov_{c11} & Cov_{c12} & \cdots & Cov_{c1n} \\ Cov_{c21} & Cov_{c22} & \cdots & Cov_{c2n} \\ \vdots & \vdots & \ddots & \vdots \\ Cov_{cn1} & Cov_{cn2} & \cdots & Cov_{cnn} \end{matrix} \quad (2.7)$$

Existen muchos métodos de clasificación supervisada que se aplican a datos multi-espectrales, entre los más utilizados se encuentran los siguientes:

**Distancia Mínima a la Media(DMM)[11],[13],[14],[15],[16]:** Este clasificador también conocido como Distancia Espectral, es computacionalmente simple. Emplea la distancia euclideana para clasificar y se requiere calcular el vector de medias para cada clase de los datos de entrenamiento. La idea es asignar un nuevo píxel  $x$  a la clase  $c$  que obtuvo la mínima distancia a la media.

La distancia se define como:

$$DMM(x) = \sqrt{\sum_{k=1}^n (BV_{ijk} - \mu_{ck})^2} \quad (2.8)$$

en donde  $n$  determina el número de bandas de la imagen,  $BV_{ijk}$  es el valor de brillo del píxel en la posición  $i,j$  de la banda  $k$  y  $\mu_{ck}$  es el valor medio de la clase  $c$  en la banda  $k$ .

**Distancia Mahalanobis(DMH)[11],[13],[14],[15],[16]:** La distancia Mahalanobis es similar a DMM, a excepción que se emplea la matriz de covarianza. Además, asume

que el histograma de las bandas tiene distribución normal. La DMH se define como:

$$\mathbf{DMH}(x) = (x - M_c)^T (V_c^{-1})(x - M_c) \quad (2.9)$$

en donde  $x$  es el vector que contiene los valores de brillo de los píxeles de la imagen,  $M_c$  es el vector de medias de los datos obtenidos y  $V_c$  es la matriz de covarianza.

**Máxima Probabilidad (MP)**[11],[13],[14],[15],[16]. El Clasificador de Máxima Probabilidad, asume que las estadísticas de los datos de entrenamiento para cada clase en cada banda están normalmente distribuidas, en otras palabras, si los histogramas son bi o tri-modales en una sola banda este clasificador no obtendrá buenos resultados. Entonces el píxel  $x$  se clasificará en la clase  $c$  con mayor probabilidad  $P_c \geq p_i$  donde  $i = 1, 2, \dots, c$  son las clases posibles. El clasificador se define como:

$$P_c = \{-\{0,5 \log_e[\det(V_c)]\} - [(x - M_c)^T (V_c^{-1})(x - M_c)]\} \quad (2.10)$$

La fórmula anterior se aplica cuando se asume que las clases son igualmente probables. Siendo  $V_c$  la matriz de covarianza,  $x$  es el vector que contiene los valores de brillo de los píxeles de la imagen y  $M_c$  es el vector de medias de los datos obtenidos.

### 2.4.3. Clasificación No Supervisada

La clasificación no supervisada, a diferencia de la supervisada, no utiliza muestras de entrenamiento y su objetivo es el mismo, dada una muestra no clasificada encontrar la clasificación de ésta.

La clasificación no supervisada se aplica cuando no se tienen áreas de entrenamiento disponibles, y se desarrolla a través de un análisis de "cluster". Estos clusters o agrupamientos se forman a partir de semillas o centroides, los cuales pueden ser asignados por los usuarios, o pueden ser seleccionados arbitrariamente. Durante el proceso, cada objeto es asociado con un centro de clase basandose sobre un criterio de similitud.

Entre los algoritmos de clasificación no supervisada más importantes se encuentra el Clasificador ISODATA[11]. **ISODATA** proviene de Iterative Self-Organizing Data Analysis Techniques (Técnica de Análisis de Datos Auto-Organizados), al que se le ha agregado la letra A para que sea un nombre fácil de pronunciar.

Se trata de un método de agrupación de datos cuyo esqueleto es el algoritmo K-medias[11], al que se le han agregado parámetros y operaciones heurísticas.

ISODATA es de naturaleza interactiva con el usuario, lo que obliga a que su implementación disponga de una buena interfaz para que el usuario del programa pueda tener una participación activa en su ejecución.

Los parámetros esenciales que maneja el algoritmos son:

- $N_c$ : Es el número deseado o estimado a priori de las clases.
- $\theta_N$ : Número mínimo de miembros o elementos de una clase para constituirse como tal.
- $\theta_S$ : Desviación Estandar Máxima que servirá para aplicar un criterio de división de un grupo o clase en dos, cuando la desviación típica  $\sigma$  del grupo sea superior a  $\theta_S$ .
- $\theta_c$ : Es un parametro de unión de dos agrupamientos. Se utilizará para comprobar si la distancia Euclídea entre dos agrupamientos es menor que  $\theta_c$ , en cuyo caso seran dos grupos a fusionar.
- $L$ : Es el número máximo de parejas de agrupamientos que pueden ser fusionados.
- $I$ : Número máximo de iteraciones que puede ejecutar el algoritmo.

El algoritmo es el siguiente:

1. **Inicialización:** Se estima el número de agrupamientos necesarios y se selecciona un centro inicial para cada agrupamiento. En esta fase el usuario puede elegir el centro de cada agrupamiento o los puntos pueden ser seleccionados arbitrariamente.
2. **Distribución de los elementos entre las diferentes clases:** Se asignan los elementos a ser clasificados  $(x_1, x_2, \dots, x_n)$  al centro de la clase más cercana, usando el principio de la mínima distancia euclídea:

$$\text{Distancia Euclídea} = \sqrt{\sum_{k=1}^n (BV_{ijk} - \mu_{ck})^2} \quad (2.11)$$

3. **Eliminación de clases con número insuficiente de clases:** Se eliminan los agrupamientos con un número inferior de elementos a  $\theta_c$ , actualizando también el número de agrupamientos  $\theta_N$ .

4. **Actualización de los centros de las clases:** Esta actualización se realiza calculando la media de cada grupo o clase  $\mu_{ck}$ .
5. **Cálculo de la distancia euclídea media de cada cluster( $D_j$ ):** Se obtiene para cada clase la distancia euclídea media de sus elementos respecto al correspondiente centroide. Este parametro devuelve la medida de dispersión de las muestras de cada clase respecto de su media, el cual servirá posteriormente junto con otras condiciones, para la posible división de agrupamientos.
6. **Cálculo de la distancia media de todos los agrupamientos( $D$ ):** Se calcula la distancia media total de las muestras de los centros de los agrupamientos.
7. **Comprobación de bifurcaciones:** En este paso se verifica si el número de agrupamientos es menor o igual a la mitad del número de agrupamientos ingresado por el usuario. Si es menor o igual existe una posible división de agrupamientos y se prosigue con el paso 8, en caso contrario se verifica que el número de agrupamientos actual sea mayor o igual al doble de los agrupamientos ingresados por el usuario, en caso afirmativo existe una posible unión de agrupamientos y se prosigue con el paso 11, otro caso se prosigue con el paso 14.
8. **Cálculo de la desviación estandar de cada grupo( $\sigma_i^{(j)}$ ):** Éste es el primer paso para realizar la división de agrupamientos, en el cual se encuentra el vector de desviaciones estandar de cada agrupamiento.
9. **Obtención de las desviaciones estandar maximas de cada grupo( $\sigma_{max}^{(j)}$ ):** Se selecciona en cada clase la componente mayor del correspondiente vector de desviaciones estandar, formandose un grupo de desviaciones estandar máximas de cada agrupamiento.
10. **Posible división de agrupamientos:** Un agrupamiento puede ser dividido solo si cumple las siguientes condiciones:
  - $\sigma_{max}^{(j)} > \theta_S$
  - $D_j > D$
  - $N_j > 2\theta_N$

11. **Cálculo de las distancias entre clases**( $D_{ij}$ ): Para realizar la unión de dos clases, se calculan previamente todas las distancias entre parejas de agrupamientos.
12. **Posible unión**: Se comparan estas distancias  $D_{ij}$  con el parametro  $\theta_c$  de forma que se toman las  $L$  más pequeñas en orden creciente.
13. **Proceso de unión**: Para realizar el proceso de unión de agrupamientos se toman las parejas de agrupamientos con las distancias menores. Si y sólo si ninguno de los agrupamientos ha sido fusionado con otro agrupamiento en esta misma iteración, entonces se forma un agrupamiento único cuyo centroide es:  $Z_{ij} = 1/(N_i + N_j) * (N_i Z_i + N_j Z_j)$  siendo  $N_i$  y  $N_j$  el número de muestras de los agrupamientos  $\alpha_i$  y  $\alpha_j$  respectivamente antes de la fusión. En cada unión se actualiza el número de agrupamientos  $N_c$ .
14. **Comprobación de la última iteración**: Se comprueba si se ha llegado a la última iteración. En caso que se trate de la última iteración el algoritmo termina y en caso contrario se regresa al paso 2.

## 2.5. Métodos de Evaluación

### 2.5.1. Matriz de Confusión

Para valorar la confiabilidad de los métodos de clasificación y tener una medida cuantitativa de cómo estos clasificadores se están desempeñando es necesario una evaluación de los resultados de cada clasificador.

Existen diferentes métodos para evaluar a los clasificadores [11], entre los que encontramos:

- Método de resustitución
- Validación simple o Hold out
- Leave one out
- Validación cruzada
- Bootstrap

**Método de resustitución:** Este método usa los mismos datos para entrenar y para probar, por lo tanto los resultados obtenidos son optimistas. Es recomendado para sistemas de clasificadores lineales.

**Validación simple o Hold out:** Este método selecciona aleatoriamente del conjunto de datos disponibles dos conjuntos aleatoriamente excluyentes: datos para entrenamiento (habitualmente  $2/3$  del total) y datos para prueba (habitualmente  $1/3$  del total). Ambos subconjuntos deben ser representativos de las clases. El subconjunto de entrenamiento es usado para entrenar el clasificador y el subconjunto de prueba para estimar la tasa de error verdadera. Debido a que sólo un experimento se realiza, el resultado puede ser engañoso si se hizo una división desafortunada, por lo tanto es considerado como una evaluación pesimista. Existe otro método que es una generalización del método de validación simple, este método se llama de **Remuestreo o Random Subsampling** en el cual se repite  $k$  veces el procedimiento anterior sobre particiones independientes del subconjunto de entrenamiento y el subconjunto de muestra, así la estimación de la tasa de error se efectúa a partir de la media de las tasas de error obtenidas en los diferentes experimentos.

**Leave one out:** La principal característica de este método es la de separar  $k$  datos y entrenar con los demás, después se hace una evaluación con los  $k$  datos apartados, este procedimiento se repite  $n$  veces separando otros  $k$  datos diferentes. Con esto se obtiene una media de los resultados ofreciendo resultados no sesgados por la elección de los datos de prueba.

**Validación cruzada:** Es una modificación del Leave-K-out. El método consiste en dividir aleatoriamente el conjunto de datos  $D$  en  $K$  partes disjuntas  $D_1, \dots, D_K$ , procurando que tengan tamaños parecidos (Fig. 2.5). Con esto el clasificador es entrenado y entrenado  $K$  veces, excluyendo en el entrenamiento cada vez uno de los subconjuntos. El método entrena con todos los subconjuntos  $D_i$  excepto el  $k$ -ésimo ya que ese subconjunto será el de prueba. La media de las tasas de error de los subconjuntos de muestra se utilizará como estimación de la tasa de error verdadero.

Este método garantiza haber provado con todos los datos y es ligeramente pesimista.

Cuando la validación se realiza con subconjuntos mezclados aleatoriamente se denomina **validación cruzada estratificada**. De este modo se consigue que una clase determinada aparezca con la misma probabilidad en todos los subconjuntos de validación. Su objetivo es conseguir que en cada subconjunto  $D_i$  haya igual número de

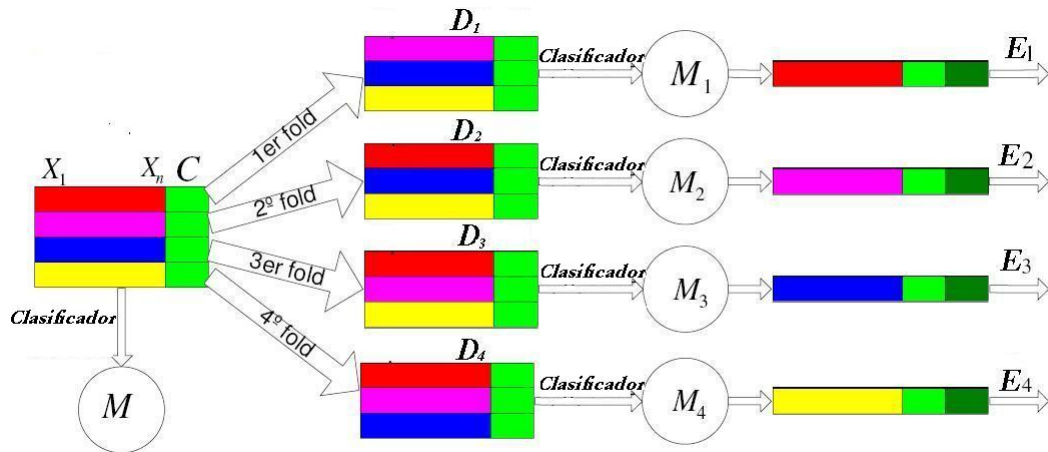


Figura 2.5: Ejemplo de validación cruzada con  $k = 4$

ejemplos de cada clase representada.

$$\text{Error de estimacion} = \frac{1}{k} \sum_{i=1}^k E_i \quad (2.12)$$

**Bootstrap:** En este método se seleccionan las muestras de entrenamiento de forma aleatoria, permitiendo la repetición de los ejemplos. Esto ocasiona que en un conjunto de entrenamiento puede que no estén representadas todas las clases o que en un conjunto de entrenamiento puede haber muestras repetidas varias veces.

Una **matriz de confusión** es un diagrama usado para evaluar la cuantificación de la exactitud de un clasificador durante la clasificación supervisada.

Dicha matriz contiene información sobre clasificaciones reales y predichas elaboradas por un sistema de clasificación, la cual nos permite ver sobre una tabla la distribución de los errores cometidos por un clasificador a lo largo de las distintas categorías. El desempeño ó exactitud de los clasificadores se evalúa usando los datos en la matriz.

Un ejemplo de matriz de confusión se muestra en la Fig. 2.6. Las filas en la tabla enumeran las clases e indican como los píxeles etiquetados de cada clase son asignados a clases por el clasificador. Uno puede ver así rápidamente qué muestras fueron etiquetadas correctamente y qué muestras incorrectamente. Un valor de exactitud puede ser obtenido para cada clase. El valor de exactitud evaluado por filas se llama **exactitud del usuario** (user's accuracy) ó **Error por comisión**, en la cual un píxel puede ser asignado a la clase a la que no pertenece. Mientras que la exactitud evaluada por columnas se

conoce como la **exactitud del productor** (producer's accuracy) ó **error por omisión**, en la que un píxel no es asignado la clase apropiada [13].

Nombre de la clase	Error por comisión (%)	No. de ejemplos	Número de muestras en la clase						
			Techos	Calles	Rutas	Pasto	Arboles	Agua	Refugios
Techos	98.8	15364	15176	96	91	0	0	0	1
Calles	93.4	1812	75	1693	0	0	2	2	40
Rutas	99.5	429	2	0	427	0	0	0	0
Pasto	99.9	5904	0	0	0	5900	4	0	0
Arboles	99.4	1748	0	0	0	11	1737	0	0
Agua	80.3	4791	383	3	0	0	7	3847	551
Refugios	87.5	279	14	19	0	0	1	1	244
<b>TOTAL</b>		30327	15650	1811	518	5911	1751	3850	836
<b>Error por omisión (%)</b>			97.0	93.5	82.4	99.8	99.2	99.9	29.2
<b>Desempeño global de las clases (29024/30327) = 95.7%</b>									
<b>Estadística Kappa (X100) = 93.6%</b>									

Figura 2.6: Ejemplo de matriz de confusión

La exactitud total ó **Desempeño Global** es el número total de objetos correctamente clasificados divididos entre el número total de objetos clasificados (es decir la suma total de las observaciones de la diagonal principal de la matriz de confusión dividida por el número total de observaciones). Es quizás el medio más simple para determinar mediante un solo número la exactitud del clasificador:

$$\text{Desempeño Global} = \frac{\text{objetos clasificados correctamente}}{\text{total de objetos clasificados}} \quad (2.13)$$

Las **Estadística Kappa** [13] es otro medio para tener acceso a la exactitud del clasificador y es el método más completo para expresar el valor de exactitud de cada clase.

La estadística Kappa es calculada mediante la siguiente formula:

$$K = \frac{N \sum_{i=1}^r x_{ii} - \sum_{i=1}^r (x_{i+} x_{+i})}{N^2 - \sum_{i=1}^r (x_{i+} x_{+i})} \quad (2.14)$$

donde  $r$  es el número de filas en la matriz de confusión,  $x_{ii}$  es el número de observaciones en la fila  $i$  y columna  $i$ ,  $x_{i+}$  y  $x_{+i}$  son los totales de la fila  $i$  y columna  $i$ , respectivamente, y  $N$  es el número total de observaciones.

# Capítulo 3

## Análisis y Diseño del Sistema

### 3.1. Ingeniería de Software

La ingeniería de software es una disciplina que comprende todos los aspectos de la producción de software desde las etapas iniciales de la especificación del sistema, hasta el mantenimiento de éste después de que se utiliza [17].

Para la implementación de nuestro sistema tomamos en cuenta las siguientes etapas basadas en UML [18](Unified Modeling Language), el cual es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software.

#### 3.1.1. Descripción del Problema

La problemática surge en el Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE), ya que se tienen diferentes imágenes satelitales, en las cuales su interpretación y análisis es de suma importancia para algunos investigadores expertos en el área.

Para la interpretación y el análisis de las imágenes es necesario desarrollar un software que sea capaz de clasificar imágenes satelitales y generar su mapa temático teniendo la capacidad de mostrar la información más relevante contenida en dicha imagen generada. Por lo que las principales funciones del software a desarrollar son las siguientes:

El sistema debe de leer las diferentes bandas de las imágenes satelitales. En este caso como las imágenes satelitales manejadas en dicho Instituto son del satélite SPOT

5 el sistema debe leer las cuatro bandas que registra dicho satélite.

El sistema debe de permitir hacer la elección del número de clases a identificar, así como también el número de muestras a tomar por cada clase.

El sistema en principio debe tener implementado 3 clasificadores supervisados y uno no supervisado, siendo los siguientes: Clasificador de Distancia Mínima a la Media, el Clasificador de Distancia Mahalanobis y el Clasificador de Máxima Probabilidad. Por otro lado, uno no supervisado: ISODATA.

Para poder realizar la clasificación, el sistema debe tener implementado dos opciones para obtener datos de entrenamiento: una opción debe obtener los datos de entrenamiento por medio del usuario (el usuario elige de forma arbitraria los datos de entrenamiento) y la otra opción es utilizando crecimiento de regiones.

También, el sistema debe considerar la opción de evaluar los diferentes clasificadores supervisados. De este modo, el sistema debe tener una opción que indique el grado de exactitud y desempeño de cada clasificador mediante la matriz de confusión.

Por último, el sistema debe mostrar la imagen generada, así también como los datos de referencia geográfica en cada uno de los píxeles de dicha imagen.

### 3.1.2. Casos de Uso

Uno de los primeros pasos en el análisis y diseño de sistemas mediante UML, es pensar en el funcionamiento del sistema: identificar a los actores que interactúan con el sistema y sus diferentes funciones para cada uno.

Esta identificación de actores y funciones que desempeña cada uno (casos de uso) sirven para modelar los requisitos del sistema desde la perspectiva del usuario. Esta etapa es llamada **Modelado de Casos de Uso** y se utiliza para modelar cómo un sistema funciona actualmente, o cómo los usuarios desean que funcione.

En la descripción del problema se logró identificar sólo **un Usuario** para dicho sistema, el cual puede ser un usuario experto en el tema o un principiante y las funciones identificadas que realiza el usuario son las siguientes:

- Obtención de datos de entrenamiento de forma manual.
- Obtención de datos de entrenamiento forma automática.
- Obtención de datos de entrenamiento nuevos.

- Obtención de datos de entrenamiento almacenados.
- Elección del clasificador de Distancia Mínima a la Media (DMM).
- Elección del clasificador de Distancia Mahalanobis (DMH).
- Elección del clasificador de Máxima Probabilidad (MP).
- Elección del clasificador ISODATA.
- Obtención de información de georreferenciación.
- Validación del desempeño del clasificador.

Los primero cuatro puntos corresponden a un sólo caso de uso al que llamaremos **Obtención de datos de entrenamiento**, ya que estos puntos se relacionan de forma directa unos con otros y corresponden a un sólo propósito: obtener muestras de entrenamiento que permitan realizar una clasificación.

Los siguientes cinco corresponden a un solo caso de uso al que llamaremos **Clasificación**, ya que su función principal es generar una nueva imagen temática y mostrar las coordenadas de mapa que corresponden a cada píxel.

Y el último punto corresponde a un caso de uso al que llamaremos **Validación de Resultados** ya que obtiene el desempeño de cualquier clasificador supervisado ejecutado.

El diagrama de Casos de Uso generado queda de la siguiente manera:

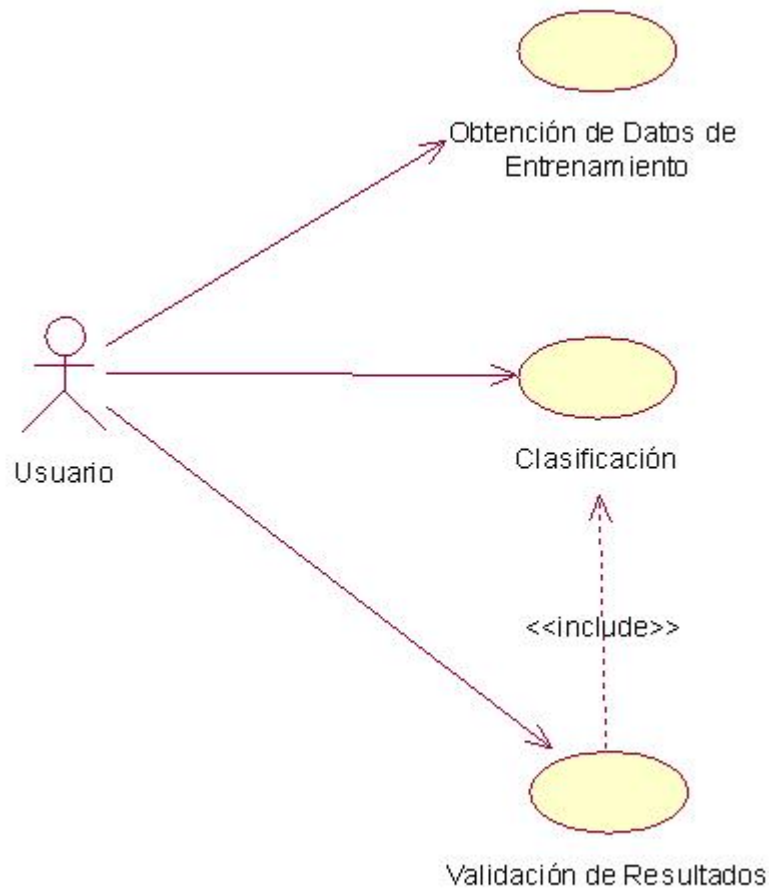


Figura 3.1: Diagrama de Casos de Uso para el sistema de Clasificación.

### 3.1.3. Especificación de Casos de Uso

La principal función de este proceso es la de detallar de forma específica a cada uno de los casos de uso, las especificaciones de casos de uso describen propiamente la funcionalidad del sistema.

Por cada caso de uso se genera una especificación de casos de uso. En el **Apéndice A** se describe de manera muy detallada la especificación de los 3 casos de uso identificados en el problema.

### **3.1.4. Escenarios**

Los escenarios describen un flujo de eventos a través de un caso de uso, en los cuales el usuario del sistema interactúa con el mismo para lograr un objetivo. Están clasificados en dos modalidades: los escenarios primarios, también llamados happy day, los cuales representan la forma en la que el sistema debe funcionar idealmente o la mayoría de las veces, y los escenarios secundarios que son los que representan rutas alternativas de ejecución o error.

En los escenarios uno debe de realizar tantos como uno necesite para entender el funcionamiento del sistema a desarrollar.

A continuación se describen los posibles escenarios que un usuario puede tener al estar ejecutando el sistema en los tres casos de uso.

#### **Escenarios para el caso de uso: Obtención de Datos de Entrenamiento**

##### **1.- "Obtención de datos de entrenamiento nuevos y en forma manual"**

- 1.1.- El usuario ejecuta el sistema.
- 1.2.- El usuario ingresa el número  $k$  en el cuadro de texto que especifica el número de clases que se van a considerar.
- 1.3.- El usuario ingresa el número  $n$  en el cuadro de texto que especifica el número de muestras que se tomarán en cuenta por cada clase.
- 1.4.- El usuario ingresa la dirección de las 4 imágenes a ser leídas en su correspondiente campo de datos.
- 1.5.- El usuario elige la opción de cargar datos nuevos.
- 1.6.- El usuario elige la opción de obtención de datos manual.
- 1.7.- El usuario ejecuta el botón Iniciar.
- 1.8.- El sistema determina que las opciones ingresadas cumplan con todas las validaciones requeridas y después muestra dos pantallas, una en donde muestra una imagen y otra en donde solicita al usuario ingresar el nombre de la clase que desea designar para elegir sus datos de entrenamiento.
- 1.9.- El usuario ingresa en el cuadro de texto el nombre de la clase.

- 1.10.- El sistema solicita que seleccione el área de entrenamiento a considerar como muestra.
- 1.11.- El usuario con ayuda del ratón selecciona el área de entrenamiento.
- 1.12.- El sistema obtiene y guarda los valores obtenidos del área de entrenamiento de acuerdo a la clase elegida.
- 1.13.- El sistema ejecuta el paso 1.10  $n-1$  veces, para obtener el número de muestras elegidas por el usuario.
- 1.14.- Una vez terminado de ejecutarse el paso 1.13, el sistema regresa a ejecutar el paso 1.9  $k-1$  veces para obtener el nombre de las clases a clasificar.
- 1.15.- El sistema obtiene los valores medios de cada una de las clases capturadas en los datos de entrenamiento y los guarda.

**Ver figuras 3.2 y 3.14**

## 2.- “Error en validación de clases y muestras”

- 2.1.- El usuario ejecuta el sistema.
- 2.2.- El usuario ingresa el número  $k$  en el cuadro de texto que especifica el número de clases que se van a considerar.
- 2.3.- El usuario ingresa el número  $n$  en el cuadro de texto que especifica el número de muestras que se tomarán en cuenta por cada clase.
- 2.4.- El usuario ingresa la dirección de las 4 imágenes a ser leídas en su correspondiente campo de datos.
- 2.5.- El usuario elige la opción cargar datos nuevos.
- 2.6.- El usuario elige la opción de obtención de datos manual.
- 2.7.- El sistema verifica que las opciones ingresadas cumplan con todas las validaciones requeridas.
- 2.8.- El sistema encuentra un error en el número  $k$  que especifica el número de clases tomadas en cuenta, ya que este número se encuentra fuera del rango permitido para elegir el número de clases.

- 2.9.- El sistema envía un mensaje de error especificando el rango permitido para ingresar el número de clases a tomar.
- 2.10.- El sistema regresa al paso 2.2.

**Ver figuras 3.3 y 3.15**

### **3.- "Error en lectura de imágenes de entrada"**

- 3.1.- El usuario ejecuta el sistema.
- 3.2.- El usuario ingresa el número  $k$  en el cuadro de texto que especifica el número de clases que se van a considerar.
- 3.3.- El usuario ingresa el número  $n$  en el cuadro de texto que especifica el número de muestras que se tomarán en cuenta por cada clase.
- 3.4.- El usuario ingresa la dirección de las 4 imágenes a ser leídas en su correspondiente campo de datos.
- 3.5.- El usuario elige cargar datos nuevos.
- 3.6.- El usuario elige la opción de obtención de datos manual.
- 3.7.- El sistema verifica que las opciones ingresadas cumplan con todas las validaciones requeridas.
- 3.8.- El sistema lee las imágenes desde la ruta especificada.
- 3.9.- El sistema encuentra un error al leer una de las imágenes.
- 3.10.- El sistema muestra un mensaje de error, especificando la imagen y el tipo de error.
- 3.11.- El sistema se cierra.

**Ver figuras 3.4 y 3.16**

### **4.- "Obtención de datos de entrenamiento nuevos y en forma automática"**

- 4.1.- El usuario ejecuta el inicio del sistema.

- 4.2.- El usuario ingresa el número  $k$  en el cuadro de texto que especifica el número de clases que se van a considerar.
- 4.3.- El usuario ingresa el número  $n$  en el cuadro de texto que especifica el número de muestras que se van a tomar en cuenta por cada clase.
- 4.4.- El usuario ingresa la dirección de las 4 imágenes a ser leídas en su correspondiente campo de datos.
- 4.5.- El usuario elige cargar datos nuevos.
- 4.6.- El usuario elige la opción de obtención de datos automática.
- 4.7.- El sistema verifica que las opciones ingresadas cumplan con todas las validaciones requeridas.
- 4.8.- El sistema lee las imágenes desde la ruta especificada.
- 4.9.- El sistema muestra una pantalla en donde se visualiza una nueva imagen generada del producto de la combinación de las imágenes leídas.
- 4.10.- El sistema muestra una pantalla en donde se le especifica al usuario que ingrese el nombre de la clase que desea designar para elegir sus datos de entrenamiento.
- 4.11.- El usuario ingresa en el cuadro de texto el nombre de la clase.
- 4.12.- El sistema solicita que seleccione el píxel base de entrenamiento para generar el crecimiento de regiones.
- 4.13.- El usuario con ayuda del ratón selecciona el píxel de entrenamiento.
- 4.14.- El sistema obtiene y guarda los valores obtenidos del área de entrenamiento generadas por el crecimiento de regiones de los píxeles con clases similares al píxel ingresado.
- 4.15.- El sistema regresa a ejecutar el paso 4.12  $n-1$  veces, para obtener el número de muestras elegidas por el usuario.
- 4.16.- Una vez ejecutado el paso 4.15, el sistema regresa a ejecutar el paso 4.10  $k - 1$  veces para obtener el nombre de la clase a clasificar.
- 4.17.- El sistema obtiene los valores medios de cada una de las clases capturadas en los datos de entrenamiento y los guarda.

**Ver figuras 3.5 y 3.17**

**5.- “Obtención de datos de entrenamiento almacenados”**

- 5.1.- Este paso se ejecuta solamente cuando ya se hayan obtenido al menos una vez datos de entrenamiento de forma manual.
- 5.2.- El usuario elige la opción cargar datos almacenados.
- 5.3.- El usuario ejecuta el botón Iniciar.
- 5.4.- El sistema obtiene los valores de entrenamiento almacenados de las clases.

**Ver figuras 3.6 y 3.18**

**Escenarios para el caso de uso: Clasificación**

**1.- “Clasificador de Distancia Mínima a la Media”**

- 1.1.- El usuario selecciona la opción de Distancia Mínima a la Media.
- 1.2.- El usuario ejecuta el botón de iniciar el clasificador.
- 1.3.- El sistema ejecuta el algoritmo para dicha clasificación
- 1.4.- El sistema genera una imagen temática.
- 1.5.- El sistema muestra en pantalla la imagen temática.
- 1.6.- El usuario recorre alguna parte de la imagen mostrada con ayuda el ratón.
- 1.7.- El sistema obtiene las coordenadas de mapa terrestres de la imagen con base en el punto registrado por la posición del ratón.
- 1.8.- El sistema muestra las coordenadas terrestres obtenidas de la imagen en el punto registrado.

**Ver figuras 3.7 y 3.19**

**2.- “Clasificador de Distancia Mahalanobis”**

- 2.1.- El usuario selecciona la opción de Distancia Mahalanobis.

- 2.2.- El usuario ejecuta el botón de iniciar el clasificador.
- 2.3.- El sistema ejecuta el algoritmo para dicha clasificación
- 2.4.- El sistema genera una imagen temática.
- 2.5.- El sistema muestra en pantalla la imagen temática.
- 2.6.- El usuario recorre alguna parte de la imagen mostrada con ayuda del ratón .
- 2.7.- El sistema obtiene las coordenadas de mapa de la imagen con base en el punto registrado por la posición del ratón.
- 2.8.- El sistema muestra las coordenadas terrestres obtenidas de la imagen en el punto registrado.

**Ver figuras 3.8 y 3.20**

### **3.- “Clasificador de Máxima Probabilidad”**

- 3.1.- El usuario selecciona la opción de Máxima Probabilidad.
- 3.2.- El usuario ejecuta el botón de iniciar el clasificador.
- 3.3.- El sistema ejecuta el algoritmo para dicha clasificación.
- 3.4.- El sistema genera una imagen temática.
- 3.5.- El sistema muestra en pantalla la imagen temática.
- 3.6.- El usuario recorre alguna parte de la imagen mostrada con ayuda del ratón.
- 3.7.- El sistema obtiene las coordenadas de mapa terrestres de la imagen con base en el punto registrado por la posición del ratón.
- 3.8.- El sistema muestra las coordenadas terrestres obtenidas de la imagen en el punto registrado.

**Ver figuras 3.9 y 3.21**

### **4.- “Clasificador ISODATA”**

- 4.1.- El usuario selecciona la opción de ISODATA.
- 4.2.- El usuario ejecuta el botón de iniciar el clasificador.

- 4.3.- El sistema muestra una pantalla en donde solicita al usuario que ingrese los datos requeridos para ejecutar dicho clasificador.
- 4.4.- El usuario ingresa los datos solicitados, los cuales son: el número deseado de clases, los miembros mínimos de cada clase, la desviación estándar máxima, la mínima distancia entre medias, el número máximo de iteraciones, y el número máximo de parejas a unir por iteración.
- 4.5.- El sistema verifica que los datos ingresados sean correctos y que estén dentro del rango especificado.
- 4.6.- El sistema ejecuta el clasificador seleccionado.
- 4.7.- El sistema muestra en pantalla la imagen temática.
- 4.8.- El usuario recorre alguna parte de la imagen mostrada con ayuda del ratón.
- 4.9.- El sistema obtiene las coordenadas de mapa terrestres de la imagen con base en el punto registrado por la posición del ratón.
- 4.10.- El sistema muestra las coordenadas terrestres obtenidas de la imagen en el punto dado.

**Ver figuras 3.10 y 3.22**

#### **5.- "Error en datos ingresados del Clasificador ISODATA"**

- 5.1.- El usuario selecciona la opción de ISODATA.
- 5.2.- El usuario ejecuta el botón de iniciar el clasificador.
- 5.3.- El sistema muestra una pantalla en donde solicita al usuario que ingrese los datos requeridos para ejecutar dicho clasificador.
- 5.4.- El usuario ingresa los datos solicitados, los cuales son: el número deseado de clases, los miembros mínimos de cada clase, la desviación estándar máxima, la mínima distancia entre medias, el número máximo de iteraciones, y el número máximo de parejas a unir por iteración.
- 5.5.- El sistema verifica que los datos ingresados sean correctos y que estén dentro del rango especificado.

- 5.6.- El sistema registra un error en los datos ingresados y regresa la información generada en donde se encontró el error.
- 5.7.- El sistema muestra una pantalla en donde se especifica el error generado.
- 5.8.- El sistema regresa al paso 5.3.

**Ver figuras 3.11 y 3.23**

### **Escenarios para el caso de uso: Validación de Resultados**

#### **1.- "Matriz de confusión"**

- 1.1.- El usuario ingresa el número  $k$  de subconjuntos a dividir el archivo.
- 1.2.- El usuario ejecuta el botón de generar matriz de confusión.
- 1.3.- El sistema verifica que el dato ingresado sea correcto y se encuentre dentro del rango especificado.
- 1.4.- El sistema verifica la opción seleccionada en el caso de uso de clasificación.
- 1.5.- El sistema lee los resultados obtenidos de clasificación de dicho clasificador elegido.
- 1.6.- El sistema divide el archivo de resultados en  $k$  partes.
- 1.7.- El sistema ejecuta nuevamente el mismo clasificador para obtener la matriz de confusión.
- 1.8.- El sistema obtiene y guarda los datos de la validación cruzada.
- 1.9.- El sistema muestra los resultados de la matriz de confusión.

**Ver figuras 3.12 y 3.24**

#### **2.- "Error en obtención de datos de la matriz de confusión"**

- 2.1.- El usuario ingresa el número  $k$  de subconjuntos a dividir el archivo.
- 2.2.- El usuario ejecuta el botón de generar matriz de confusión.

- 2.3.- El sistema verifica que el dato ingresado sea correcto y se encuentre dentro del rango especificado.
- 2.4.- El sistema genera un error al verificar el dato ingresado.
- 2.5.- El sistema muestra un mensaje de error al usuario, especificando cuál fue el error.
- 2.6.- El sistema regresa al paso 2.1.

**Ver figuras 3.13 y 3.25**

### **3.1.5. Diagramas de Secuencia**

Los diagramas UML de secuencia y de colaboración (llamados diagramas de interacción) se utilizan para modelar los aspectos dinámicos de un sistema. Un diagrama de interacción consiste en un conjunto de objetos y sus relaciones, incluyendo los mensajes que se pueden enviar entre ellos.

Los diagramas de secuencia destacan el orden temporal de los mensajes. Los diagramas de colaboración destacan la organización estructural de los objetos que envían y reciben mensajes.

Los diagramas de secuencia representan la interacción del actor y del sistema a través del tiempo. La creación de los diagramas de secuencia depende de la formulación de los casos de uso. Los casos de uso indican cómo los actores interactúan con el sistema. Durante la operación del sistema, los actores generan eventos, solicitando alguna operación a cambio. Para cada escenario descrito le corresponde un diagrama de secuencia, por lo tanto tendremos tantos diagramas de secuencia como escenarios tengamos. Un diagrama de secuencia muestra los objetos de un escenario mediante líneas verticales y los mensajes entre objetos como flechas conectando objetos. Los mensajes son dibujados cronológicamente desde arriba hacia abajo y los rectángulos en las líneas verticales representan los periodos de actividad de los objetos.

A continuación se muestran los diagramas de secuencia de los escenarios de los casos de uso descritos anteriormente.

## Diagramas de secuencia para el caso de uso: Obtención de Datos de Entrenamiento

### 1.- Obtención de Datos de entrenamiento nuevos en forma manual

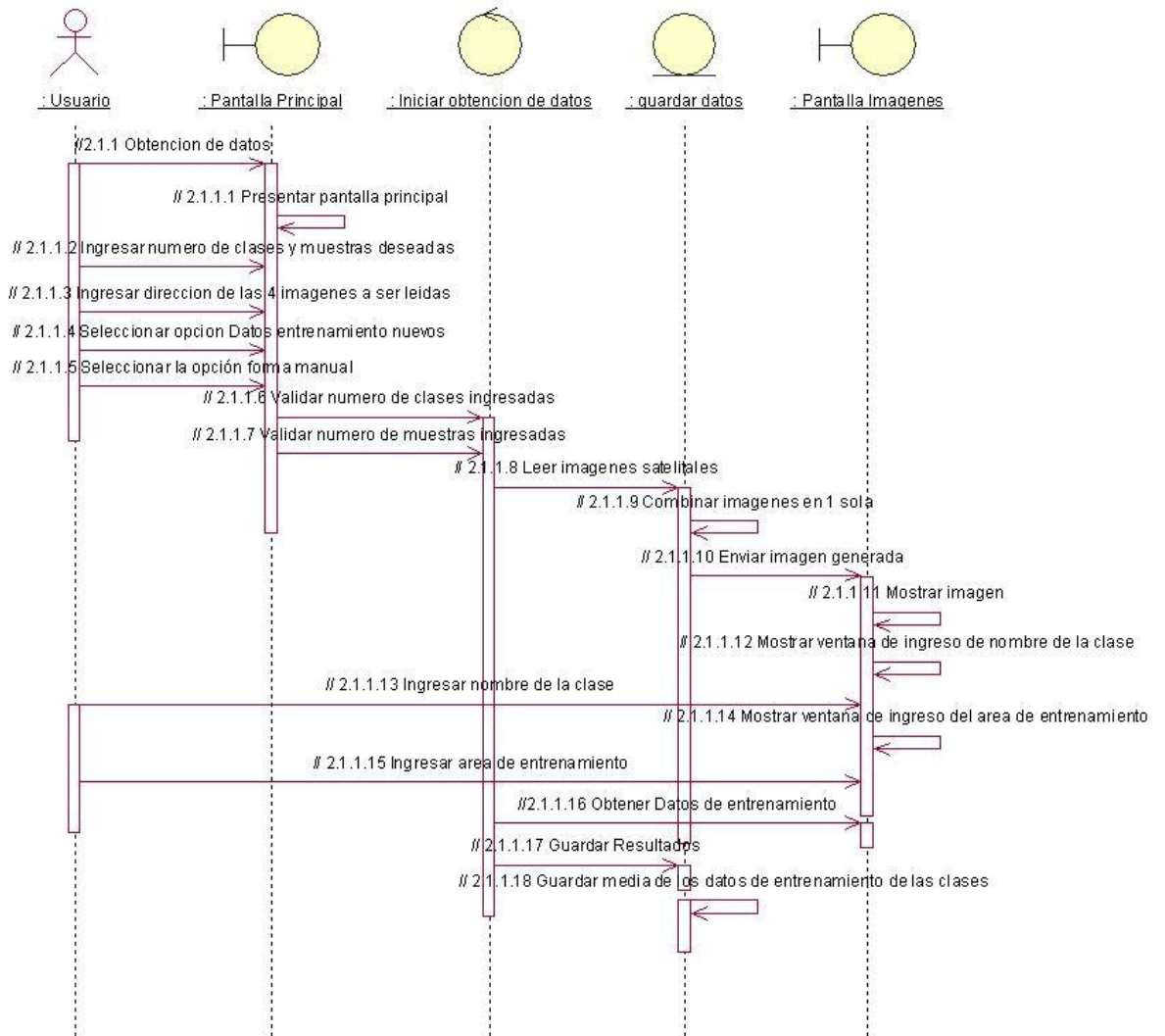


Figura 3.2: Obtención de Datos de entrenamiento nuevos de forma manual.

## 2.- Error en Validación de Clases y Muestras

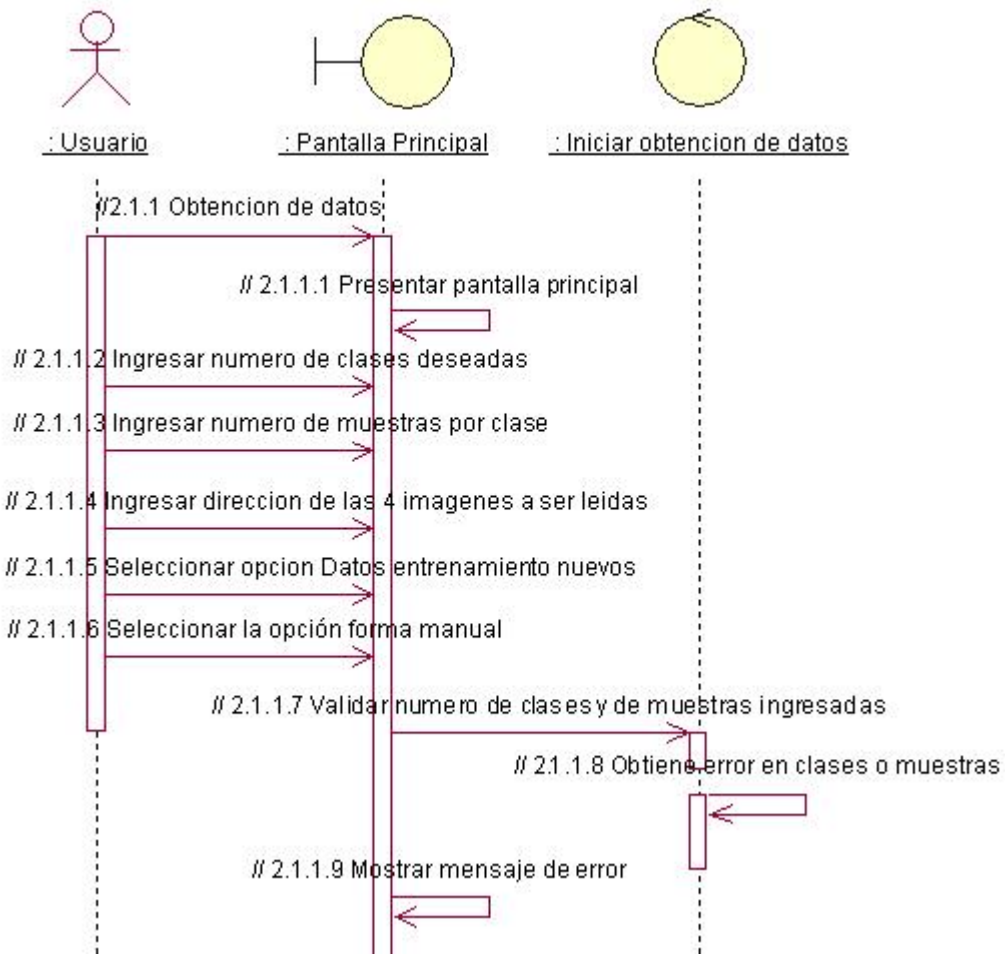


Figura 3.3: Error validacion de Clases y Muestras.

### 3.- Error en Lectura de Imagenes de Entrada

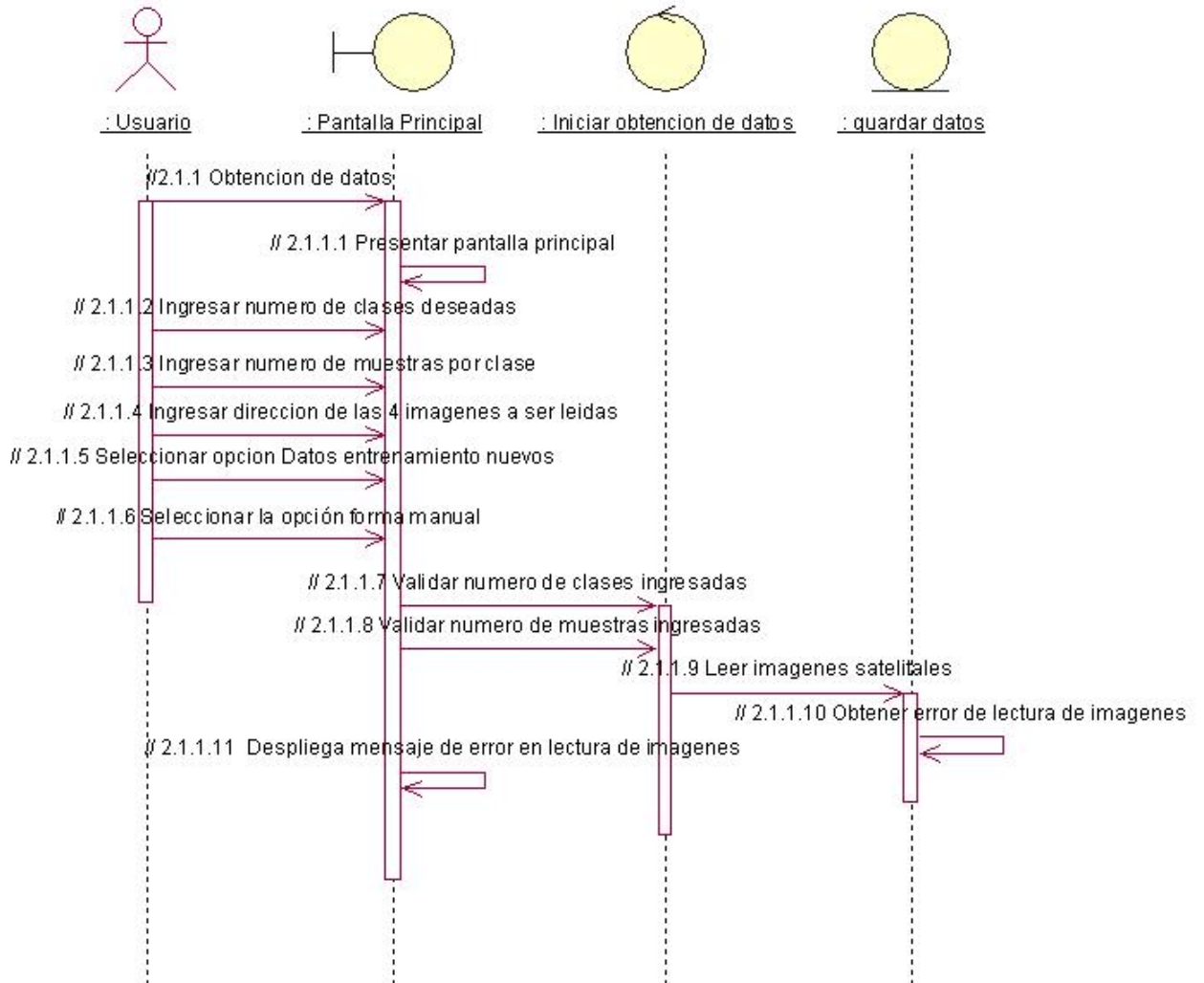


Figura 3.4: Error en lectura de imágenes de entrada.

#### 4.- Obtención de Datos de Entrenamiento Nuevos en Forma Automatica

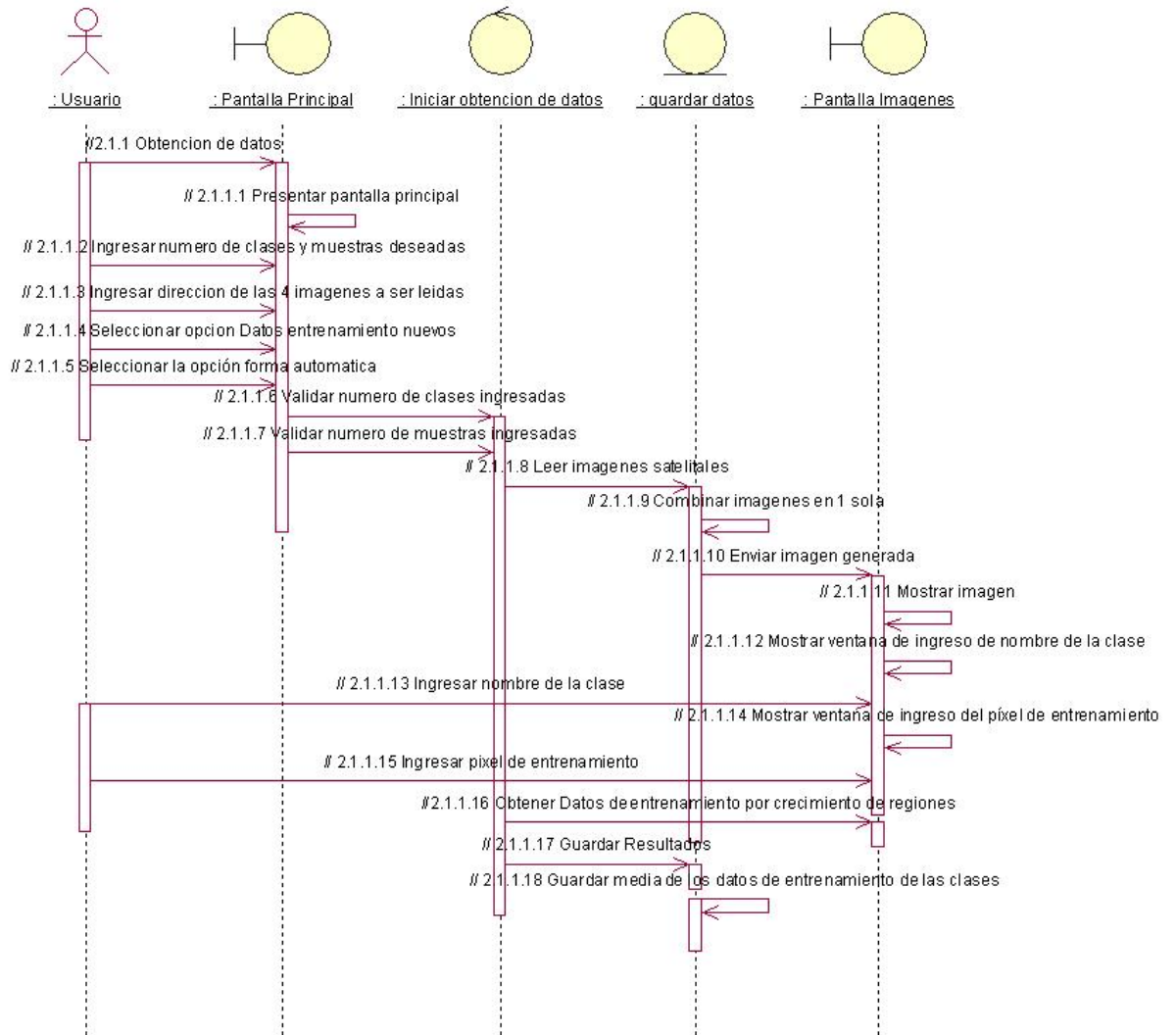


Figura 3.5: Obtención de datos de entrenamiento nuevos en forma automatica.

## 5.- Obtención de Datos de Entrenamiento Almacenados

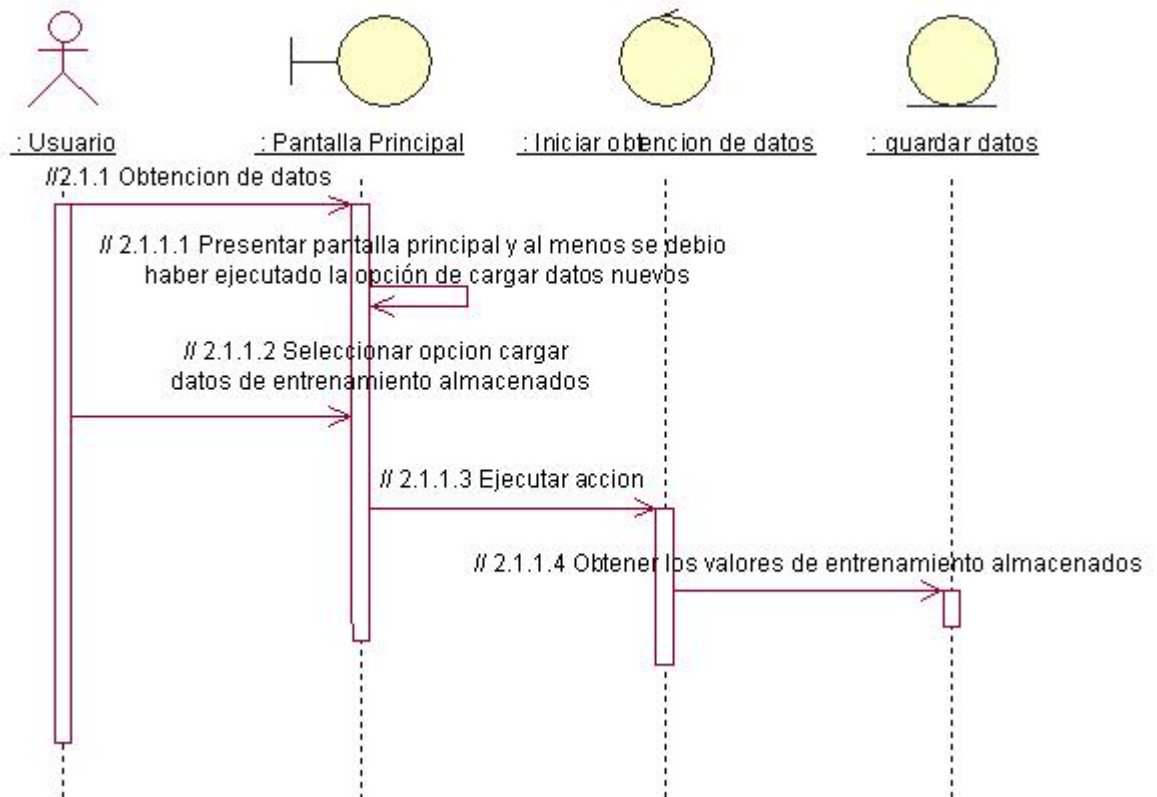


Figura 3.6: Obtención de datos de entrenamiento almacenados.

## Diagramas de secuencia para el caso de uso: Clasificación

### 1.- Clasificador de Distancia Mínima a la Media

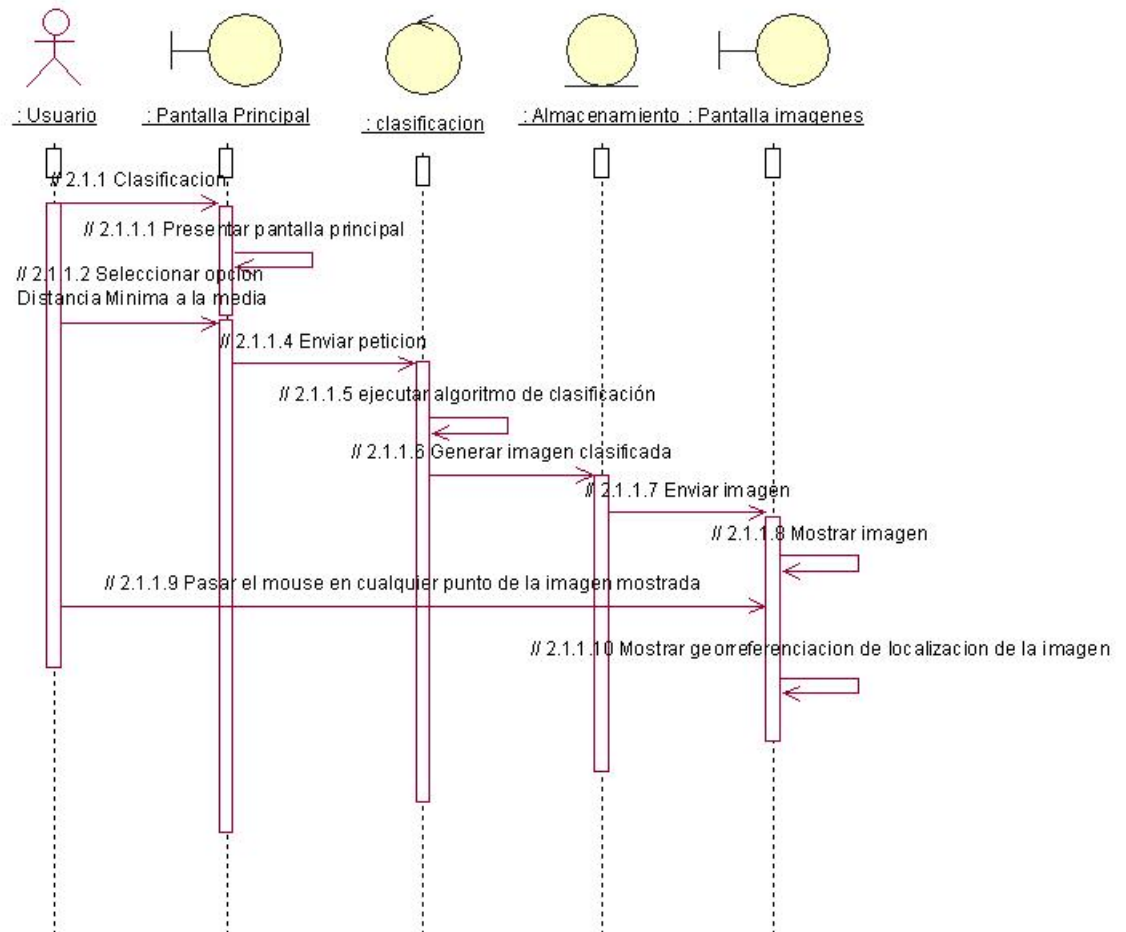


Figura 3.7: Clasificador de Distancia Mínima a la Media.

## 2.- Clasificador de Distancia Mahalanobis

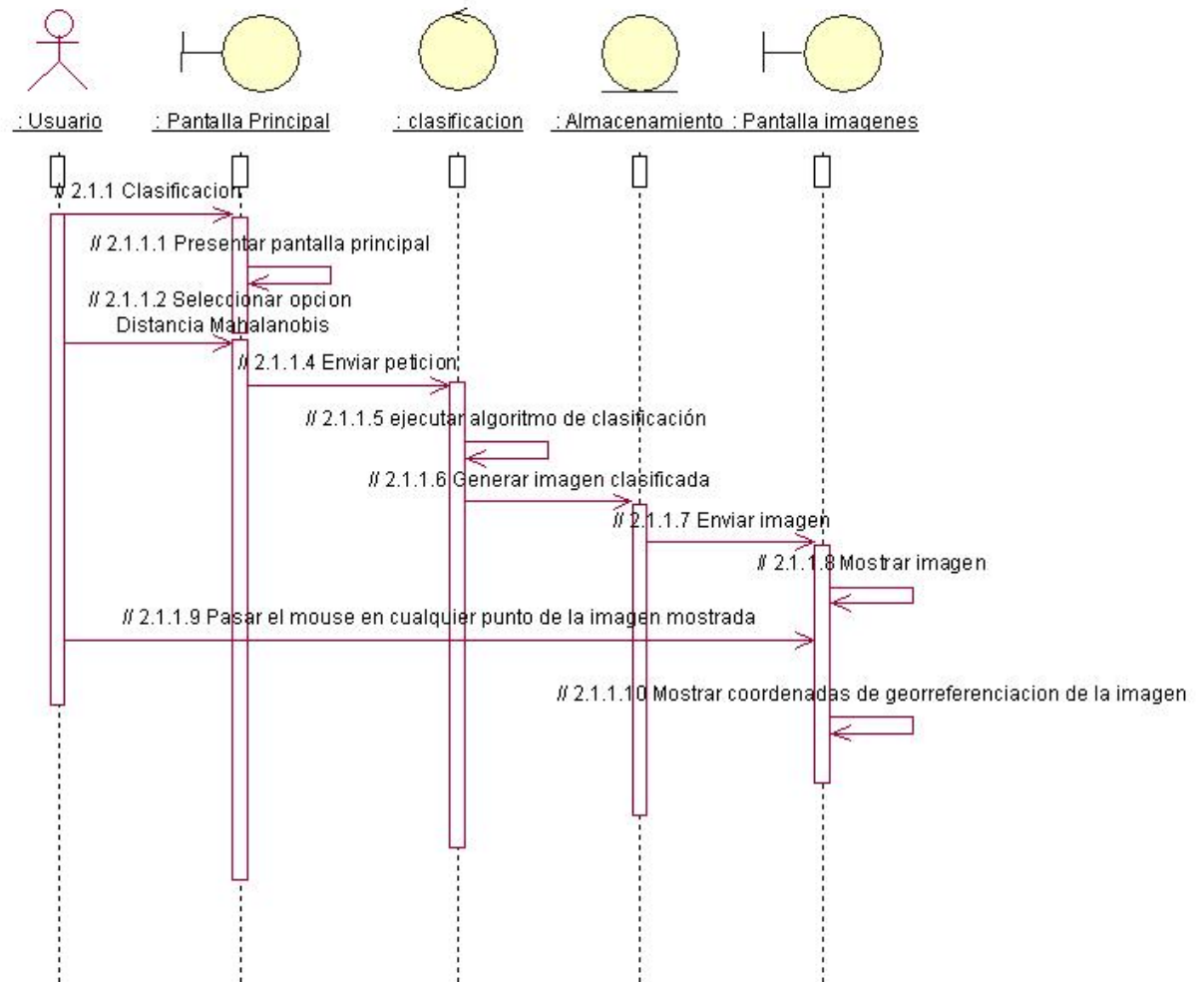


Figura 3.8: Clasificador de Distancia Mahalanobis.

### 3.- Clasificador de Máxima Probabilidad

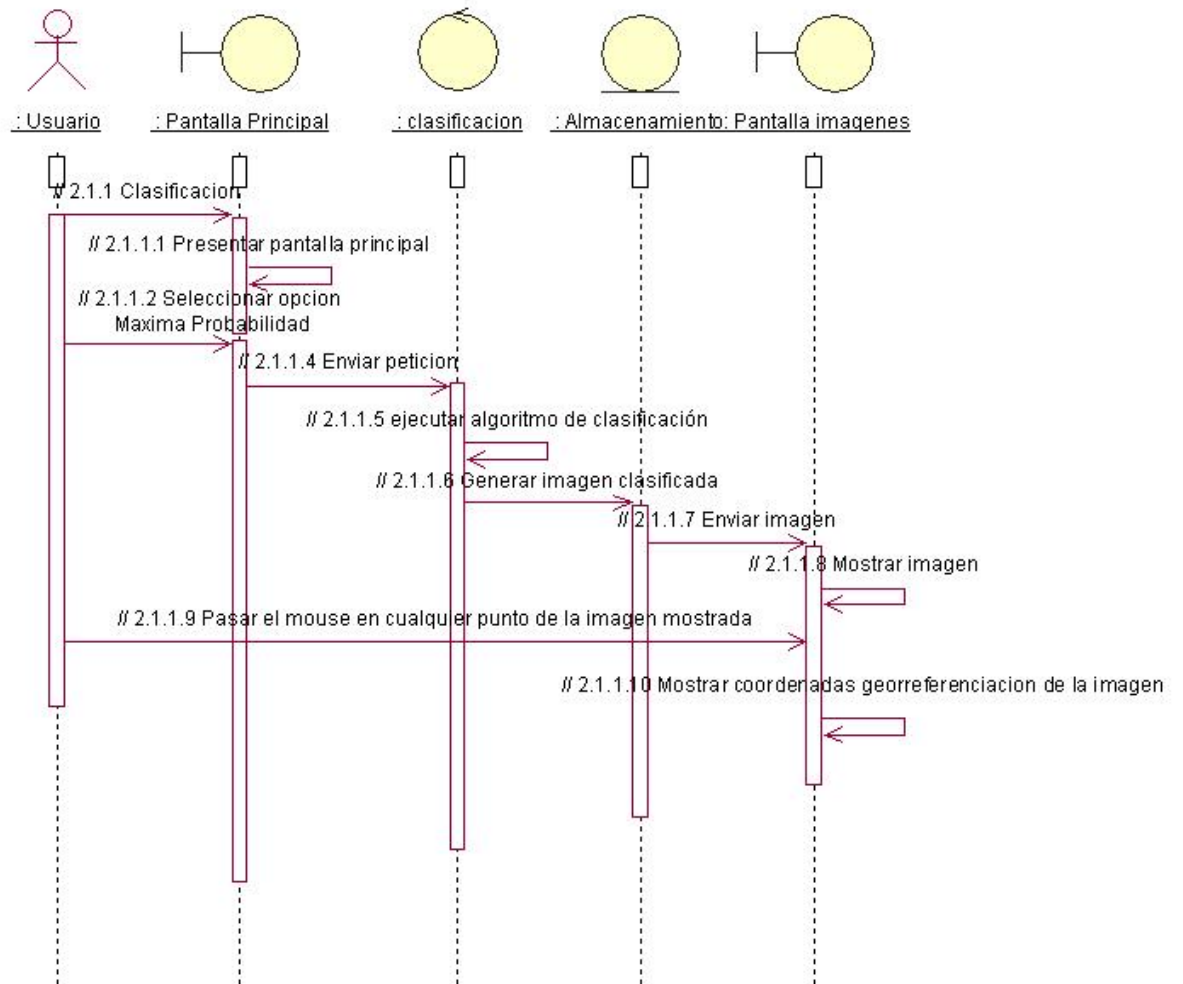


Figura 3.9: Clasificador de Máxima Probabilidad.

#### 4.- Clasificador ISODATA

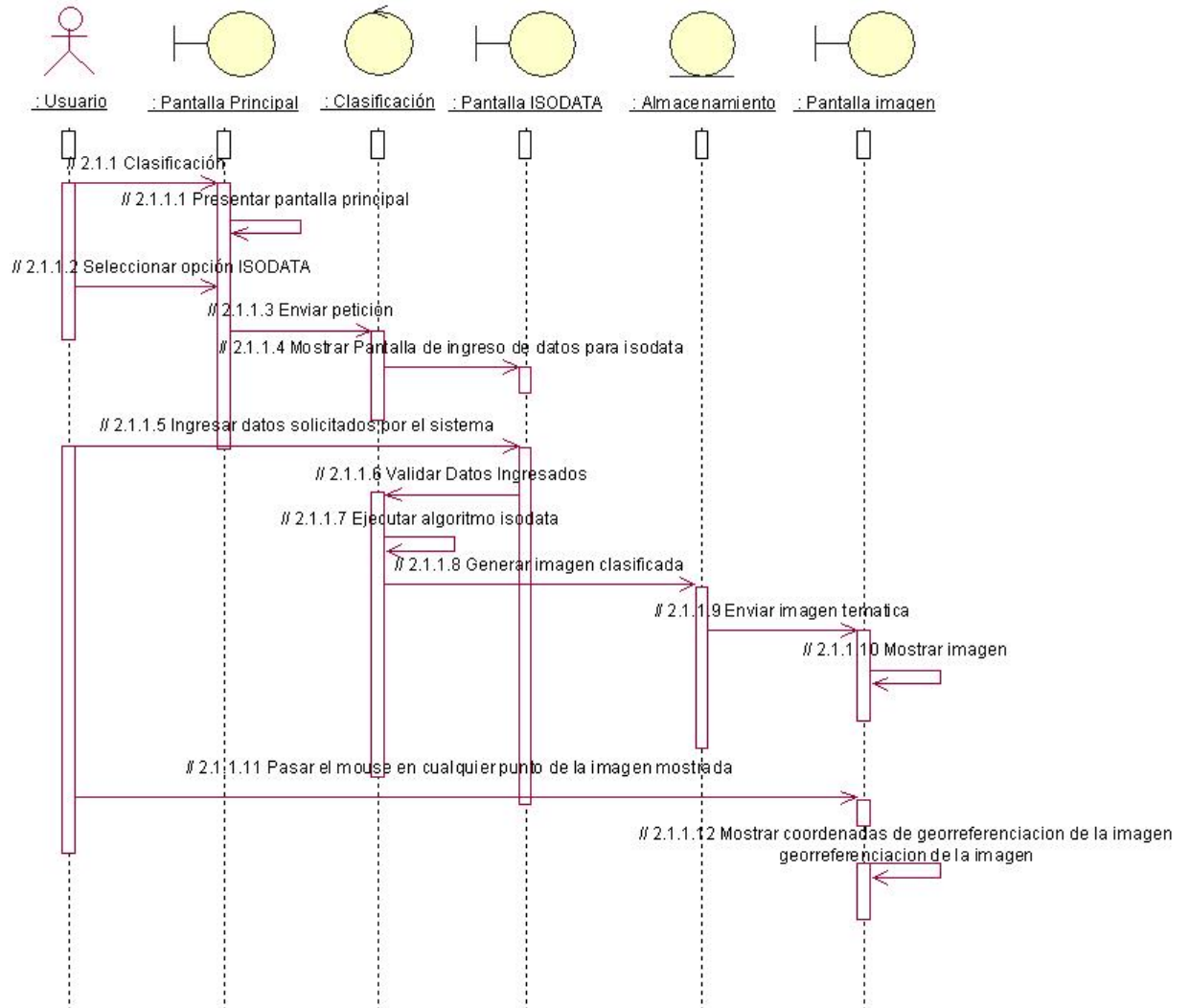


Figura 3.10: Clasificador ISODATA.

## 5.- Error en Datos Ingresados del Clasificador ISODATA

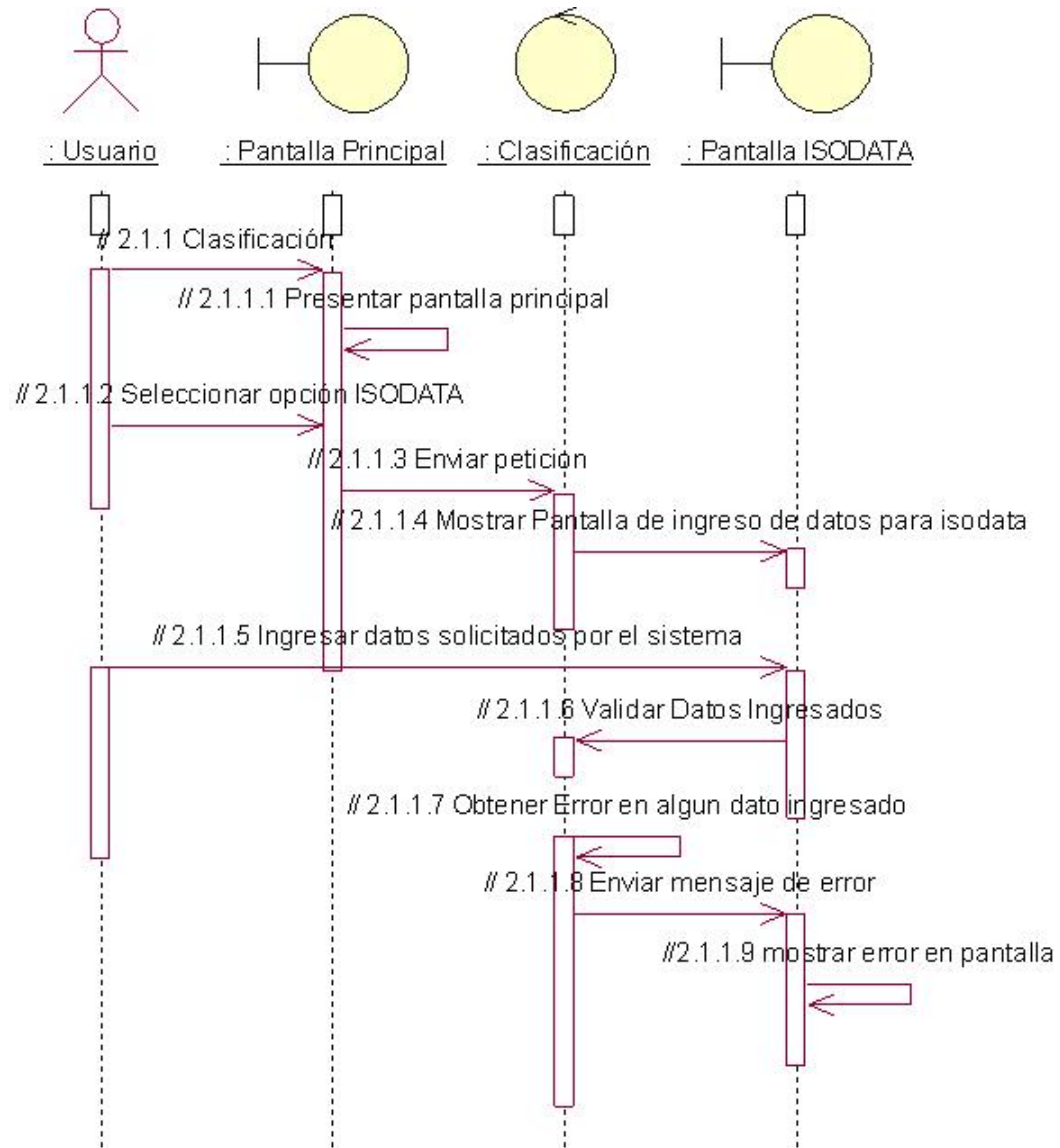


Figura 3.11: Error al ingresar datos ISODATA.

## Diagramas de secuencia para el caso de uso: Validación de Resultados

### 1.- Matriz de Confusión

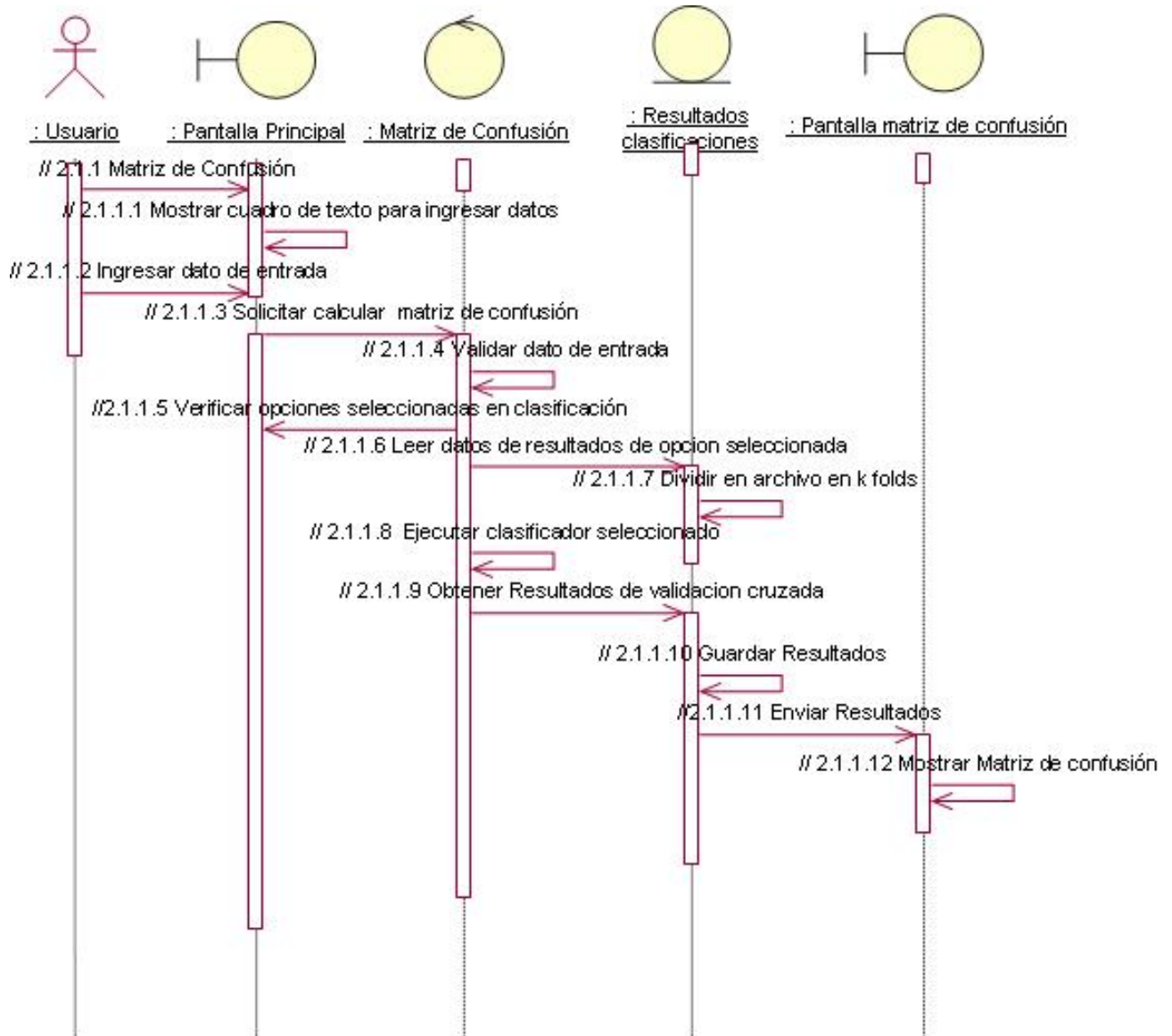


Figura 3.12: Matriz de confusión.

## 2.- Error en Obtención de datos de la Matriz de Confusión

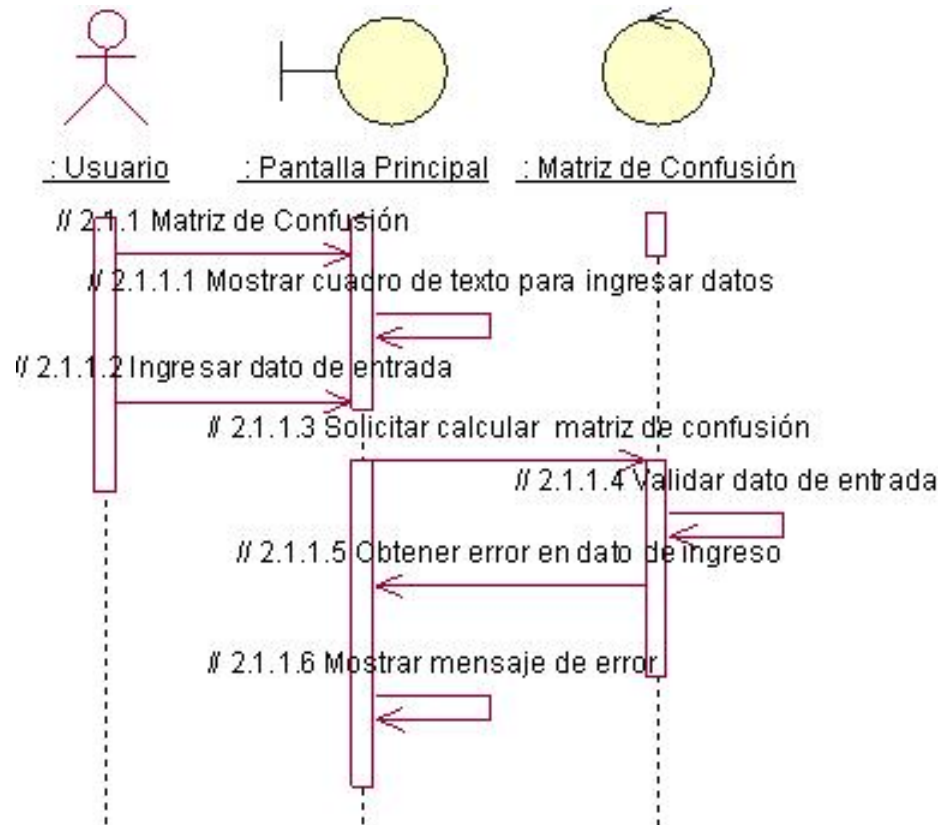


Figura 3.13: Error en obtención de datos de la matriz de confusión

### 3.1.6. Diagramas de Colaboración

Un diagrama de colaboración muestra una interacción organizada basándose en los objetos que toman parte en la interacción y los enlaces entre los mismos. A diferencia de los diagramas de secuencia, los diagramas de colaboración muestran las relaciones entre los roles de los objetos, en la cual los objetos están conectados por enlaces (links) en los cuales se representan los mensajes enviados acompañados de una flecha que indica su dirección. La secuencia de los mensajes y los flujos de ejecución concurrentes deben determinarse explícitamente mediante números de secuencia.

Mientras que el diagrama de secuencia se centra en la secuencia cronológica del escenario que estamos modelando, el diagrama de colaboración se centra en estudiar

todos los efectos de un objeto dado durante un escenario. El diagrama de colaboración ofrece una mejor visión del escenario cuando el analista está intentando comprender la participación de un objeto en el sistema.

A continuación se presentan los diagramas de colaboración de los casos de uso antes descritos.

### Diagramas de Colaboración para el caso de uso: Obtención de Datos de Entrenamiento

#### 1.- Obtención de datos de entrenamiento nuevos en forma manual

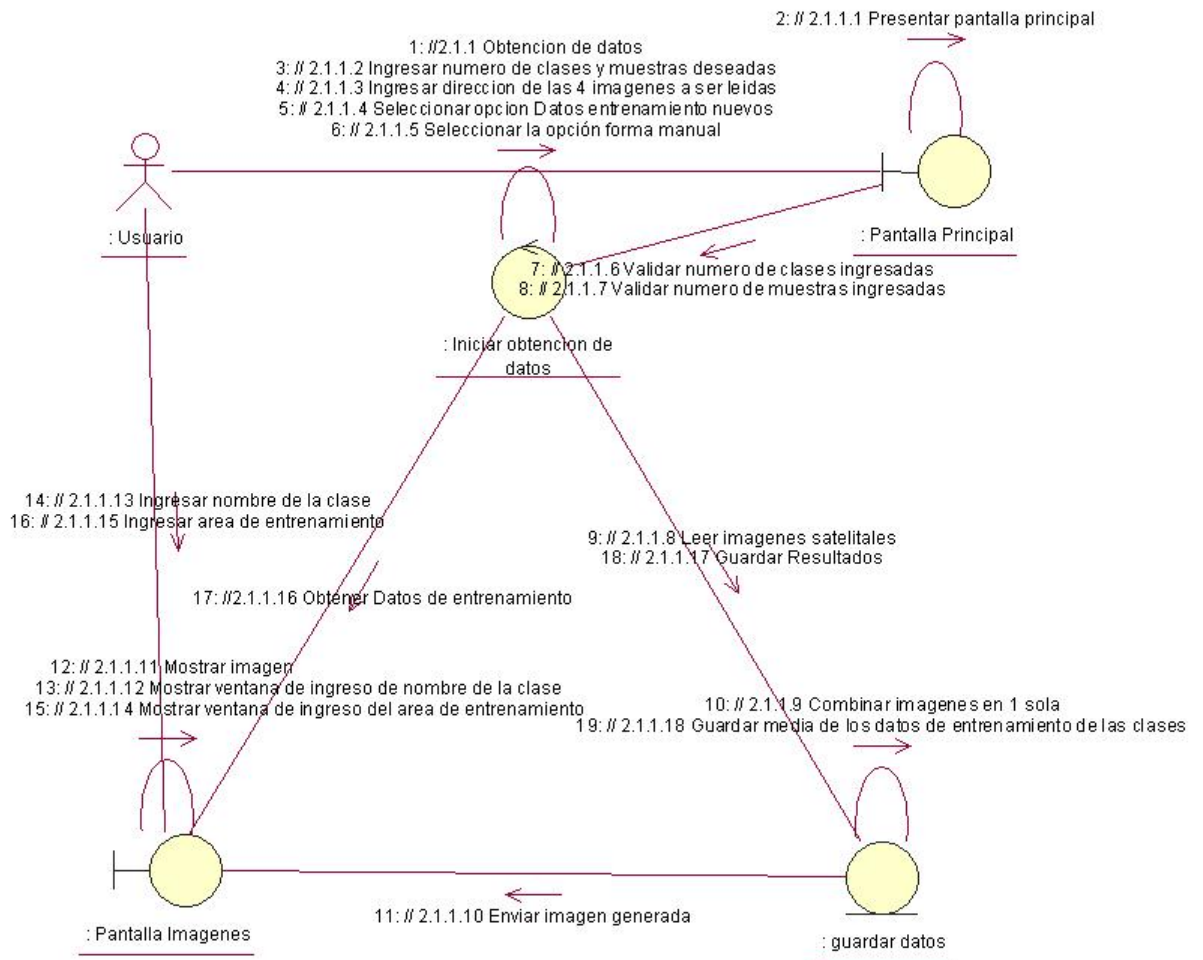


Figura 3.14: Obtención de datos de entrenamiento nuevos en forma manual.

## 2.- Error en Validación de Clases y Muestras

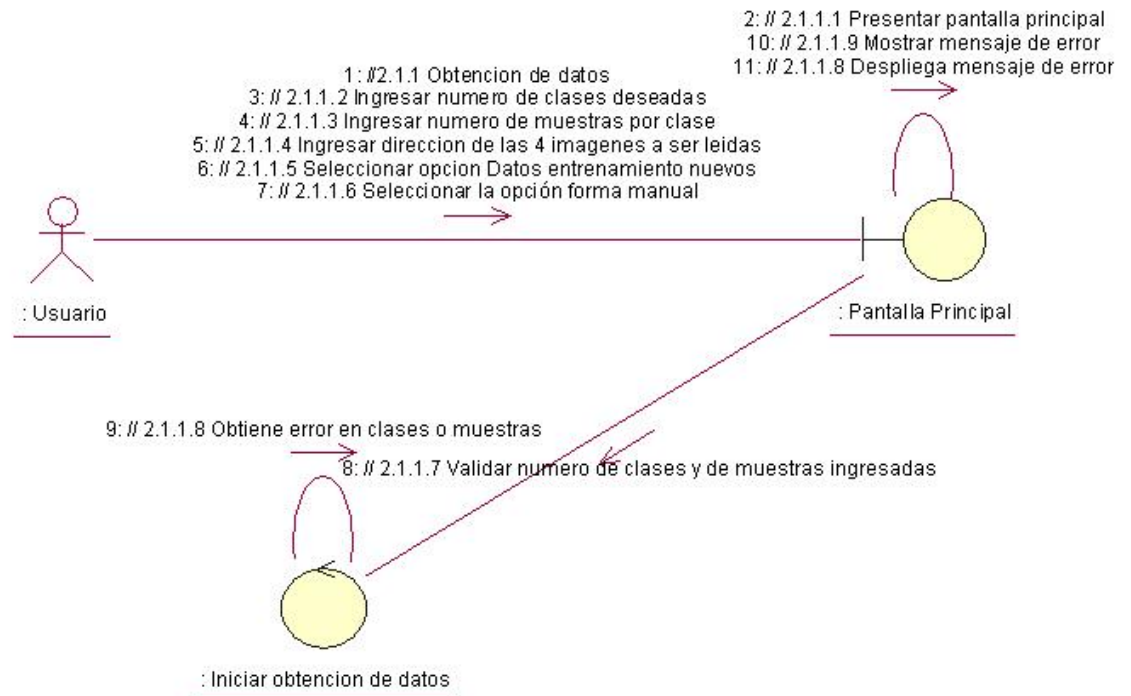


Figura 3.15: Error validacion de Clases y Muestras.

### 3.- Error en Lectura de Imagenes de Entrada

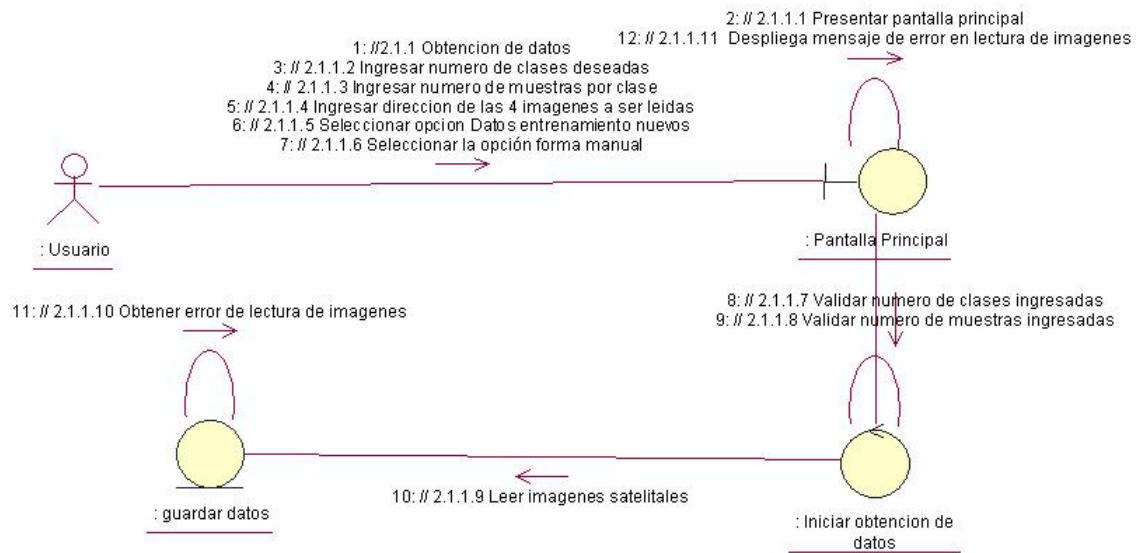


Figura 3.16: Error en lectura de imagenes de entrada.

#### 4.- Obtención de Datos de Entrenamiento Nuevos en Forma Automatica

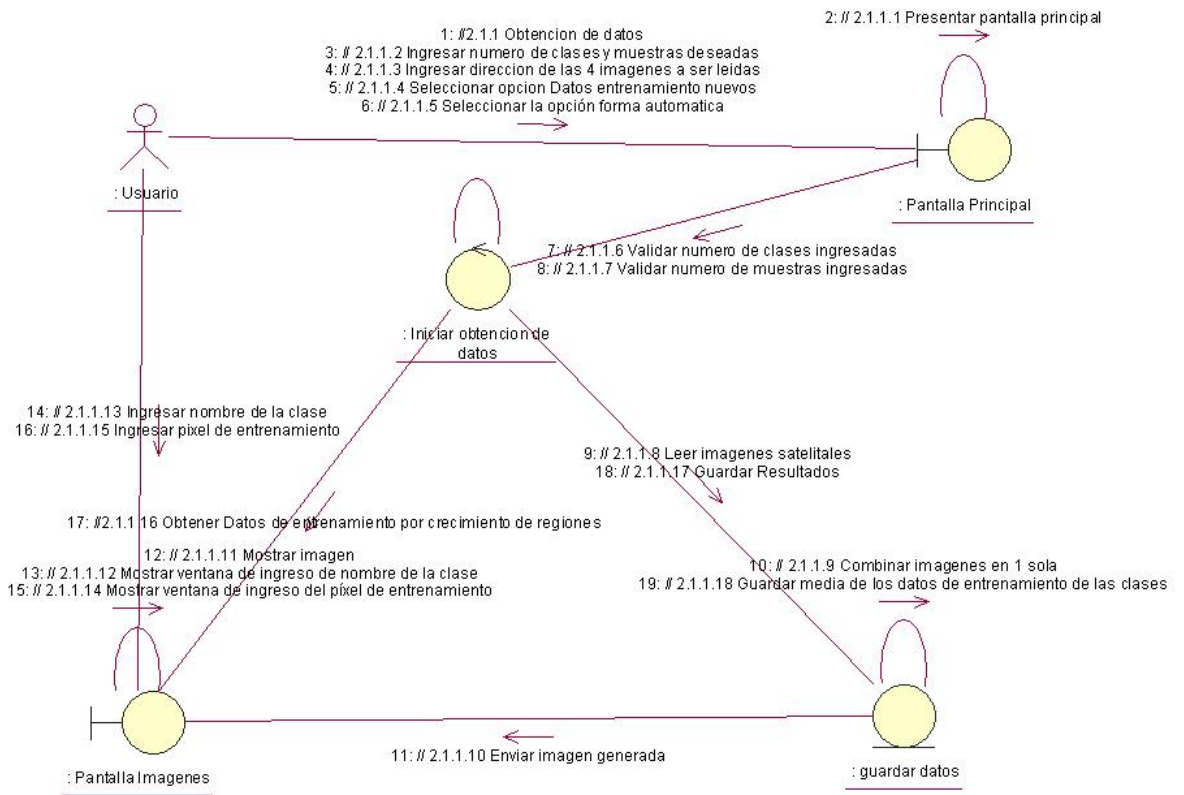


Figura 3.17: Obtención de datos de entrenamiento nuevos en forma automatica.

## 5.- Obtención de Datos de Entrenamiento Almacenados

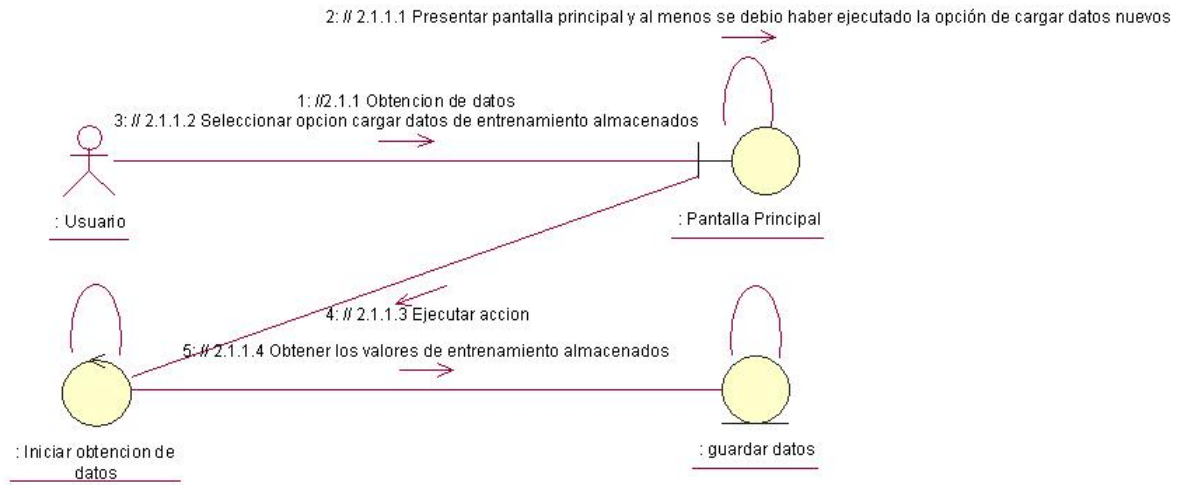


Figura 3.18: Obtención de datos de entrenamiento almacenados.

## Diagramas de Colaboración para el caso de uso: Clasificación

### 1.- Clasificador de Distancia Mínima a la Media

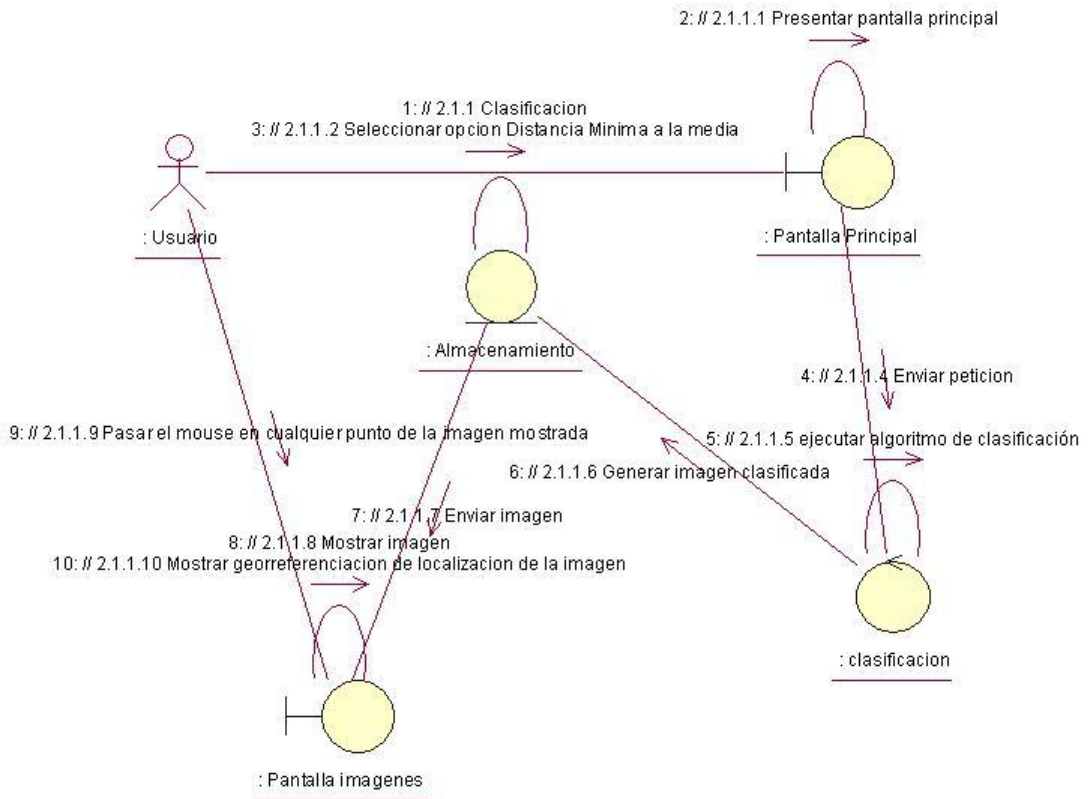


Figura 3.19: Clasificador de Distancia Mínima a la Media.

## 2.- Clasificador de Distancia Mahalanobis

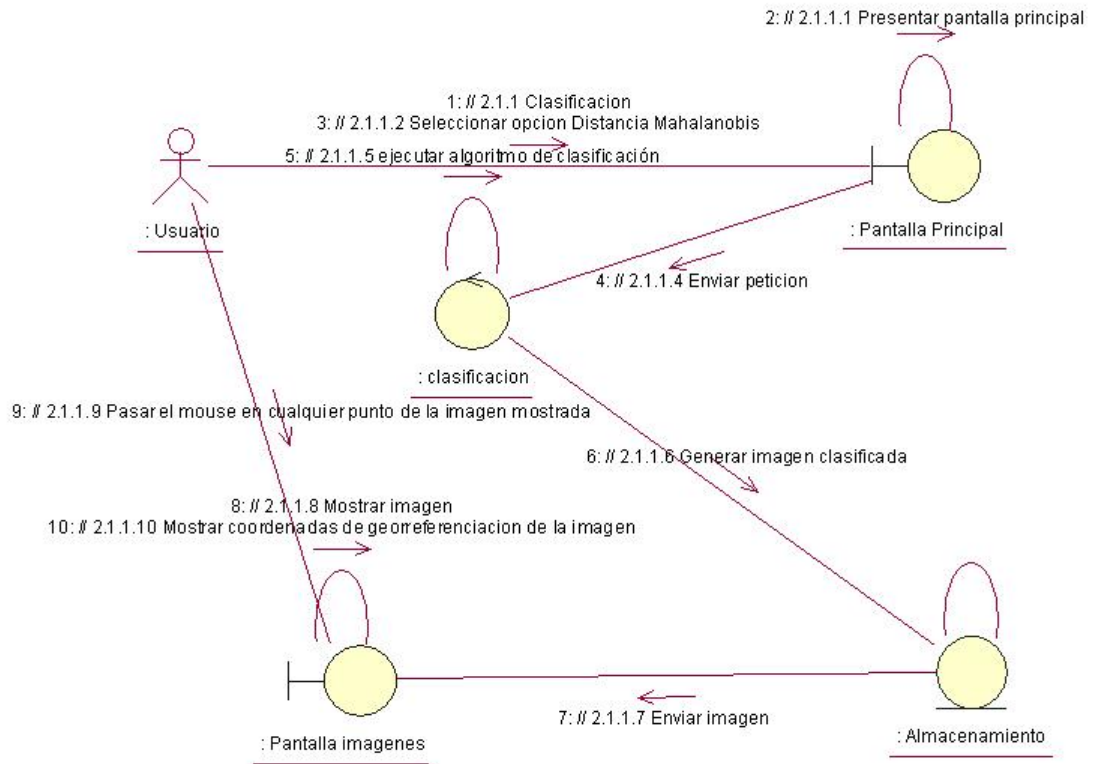


Figura 3.20: Clasificador de Distancia Mahalanobis.

### 3.- Clasificador de Máxima Probabilidad

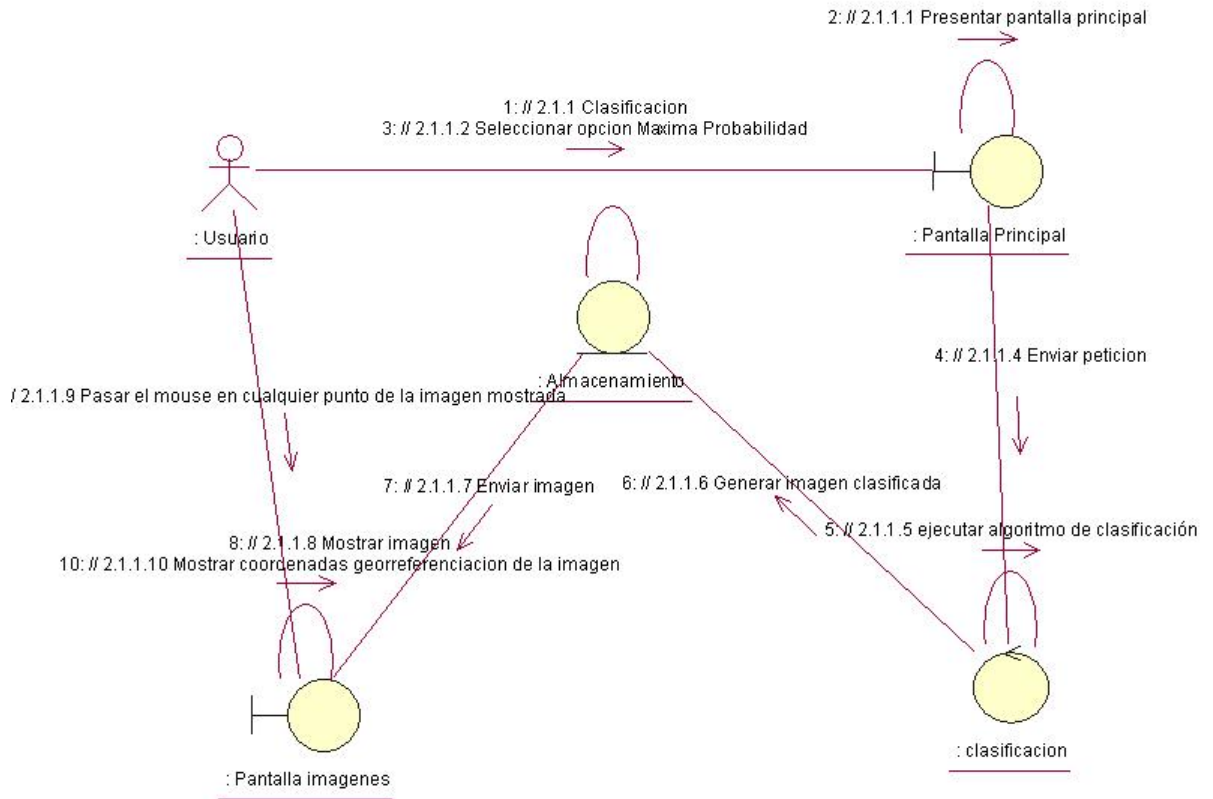


Figura 3.21: Clasificador de Maxima Probabilidad.

#### 4.- Clasificador ISODATA

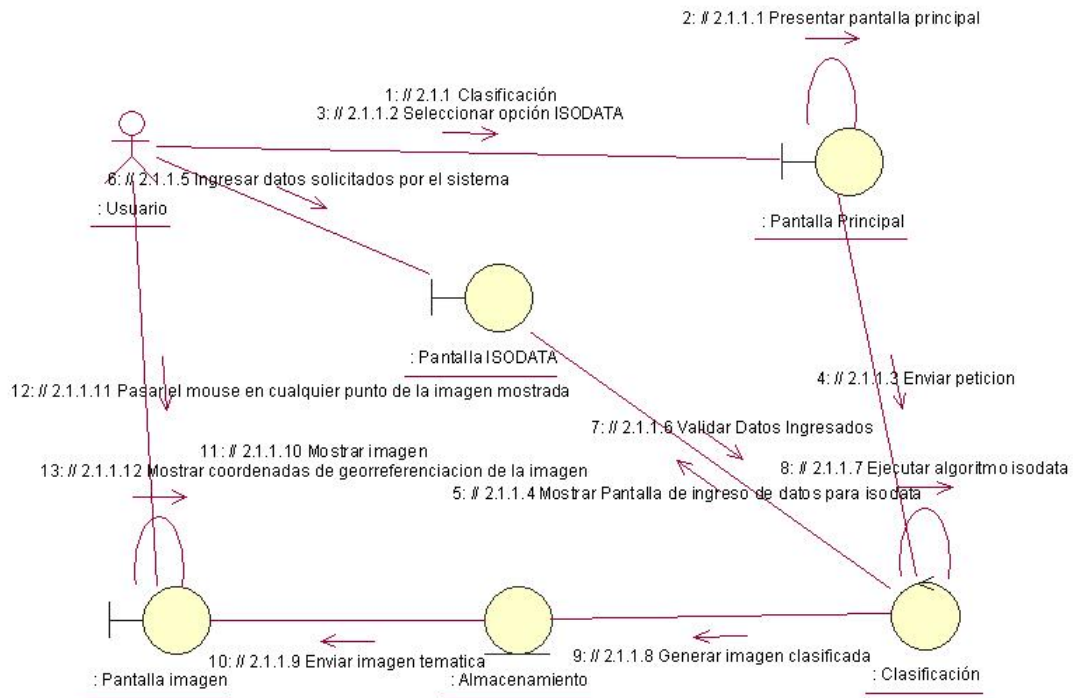


Figura 3.22: Clasificador ISODATA.

## 5.- Error en Datos Ingresados del Clasificador ISODATA

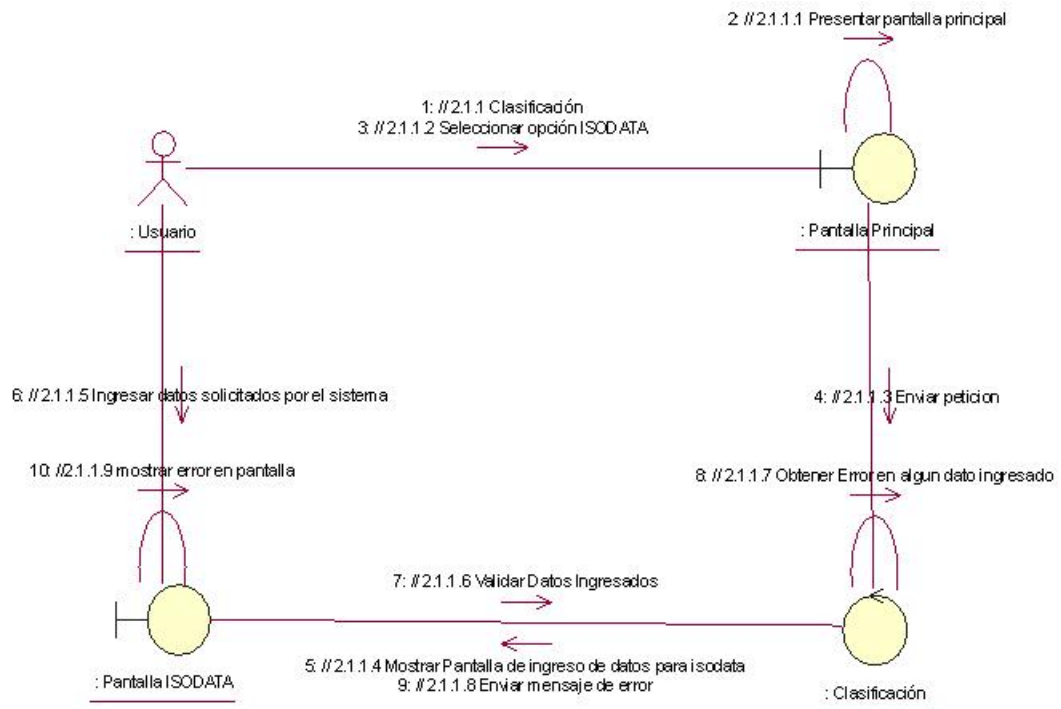


Figura 3.23: Error al ingresar datos ISODATA.

## Diagramas de Colaboración para el caso de uso: Validación de Resultados

### 1.- Matriz de Confusión

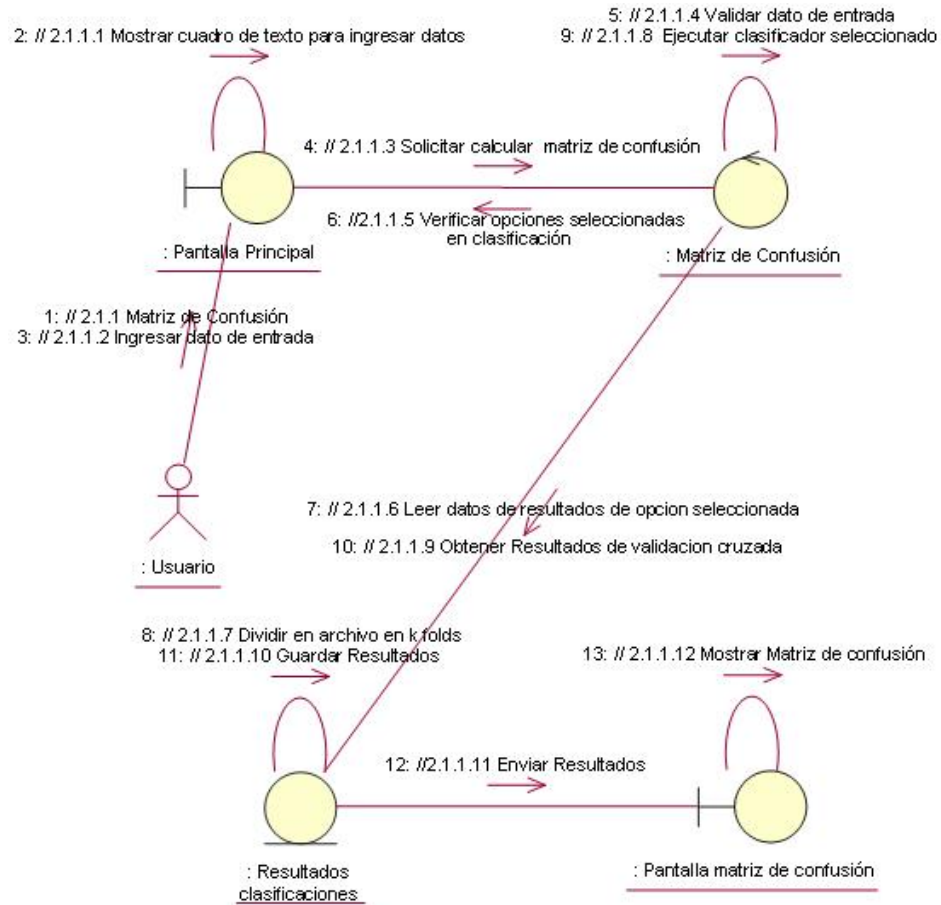


Figura 3.24: Matriz de confusión.

## 2.- Error en Obtención de datos de la Matriz de Confusión

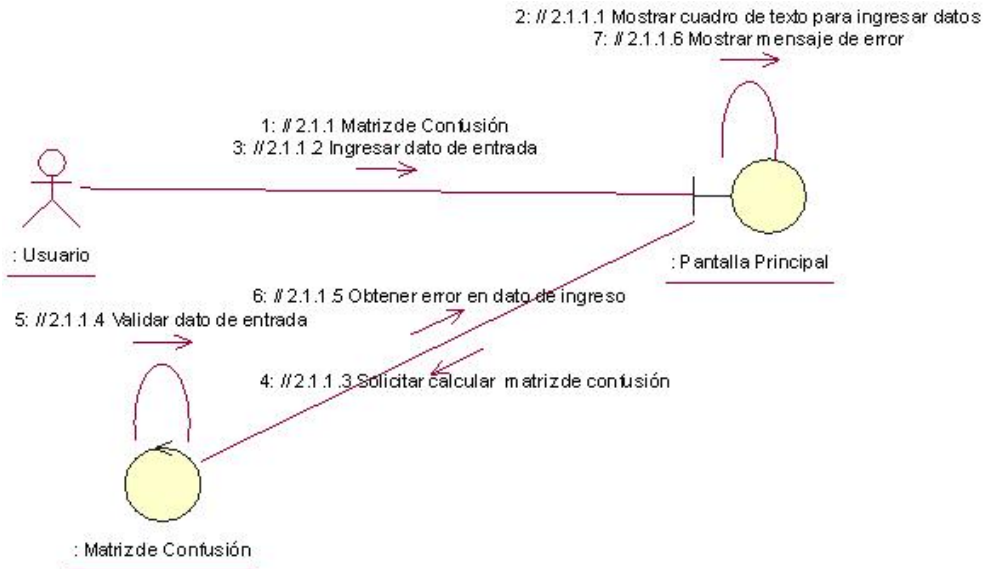


Figura 3.25: Error en obtención de datos de la matriz de confusión.

# Capítulo 4

## Desarrollo del Sistema

Para el desarrollo e implementación del sistema se utilizó el lenguaje de programación C++. Para el desarrollo de la interfaz gráfica se optó por el entorno de desarrollo integrado (IDE) de Visual C++, el cual contiene una librería de clases MFC (Microsoft Foundation Classes) que facilitan la programación gráfica sin tener que utilizar directamente el API de Windows, ya que agrupan las librerías de Windows en clases de C++. Una ventaja de Visual C++ es que puede agregar librerías de otras herramientas de desarrollo y manipularlas como propias. Las librerías extras incluidas en VC++ son las librerías del software Halcón [19], las cuales son una poderosa herramienta para el procesamiento y análisis de imágenes.

Con base en el análisis del sistema del capítulo anterior se han identificado las clases necesarias para la implementación y desarrollo del mismo. Las clases identificadas se dividen principalmente en 3 estereotipos, los cuales son:

Clase Límite (Boundary). Esta clase modela la comunicación entre lo que rodea al sistema y su funcionamiento interno, es decir son principalmente pantallas o interfaces en las que el usuario interactúa de forma directa con el sistema.

Clase Control (Control). Esta clase modela comportamiento de control o coordinación del flujo de eventos asociados a uno o más casos de uso. Esta clase sirve como intermedio entre las clases de límite y las de entidad. Su principal papel es coordinar el comportamiento del caso de uso.

Clase Entidad (Entity). Esta clase corresponde a las abstracciones principales del modelo conceptual y modela la estructura y comportamiento asociado a una clase que generalmente es de larga duración, es decir el principal papel es almacenar y adminis-

trar la información del sistema.

Para mostrar las clases del sistema identificadas, así también como sus interrelaciones es necesario el diseño del diagrama de clases. Este diagrama es utilizado para mostrar como puede ser construido el sistema y qué puede hacer. El diagrama de clases además muestra la interacción que existe entre las clases para la realización de los objetivos del sistema. El diagrama de clases del sistema se muestra en la Fig. 4.1 y la descripción de cada clase se muestra en el Apéndice B.

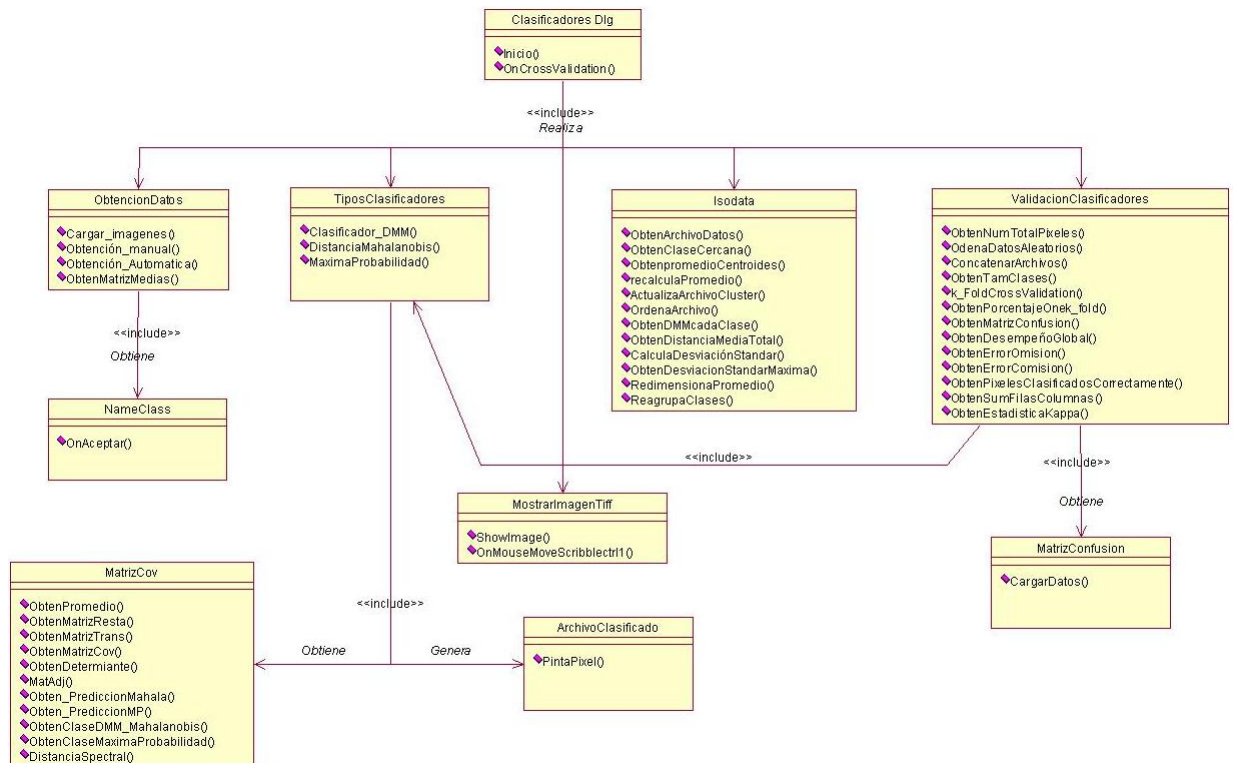


Figura 4.1: Diagrama de Clases del sistema

A continuación mostramos la interfaz del sistema y su funcionalidad. En la Fig. 4.2 se presenta la pantalla principal en la cual se solicita al usuario que ingrese los datos necesarios para poder realizar la clasificación. En esta pantalla el usuario ingresa los datos como son el número de clases deseadas para poder realizar la clasificación, el número de datos de entrenamiento que se tomarán por cada clase a identificar, la dirección de las 4 bandas de la imagen satelital, el tipo de elección de información, la forma en que se van a tomar los datos de entrenamiento y el clasificador a ejecutar.

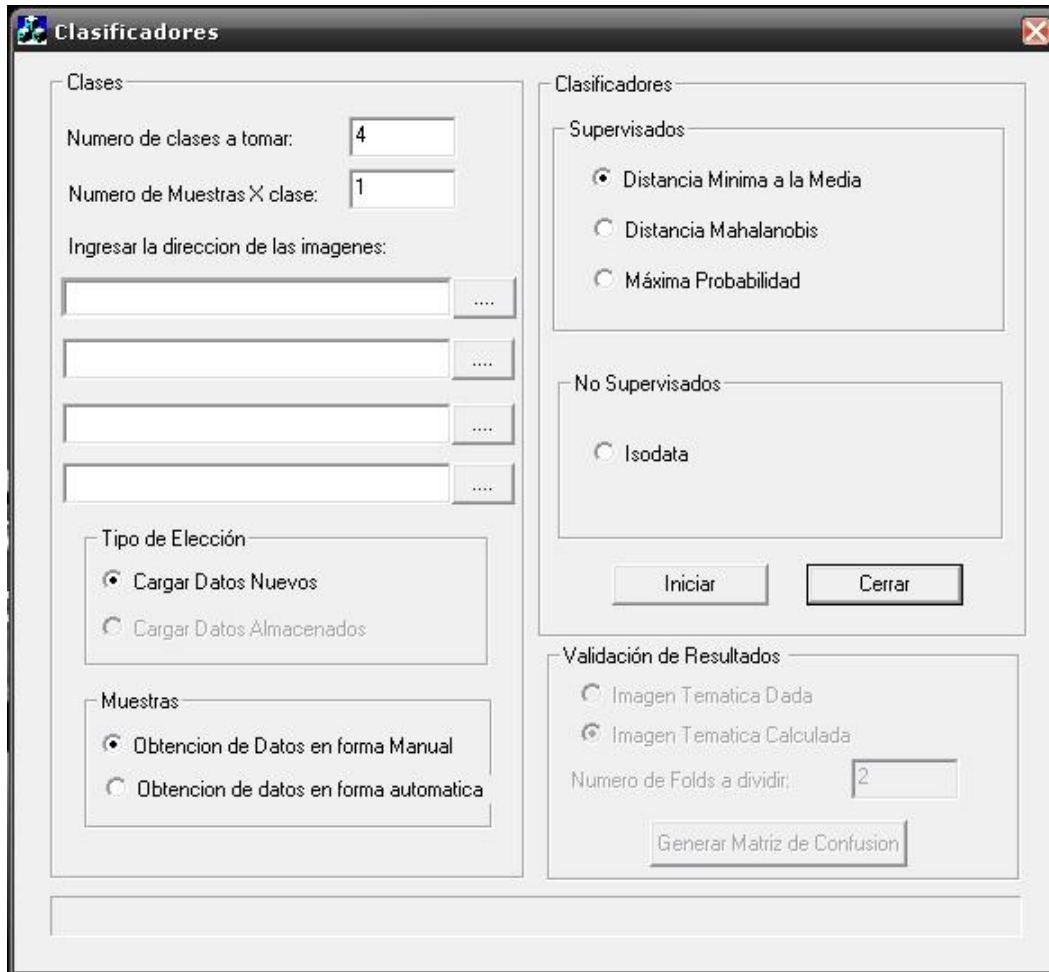


Figura 4.2: Pantalla de inicio del sistema.

En un principio algunas opciones se encuentran desactivadas porque necesitan que primero se ejecute una clasificación para poder activarse y ejecutar sus funciones respectivas.

Si el número de clases o el número de datos de entrenamiento elegido por el usuario no se encuentra dentro del rango permitido, se muestra un mensaje de error y se solicita que ingrese un valor entre ese rango (ver Fig. 4.3).

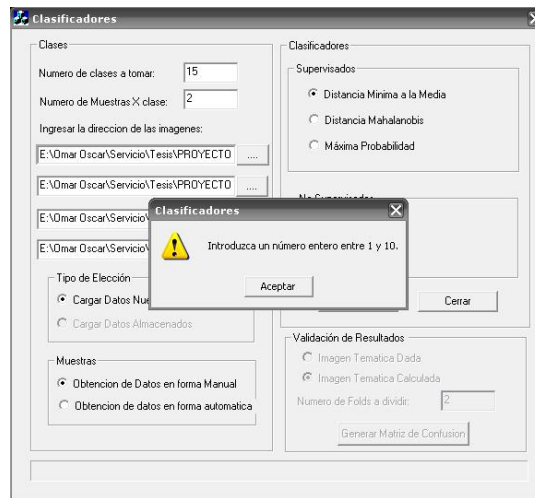


Figura 4.3: Mensaje de error cuando un valor no se encuentra entre el rango [1,10].

Para la lectura de las imágenes se muestra un explorador de archivos en donde el usuario busca la imagen a ingresar y la selecciona. Automáticamente se obtiene la ruta de la imagen seleccionada como se muestra en la Fig. 4.4.

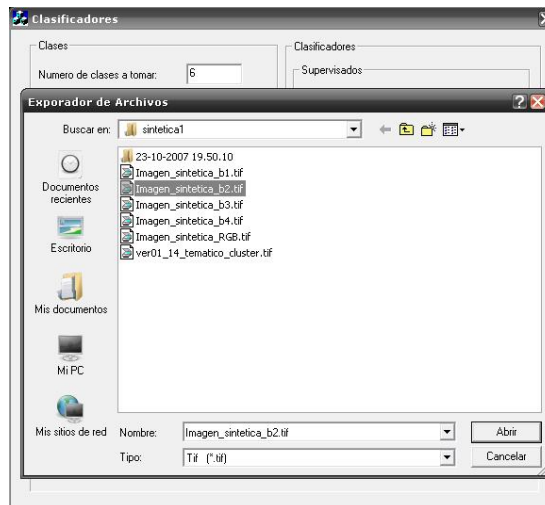


Figura 4.4: Explorador de archivos.

Para realizar la clasificación es necesario haber cargado las 4 bandas de la imagen SPOT, si un campo en donde se ingresa la ruta esta vacío, se envía un mensaje de error (ver Fig. 4.5).

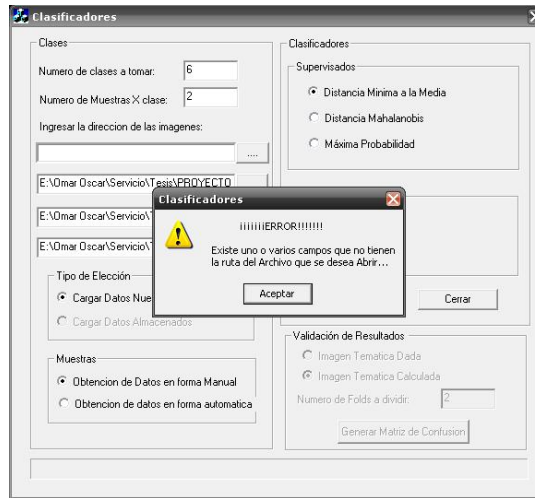


Figura 4.5: Mensaje de error al ingresar las rutas de los archivos.

Una vez ingresado el número de clases, datos de entrenamiento y los archivos de imagen a ser leídos, es posible seleccionar un clasificador y ejecutarlo con el botón de Iniciar (ver Fig. 4.6).

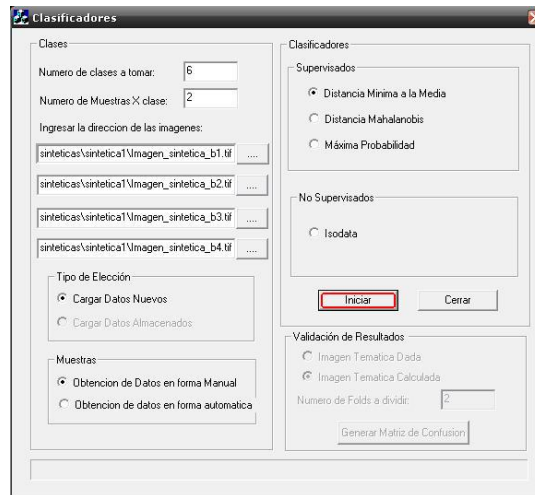


Figura 4.6: Inicio del sistema.

Al iniciarse el clasificador primero se muestra una imagen generada de la composición de las imágenes leídas y se solicita al usuario que ingrese el nombre de cada clase para después obtener los datos de entrenamiento (ver Fig. 4.7).

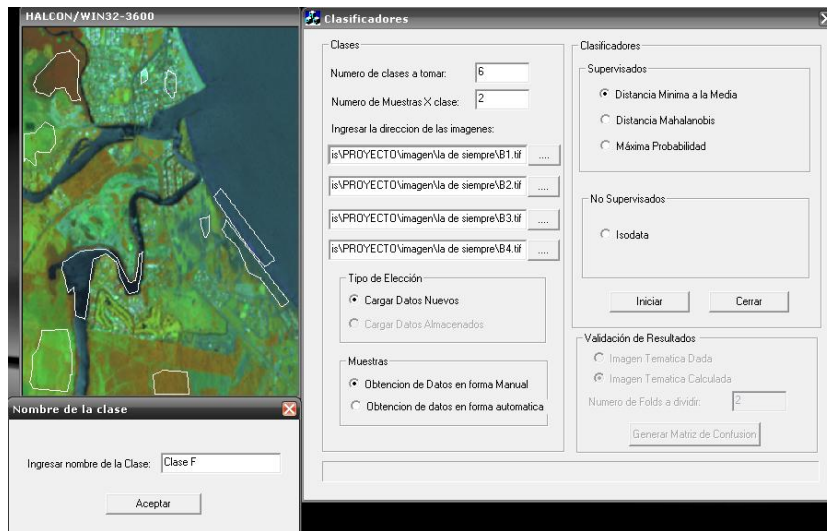


Figura 4.7: Mensaje que indica al usuario ingresar el nombre de la clase.

El usuario ingresa el nombre de la clase y el sistema solicita al usuario que ingrese la muestra de entrenamiento de la imagen mostrada en pantalla como se muestra en la Fig. 4.8

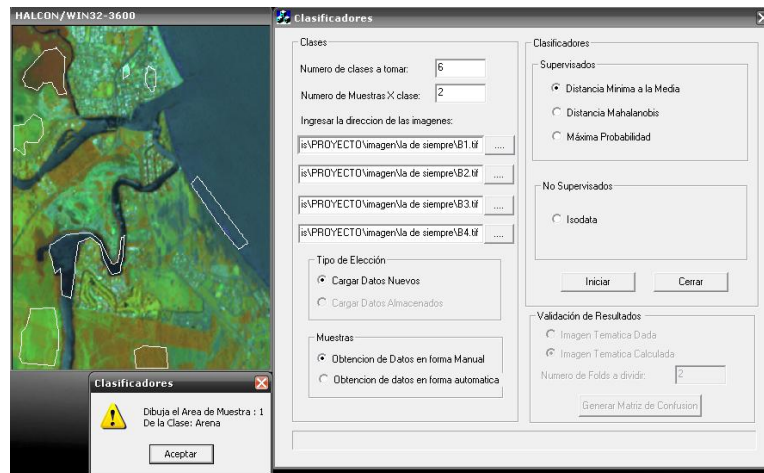


Figura 4.8: Mensaje que indica al usuario ingresar los datos de entrenamiento.

Una vez terminada la obtención de los datos de entrenamiento el sistema ejecuta el clasificador seleccionado y comienza la clasificación (ver Fig. 4.9).

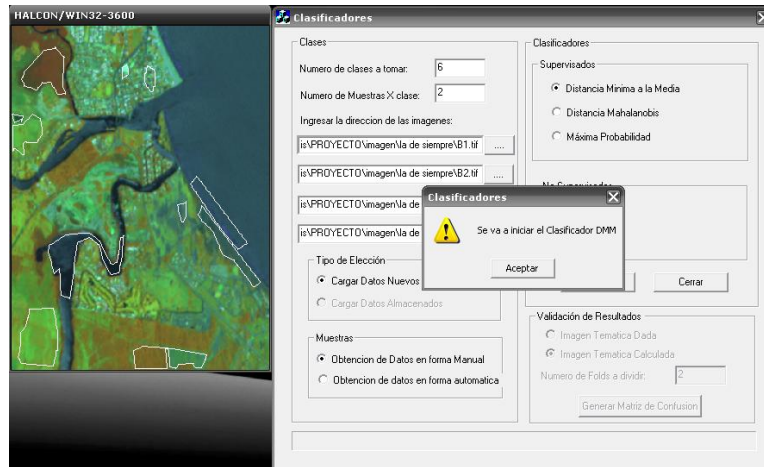


Figura 4.9: Mensaje que indica al usuario que se va a iniciar la clasificación.

Cuando se termina de ejecutar la clasificación, el sistema muestra una nueva ventana en donde se presenta al usuario la imagen temática obtenida (ver Fig. 4.10).

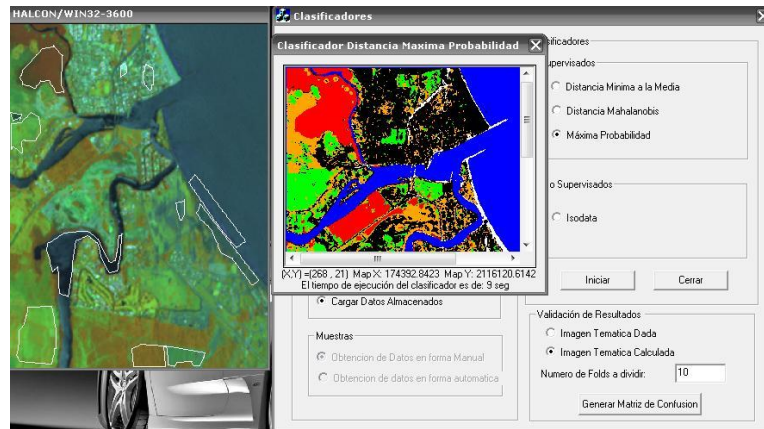


Figura 4.10: Se muestra imagen temática obtenida.

Además se activan opciones como la de validación de resultados del clasificador y también la de obtener los datos almacenados del sistema. La nueva imagen muestra datos de georreferenciación al pasar el puntero del ratón sobre la imagen.

Si se selecciona que se obtenga la validación de resultados del clasificador ejecutado el sistema muestra en una nueva ventana la información referente al mismo, como es el desempeño global, la estadística Kappa, el error por comisión y omisión (ver Fig.

4.11).

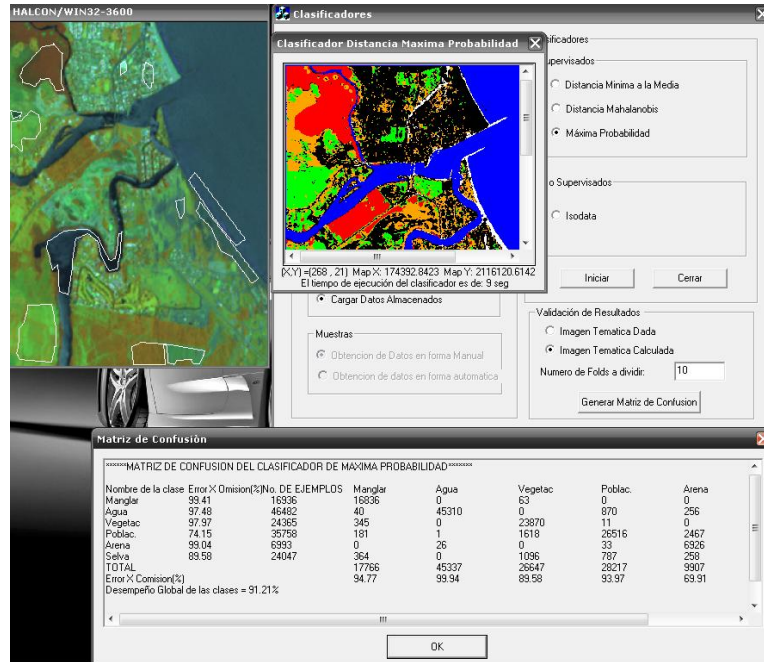


Figura 4.11: Se muestra la matriz de confusión.

Con la opción de cargar datos almacenados del sistema, se pueden hacer comparaciones entre los diferentes clasificadores implementados y verificar cuál de los clasificadores ejecutados tiene un mejor desempeño y por lo tanto una clasificación más exacta, ya que los mismos datos para realizar la primera clasificación se pueden utilizar para los demás clasificadores.

Estas comparaciones pueden ser de forma visual, observando las imágenes temáticas generadas ó pueden ser de forma cuantitativa, basándose en la matriz de confusión generada para cada clasificador.

Finalmente, en la Fig. 4.12 se muestran las imágenes temáticas de los clasificadores supervisados ejecutados con los mismos datos de entrenamiento.

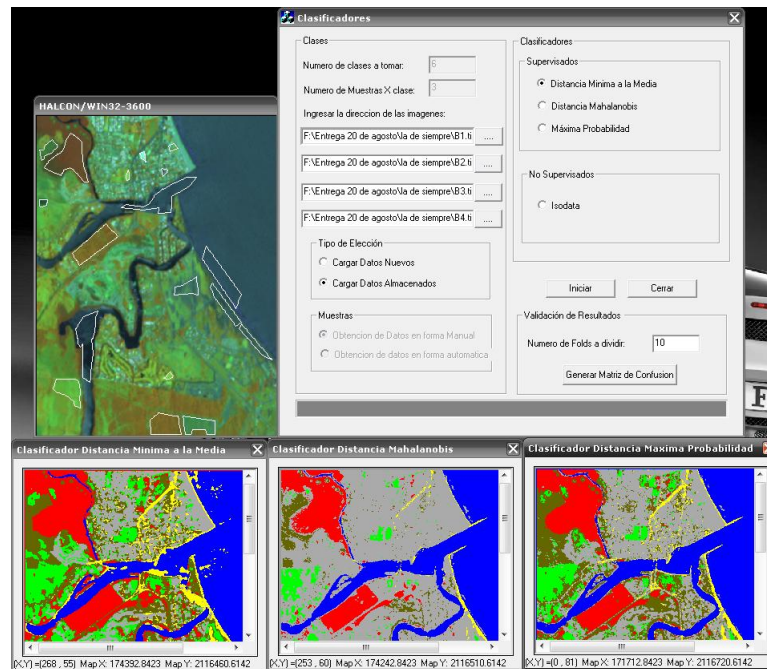


Figura 4.12: Ejecución de todos los clasificadores supervisados con los mismos datos de entrenamiento.

Los clasificadores ejecutados anteriormente son supervisados, pero si el usuario selecciona el clasificador ISODATA que es un clasificador no supervisado, el sistema muestra una nueva ventana en donde se le solicita al usuario que ingrese los datos necesarios para poder realizar la clasificación seleccionada (ver Fig. 4.13)

Los datos que el usuario ingrese son validados de acuerdo a un rango especificado por defecto para cada uno de ellos. Si al realizarse la validación de los datos existe alguno que se encuentra fuera de ese rango, se muestra un mensaje de error. El clasificador no inicia hasta que los datos ingresados sean los correctos.

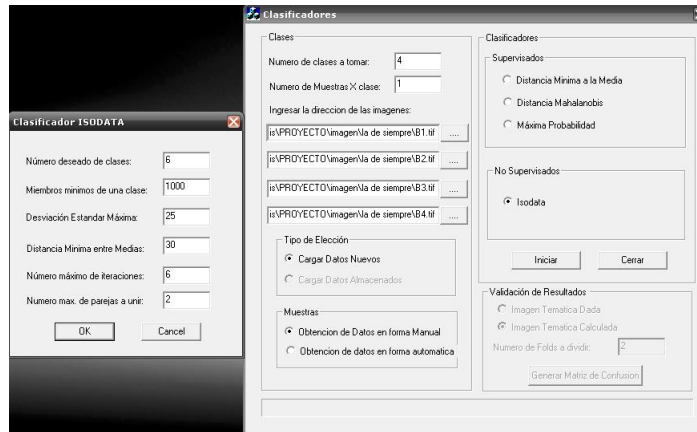


Figura 4.13: Solicitud de datos en el clasificador ISODATA.

Cuando se termina de ejecutar el clasificador ISODATA, se muestran dos imágenes: una en donde se puede observar las clases elegidas aleatoriamente y otra en donde se muestra la imagen clasificada. En esta imagen también se puede obtener información de georreferenciación de la imagen temática (ver Fig. 4.14).

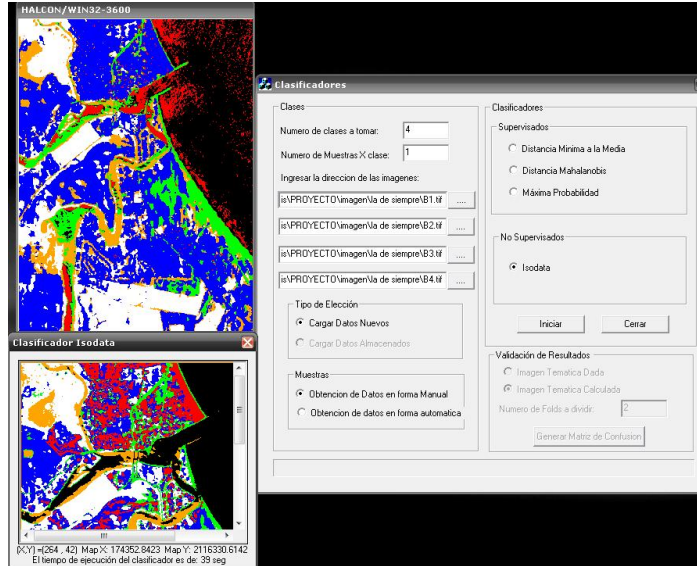


Figura 4.14: Clasificador ISODATA.

# Capítulo 5

## Resultados Experimentales

El presente capítulo muestra los resultados obtenidos con el sistema desarrollado para clasificar diferentes coberturas terrestres. Así también como el tiempo de ejecución que tarda cada clasificador en realizar la segmentación de la imagen temática.

Es importante mencionar que el tiempo que tarda cada clasificador en realizar la obtención de la imagen temática depende del equipo de cómputo utilizado. El sistema fue ejecutado en un equipo de cómputo con las siguientes características:

- **Procesador:** AMD ATLHON XP 3000+
- **Memoria RAM:** 768 MB
- **Disco Duro:** 250 GB

### 5.1. Origen de los datos experimentales

El sistema fue probado con dos tipos de imágenes:

- **Imágenes Reales**
- **Imágenes Sintéticas**

Las imágenes reales son imágenes satelitales, las cuales son proporcionadas por el satélite SPOT-5 compuestas por 4 bandas espectrales. Las 4 bandas son las ingresadas al sistema para realizar la clasificación de coberturas (ver Fig. 5.1).

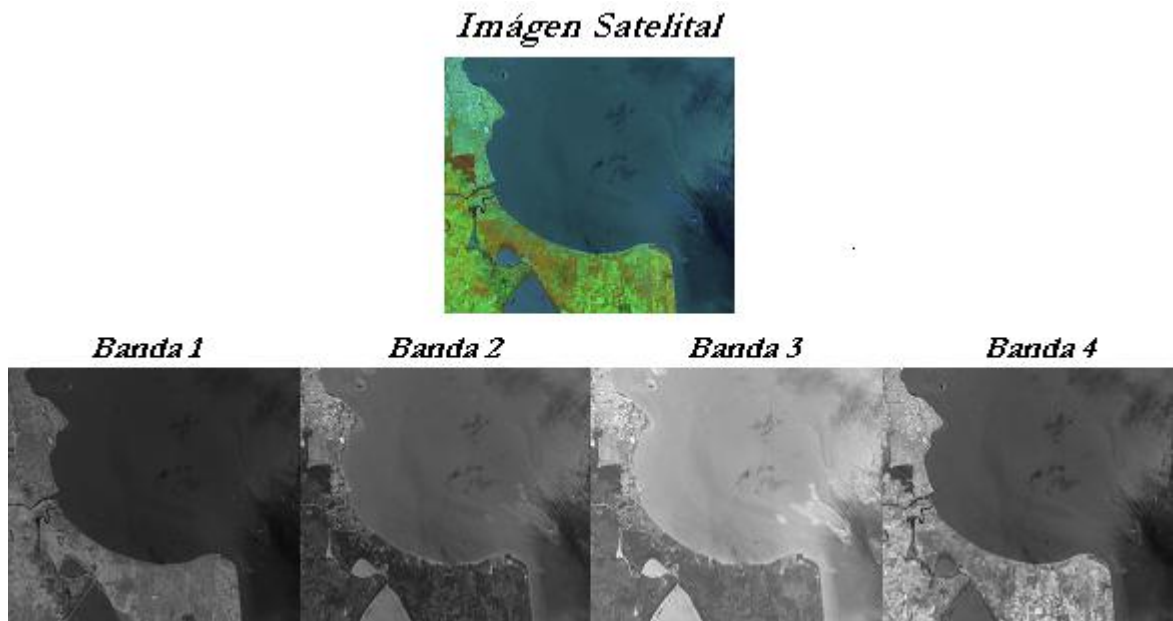


Figura 5.1: Imágen Satelital (corte de tamaño 2194 x 1931).

El segundo tipo de imágenes utilizadas son Imágenes Multiespectrales Sintéticas (IMS), las cuales son generadas a partir de un número conocido de valores de grises para cada clase. Estas IMS fueron propuestas en la tesis de maestría [20] y fueron utilizadas para evaluar la clasificación de coberturas.

La inicialización de la imagen sintética, a diferencia de la tesis de maestría la estamos realizando a partir del método ISODATA. El procedimiento para generar la IMS es la siguiente:

1. A partir de una imagen satelital se realiza primero una clasificación no supervisada con ISODATA.
2. Después, a todos los píxeles de las regiones generadas en el paso anterior, se les asigna un nuevo valor de gris, de acuerdo a la clase identificada por ISODATA. Estos valores de gris corresponden a valores verdaderos de cada clase.
3. El procedimiento del paso anterior se realiza para cada banda.

Un ejemplo de una IMS se muestra en la Fig. 5.2.

Debido a que no tenemos un mapa temático real, el desempeño de los clasificadores se realiza de dos maneras:

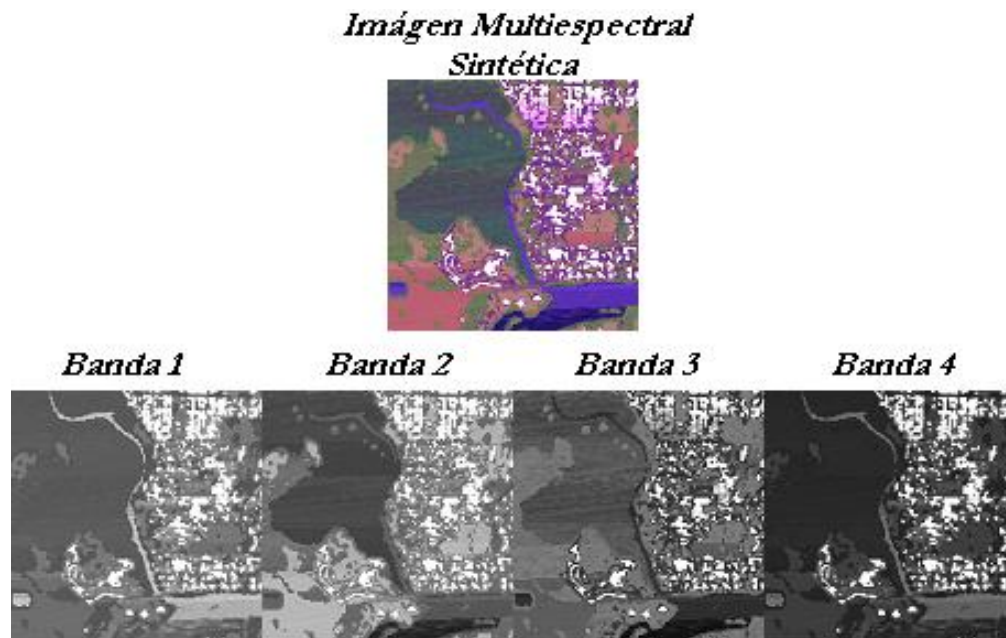


Figura 5.2: Imágen Multiespectral Sintética.

1. Utilizando validación cruzada: en donde se obtienen datos de entrenamiento y se clasifica la imagen. Esta clasificación proporciona el mapa temático de referencia. Después se realiza nuevamente la clasificación aplicando validación cruzada tomando como referencia el mapa generado anteriormente y se evalúa el clasificador.
2. Utilizando una imagen de referencia: en donde se obtienen datos de entrenamiento y se clasifica la imagen. Esta clasificación nos proporciona un mapa temático que será al que compararemos con la imagen de referencia obtenida por ISODATA. Después tomando la imagen de referencia se compara píxel por píxel con el mapa generado y se evalúa el clasificador.

En ambos casos se genera una matriz de confusión con los resultados obtenidos.

## 5.2. Resultados experimentales obtenidos

Los resultados presentados en esta sección fueron comparados visualmente contra los resultados obtenidos por el software comercial ERDAS.

A partir de esta comparación observamos que las imágenes temáticas obtenidas por el sistema implementado y por ERDAS fueron muy similares. Las pocas diferencias existentes se deben principalmente a que los datos de entrenamiento no fueron exactamente los mismos en ambas herramientas, provocando que las imágenes temáticas generadas por los clasificadores no fueran similares totalmente.

En esta tesis sólo se presentan las comparaciones hechas por el sistema.

Las clases establecidas se presentan en la Fig. 5.3.



Figura 5.3: Esquema de Clasificación

El sistema proporciona dos formas para obtener los datos de entrenamiento:

- La primera es manualmente, donde el usuario selecciona polígonos que corresponden a regiones representativas de cada clase.
- La segunda es utilizando crecimiento de regiones, donde el usuario sólo selecciona un píxel en la imagen.

En los experimentos con imágenes reales empleamos la selección de datos de entrenamiento de forma manual. Para los experimentos con imágenes sintéticas empleamos la obtención de datos de entrenamiento de manera automática (crecimiento de regiones).

A continuación se muestran los resultados de clasificación obtenidos con diferentes imágenes satelitales y sintéticas, así también como su respectiva matriz de confusión y tiempo de ejecución.

La Fig. 5.4 muestra las imágenes temáticas obtenidas con los 3 clasificadores supervisados para un corte de 2194 x 1931 píxeles.

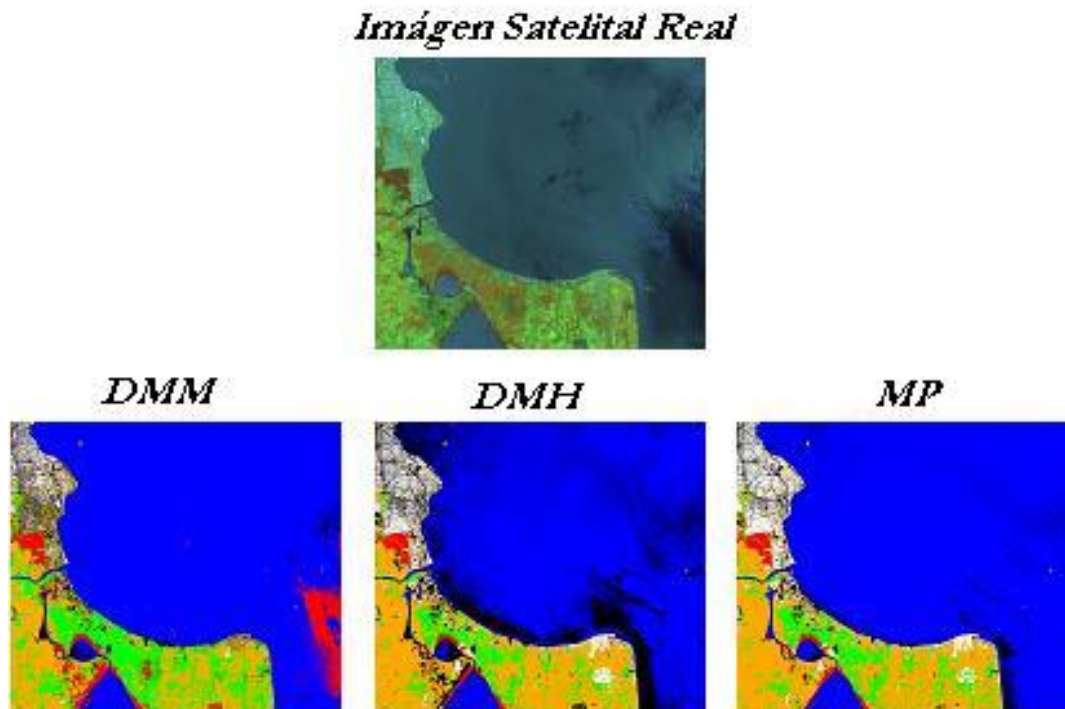


Figura 5.4: Mapas temáticos generados por cada clasificador para la imagen de 2194 x 1931.

En las tablas 5.1, 5.2 y 5.3 se muestra la matriz de confusión para cada clasificador así también como su estadística kappa, error por comisión, error por omisión y desempeño global.

En la tabla 5.1 observamos la matriz de confusión generada por validación cruzada del clasificador Distancia Mínima a la Media (DMM).

Nombre de la clase	Error por Comisión	No. de ejemplos	Número de muestras en la clase					
			Manglar	Agua	Veget.	Carret	Zonas U	Suelo D
Manglar	75.66	226750	171566	222	54962	0	0	0
Agua	93.77	2819338	155207	2643657	465	19874	0	135
Veget.	99.80	411283	0	0	410459	89	0	735
Carret	99.95	146221	0	11	0	146153	47	10
Zonas U	85.38	100761	0	0	0	14701	86026	34
Suelo D	92.68	532261	0	0	25791	13152	9	493309
TOTAL			326773	2643890	491677	193969	86082	494223
Error por Omisión (%)			52.50	99.99	83.48	75.35	99.93	99.82
Desempeño Global: 93.26 %				Estadística Kappa: 87.81 %				

Tabla 5.1: Resultados del clasificador DMM para imagen de 2194 x 1931.

En la tabla 5.2 se muestra la matriz de confusión generada por validación cruzada del clasificador Distancia Mahalanobis (DMH).

Nombre de la clase	Error por Comisión	No. de ejemplos	Número de muestras en la clase					
			Manglar	Agua	Veget.	Carret	Zonas U	Suelo D
Manglar	96.24	47403	45623	0	23	9	1052	696
Agua	29.58	2360327	0	698073	0	1662254	0	0
Veget.	87.02	186002	539	0	161858	0	3838	19767
Carret	87.71	753561	370	0	0	660986	83272	8933
Zonas U	99.95	209286	1	0	0	0	209171	114
Suelo D	90.27	680035	2264	0	1186	465	62238	613882
TOTAL			48797	698073	163067	2323714	359571	643392
Error por Omisión (%)			93.50	100.00	99.26	28.45	58.17	95.41
Desempeño Global: 56.40 %				Estadística Kappa: 44.13 %				

Tabla 5.2: Resultados del clasificador DMH para imagen de 2194 x 1931.

La tabla 5.3 muestra la matriz de confusión generada por validación cruzada del clasificador Máxima Probabilidad (MP).

Nombre de la clase	Error por Comisión	No. de ejemplos	Número de muestras en la clase					
			Manglar	Agua	Veget.	Carret	Zonas U	Suelo D
Manglar	98.49	51920	51137	0	0	257	420	106
Agua	95.87	2816988	0	2700755	0	116233	0	0
Veget.	97.97	223019	2109	0	218501	0	1866	543
Carret	79.75	234974	323	27572	0	187401	17080	2598
Zonas U	98.46	216605	28	0	0	1233	213267	2077
Suelo D	92.66	693108	4902	0	33402	4227	8361	642216
TOTAL			58499	2728327	251903	309351	240994	647540
Error por Omisión (%)			87.42	98.99	86.74	60.58	88.49	99.18
Desempeño Global: 94.73 %				Estadística Kappa: 90.17 %				

Tabla 5.3: Resultados del clasificador MP para imagen de 2194 x 1931.

A continuación se muestra en la tabla 5.4 un comparativo entre los diferentes clasificadores ejecutados para la imagen satelital anterior.

Clasificadores \ Resultados	DMM	DMH	MP
Desempeño Global:	93.26 %	56.40 %	94.73 %
Estadística Kappa:	87.81 %	44.13 %	90.17 %
Tiempo de Ejecución:	109 segundos	209 segundos	211 segundos

Tabla 5.4: Comparativo entre clasificadores de imagen satelital de 2194 x 1931.

En la tabla anterior podemos observar que el clasificador que obtuvo un mejor desempeño fue el clasificador de Máxima probabilidad. Los resultados proporcionados por la matriz de confusión muestran que dicho clasificador superó en más del 1 % al clasificador de DMM y un 40 % al clasificador DMH. Podemos observar que DMH confundió la clase agua con la clase carretera y la clase carretera con zonas urbanas y suelo descubierto. Podemos observar que, en tiempo de ejecución fue más rápido el clasificador DMM y el más tardado fue el MP, esto puede suceder debido a que dicho clasificador utiliza la matriz de covarianza teniendo que hacer más cálculos.

En la Fig. 5.5 se muestran los resultados de clasificación obtenidos con los clasificadores supervisados para una imagen satelital de 150 x 150 píxeles. En este experimento sólo se identificaron 4 clases.

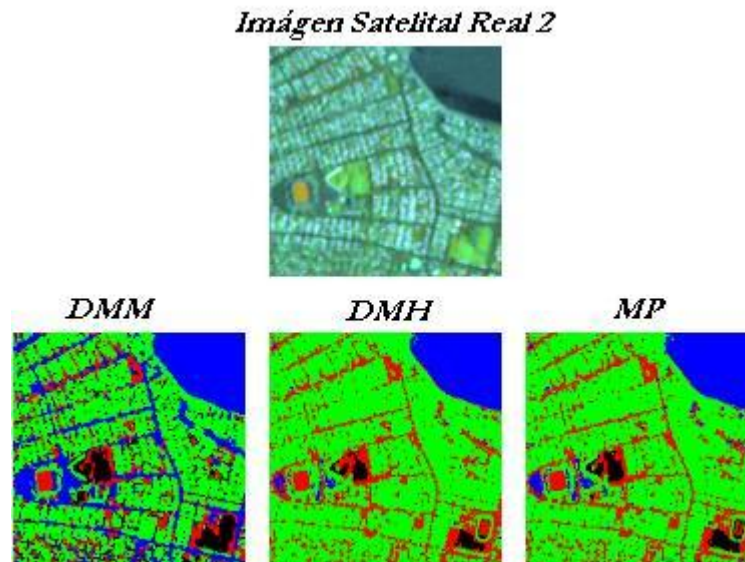


Figura 5.5: Mapas temáticos generados por cada clasificador para imagen satelital 150 x 150.

En la tabla 5.5 se muestran los datos de la matriz de confusión del clasificador Distancia Mínima a la Media (DMM).

Nombre de la clase	Error por Comisión	No. de ejemplos	Número de muestras en la clase			
			Carreteras	Agua	Zonas U.	Suelo D.
Carreteras	89.83	2666	2395	0	0	271
Agua	81.69	5757	941	4703	0	113
Zonas U.	83.07	11905	3	129	9889	1884
Suelo D.	97.47	2172	53	0	2	2117
TOTAL			3392	4832	9891	4385
Error por Omisión (%)			70.61	97.33	99.98	48.28
Desempeño Global: 84.91 %			Estadística Kappa: 77.67			

Tabla 5.5: Resultados del clasificador DMM para imagen satelital 150 x 150.

En la tabla 5.6 observamos los datos generados por la matriz de confusión para el

clasificador Distancia Mahalanobis (DMH).

Nombre de la clase	Error por Comisión	No. de ejemplos	Número de muestras en la clase			
			Carreteras	Agua	Zonas U.	Suelo D.
Carreteras	96.79	4044	3914	0	40	90
Agua	90.33	2400	120	2168	112	0
Zonas U.	97.78	15704	347	0	15356	1
Suelo D.	97.44	352	7	0	2	343
TOTAL			4388	2168	15510	434
Error por Omisión (%)			89.20	100.00	99.01	79.03
Desempeño Global: 96.80 %			Estadística Kappa: 93.25			

Tabla 5.6: Resultados del clasificador DMH para imagen satelital 150 x 150.

En la tabla 5.7 se muestran los datos de la matriz de confusión para el clasificador de Máxima Probabilidad (MP).

Nombre de la clase	Error por Comisión	No. de ejemplos	Número de muestras en la clase			
			Carreteras	Agua	Zonas U.	Suelo D.
Carreteras	95.22	4080	3885	1	2	192
Agua	94.92	2619	100	2486	33	0
Zonas U.	92.70	15292	1042	47	14175	28
Suelo D.	99.21	509	4	0	0	505
TOTAL			5031	2534	14210	725
Error por Omisión (%)			77.22	98.11	99.75	69.66
Desempeño Global: 93.56 %			Estadística Kappa: 87.53			

Tabla 5.7: Resultados del clasificador MP para imagen satelital 150 x 150.

A partir de los resultados anteriores observamos que los clasificadores que obtuvieron mejor desempeño fueron el clasificador de Distancia Mahalanobis y el clasificador de Máxima Probabilidad. Podemos observar que DMM confundió la clase zonas urbanas con suelo descubierto y la clase agua con la clase carreteras, lo que ocasionó su mal desempeño en comparación con los otros dos clasificadores.

El resumen del desempeño global y estadística Kappa para cada clasificador se muestran en la tabla 5.8.

Clasificadores Resultados	DMM	DMH	DMP
Desempeño Global:	84.91 %	96.80 %	93.56 %
Estadística Kappa:	77.67 %	93.25 %	87.53 %
Tiempo de Ejecución:	Menor a 1 seg.	1 segundo	1 segundo

Tabla 5.8: Comparativo entre clasificadores de imagen satelital 150 x 150.

La tabla nos muestra que el clasificador que obtuvo el mejor desempeño fue el clasificador DMH, lo contrario a la clasificación de la imagen anterior. En este caso los niveles de gris para cada clase no eran muy similares por lo que los clasificadores se comportaron de manera adecuada. Como la imagen era pequeña (150 x 150) el tiempo de procesamiento fue rápido y en general los clasificadores tuvieron un comportamiento bueno.

De la misma forma se analizó una imagen multiespectral sintética (IMS) de 150 x 150 en donde se tomaron muestras de entrenamiento empleando crecimiento de regiones. En esta imagen se definieron 6 clases nuevamente. Los resultados obtenidos se muestran en las tablas 5.9, 5.10 y 5.11. Para validar el desempeño de cada clasificador se utilizó el método de validación cruzada. En la Fig. 5.6 se muestran las imágenes temáticas obtenidas de los clasificadores supervisados.

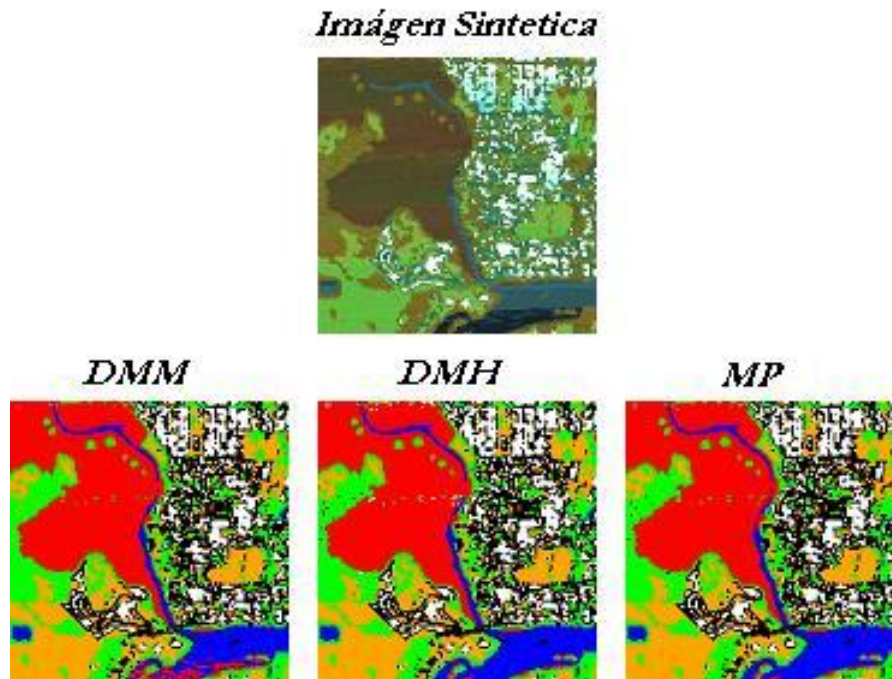


Figura 5.6: Mapas temáticos generados por cada clasificador para IMS 150 x 150.

La tabla 5.9 muestra la matriz de confusión generada por validación cruzada del clasificador de Distancia Mínima a la Media (DMM).

Nombre de la clase	Error por Comisión	No. de ejemplos	Número de muestras en la clase					
			Manglar	Agua	Veget.	Carret	Zonas U	Suelo D
Manglar	99.47	5984	5952	0	14	18	0	0
Agua	86.17	1576	218	1358	0	0	0	0
Veget.	99.81	4190	0	0	4182	0	0	8
Carret	99.10	4316	0	25	6	4277	0	8
Zonas U	99.38	2264	0	1	0	2	2250	11
Suelo D	100.00	4170	0	0	0	0	0	4170
TOTAL			6170	1384	4202	4297	2250	4197
Error por Omisión (%)			96.47	98.12	99.52	99.53	100.00	99.36
Desempeño Global: 98.62 %				Estadística Kappa: 98.29 %				

Tabla 5.9: Resultados del Clasificador DMM para IMS 150 x 150.

La tabla 5.10 muestra la matriz de confusión generada por validación cruzada del

clasificador de Distancia Mahalanobis (DMH).

Nombre de la clase	Error por Comisión	No. de ejemplos	Número de muestras en la clase					
			Manglar	Agua	Veget.	Carret	Zonas U	Suelo D
Manglar	100.00	5680	5680	0	0	0	0	0
Agua	100.00	1793	0	1793	0	0	0	0
Veget.	99.35	4140	8	0	4113	19	0	0
Carret	99.84	4339	0	0	0	4332	0	7
Zonas U	94.95	2436	31	0	0	92	2313	0
Suelo D	99.78	4112	0	0	0	9	0	4103
TOTAL			5719	1793	4113	4452	2313	4110
Error por Omisión (%)			99.32	100.00	100.00	97.30	100.00	99.83
Desempeño Global: 99.26 %				Estadística Kappa: 99.09 %				

Tabla 5.10: Resultados del Clasificador DMH para IMS 150 x 150.

Finalmente, la tabla 5.11 muestra la matriz de confusión generada por validación cruzada del clasificador de Máxima Probabilidad (MP).

Nombre de la clase	Error por Comisión	No. de ejemplos	Número de muestras en la clase					
			Manglar	Agua	Veget.	Carret	Zonas U	Suelo D
Manglar	100.00	5700	5700	0	0	0	0	0
Agua	100.00	1793	0	1793	0	0	0	0
Veget.	99.23	4157	17	0	4125	15	0	0
Carret	100.00	4418	0	0	0	4418	0	0
Zonas U	98.48	2305	0	0	0	35	2270	0
Suelo D	99.73	4127	0	0	0	11	0	4116
TOTAL			5717	1793	4125	4479	2270	4116
Error por Omisión (%)			99.70	100.00	100.00	98.64	100.00	100.00
Desempeño Global: 99.65 %				Estadística Kappa: 99.57 %				

Tabla 5.11: Resultados del Clasificador MP para IMS 150 x 150

Los resultados mostrados en las tablas anteriores muestran que la obtención de datos automática tiene una mayor exactitud, reflejándose en el desempeño de cada clasificador. Además, como la imagen es sintética los píxeles son más similares unos con otros de cada clase.

En la tabla 5.12 se muestra la comparación de resultados de los clasificadores. Se puede observar que todos los clasificadores tienen un desempeño mayor al 98 % debido en gran parte a que los píxeles de la imagen sintética presentan características similares en cada clase.

Resultados \ Clasificadores	DMM	DMH	MP
Desempeño Global:	98.62 %	99.26 %	99.65 %
Estadística Kappa:	98.29 %	99.09 %	99.57 %
Tiempo de Ejecución:	Menor a 1 seg.	1 segundo	1 segundo

Tabla 5.12: Cuadro comparativo entre clasificadores de IMS de 150x 150 utilizando validación cruzada.

El siguiente experimento se hizo para comparar los dos tipos de evaluaciones: validación cruzada e imagen de referencia.

La Fig. 5.7 nos muestra la imagen de referencia y además las imágenes temáticas producto de cada uno de los clasificadores ejecutados.

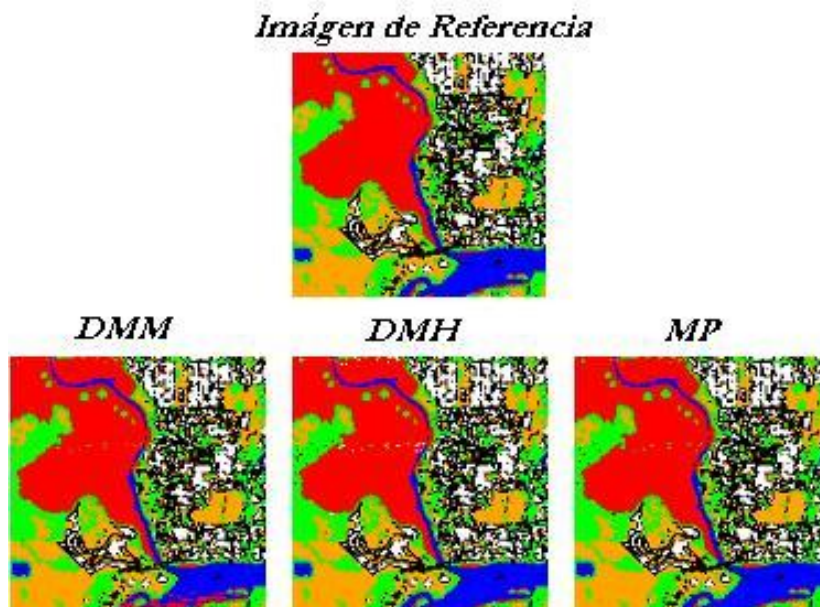


Figura 5.7: Validación de resultados entre clasificadores e imagen de referencia.

La tabla 5.13 muestra los datos de la matriz de confusión del clasificador DMM empleando la evaluación con imagen de referencia.

Nombre de la clase	Error por Comisión	No. de ejemplos	Número de muestras en la clase					
			Manglar	Agua	Veget.	Carret	Zonas U	Suelo D
Manglar	95.52	5984	5716	217	3	48	0	0
Agua	100.00	1576	0	1576	0	0	0	0
Veget.	98.23	4190	26	0	4116	48	0	0
Carret	99.91	4316	0	0	0	4312	4	0
Zonas U	100.00	2264	0	0	0	0	2264	0
Suelo D	98.68	4170	0	0	11	42	2	4115
TOTAL			5742	1793	4130	4450	2270	4115
Error por Omisión (%)			99.55	87.90	99.66	96.90	99.74	100.00
Desempeño Global: 98.22 %				Estadística Kappa: 97.80 %				

Tabla 5.13: Resultados del Clasificador DMM evaluada con imagen de referencia.

La tabla 5.14 muestra los datos de la matriz de confusión del clasificador de Distancia Mahalanobis (DMH) empleando la evaluación con imagen de referencia.

Nombre de la clase	Error por Comisión	No. de ejemplos	Número de muestras en la clase					
			Manglar	Agua	Veget.	Carret	Zonas U	Suelo D
Manglar	100.00	5680	5680	0	0	0	0	0
Agua	100.00	1793	0	1793	0	0	0	0
Veget.	99.30	4140	29	0	4111	0	0	0
Carret	99.54	4339	0	0	13	4319	0	7
Zonas U	93.19	2436	33	0	2	131	2270	0
Suelo D	99.90	4112	0	0	4	0	0	4108
TOTAL			5742	1793	4130	4450	2270	4115
Error por Omisión (%)			98.92	100.00	99.54	97.06	100.00	99.83
Desempeño Global: 99.03 %				Estadística Kappa: 98.80 %				

Tabla 5.14: Resultados del Clasificador DMH evaluada con imagen de referencia.

Finalmente, la tabla 5.15 muestra los datos de la matriz de confusión del clasificador de Máxima Probabilidad (MP).

Nombre de la clase	Error por Comisión	No. de ejemplos	Número de muestras en la clase					
			Manglar	Agua	Veget.	Carret	Zonas U	Suelo D
Manglar	100.00	5700	5700	0	0	0	0	0
Agua	100.00	1793	0	1793	0	0	0	0
Veget.	98.99	4157	42	0	4115	0	0	0
Carret	99.80	4418	0	0	9	4409	0	0
Zonas U	98.48	2305	0	0	0	35	2270	0
Suelo D	99.71	4127	0	0	6	6	0	4115
TOTAL			5742	1793	4130	4450	2270	4115
Error por Omisión (%)			99.27	100.00	99.64	99.08	100.00	100.00
Desempeño Global: 99.56 %				Estadística Kappa: 99.46 %				

Tabla 5.15: Resultados del Clasificador MP evaluada con imagen de referencia.

La tabla 5.16 muestra el resumen de los porcentajes (%) de desempeño para cada clasificador. Como puede observarse, tanto visualmente como cuantitativamente los clasificadores mostraron un buen desempeño.

Clasificadores	DMM	DMH	MP
Resultados			
Desempeño Global:	98.22 %	99.03 %	99.56 %
Estadística Kappa:	97.80 %	98.80 %	99.46 %
Tiempo de Ejecución:	Menor a 1 seg.	1 segundo	1 segundo

Tabla 5.16: Comparativo entre clasificadores de IMS de 150x 150 utilizando comparación con imagen de referencia.

Podemos comparar los resultados entre la tabla 5.12 y 5.16. La imagen de prueba fue la misma, lo que cambio fue la forma de evaluación. En la tabla 5.12 se realizó evaluación de resultados utilizando validación cruzada y en la tabla 5.16 se generaron los resultados a partir de la comparación con una imagen de referencia. Comparando los resultados entre las dos tablas, observamos una mínima diferencia de exactitud, siendo sólo de centésimas de porcentaje. La evaluación de resultados utilizando una imagen de referencia resultó en mayores porcentajes.

## Capítulo 6

# Conclusiones y Trabajo a Futuro

En este trabajo de tesis se desarrolló un sistema para Percepción Remota que contiene programados métodos de clasificación supervisada y no supervisada más utilizados en la literatura estadística. Además el sistema tiene métodos de evaluación de clasificadores y mantiene la georreferencia de las imágenes satelitales.

Para conseguir el objetivo general de la tesis se realizó un análisis y diseño del sistema basado en ingeniería de software, la cual fue una parte medular en la implementación y modelado de esta tesis. La estructura del sistema esta constituido en clases, las cuales nos permitieron la división del problema real en partes más sencillas de resolver y manipular (divide y vencerás) logrando así el fraccionamiento de un problema complejo en pequeños problemas más simples con un grado de complejidad menor al problema general.

Trabajos a futuro pretenden la implementación de distintos clasificadores, utilizando los diferentes enfoques existentes, empezando por modelos probabilistas.

Otro punto importante que se pretende realizar en un futuro es que el sistema sea lo más libre posible, es decir, que no dependa de software comercial, como es el caso de HALCON que es un software comercial especializado en la manipulación y procesamiento de imágenes, el cual para su uso es necesaria la adquisición de una licencia. Los métodos utilizados en la presente tesis con HALCON son mínimos, los cuales se podrían implementar en el lenguaje C++.

Para la visualización de las imágenes temáticas generadas también se utilizó un control del IDE de visual C++ que permite la visualización de este tipo de imágenes (.tiff), el cual también esta contemplado que mas adelante se programe su propia librería para mostrar dichas imágenes.

# Apéndice A

## Especificación de casos de uso

### Especificación del C.U. : OBTENCIÓN DE DATOS DE ENTRENAMIENTO

#### 1.- Obtención datos de entrenamiento.

##### 1.1.- Breve Descripción:

Este caso de uso permite obtener una cierta cantidad de muestras de valores de grises de la imagen y asigna el valor medio de cada una de las muestras a una clase.

#### 2.- Flujo de Eventos

##### 2.1.- Flujo Básico

##### 2.1.1.- Flujo Básico Principal <Obtención datos>

2.1.1.1.- El sistema mostrará una pantalla en donde el usuario tendrá dos cuadros de texto en donde ingresará el número de clases en las que desea clasificar la imagen a procesar <E1> y el número de muestras de entrenamiento que obtendrá por cada clase elegida (entre más muestras de entrenamiento se tomen para una clase, más muestras representativas se tendrán por clase y por lo tanto se obtendrá una mejor clasificación) <E2>.

2.1.1.2.- El sistema mostrará 4 cuadros de texto en donde el usuario ingresará la ruta de las imágenes a ser leídas <E3>.

2.1.1.3.- El sistema mostrará las opciones que tiene para elegir la obtención de datos de entrenamiento, estas opciones son: obtención de

datos en forma manual y obtención de datos en forma automática. También, mostrará las opciones en donde puede elegir cargar datos de entrenamiento almacenados en el sistema, si es que existen ó elegir capturar nuevos datos de entrenamiento. Al inicio el sistema mostrará inactiva la opción de Cargar datos de entrenamiento almacenados ya que no se tienen guardados ningún tipo de datos, pero una vez que se hayan capturado datos de entrenamiento el sistema habilitará esta opción.

- 2.1.1.4.- Si el usuario selecciona la opción de cargar datos de entrenamiento nuevos y la opción de obtener datos de entrenamiento en forma manual, ir a 2.2.1.
- 2.1.1.5.- Si el usuario selecciona la opción de cargar datos de entrenamiento nuevos y la opción de obtener datos de entrenamiento en forma automática, ir a 2.2.2.
- 2.1.1.6.- Si el usuario selecciona la opción de cargar datos de entrenamiento almacenados, el sistema inhabilita las opciones de obtener datos de entrenamiento en forma manual y obtener datos de entrenamiento en forma automática ya que no se utilizará ninguna de estas opciones, porque los datos se cargaran desde el sistema. Ir a 2.2.3.
- 2.1.1.7.- Se ejecuta el caso de uso Clasificación.

## 2.2.- Flujos Alternos

### 2.2.1.- FA <1>: <Cargar datos de entrenamiento nuevos y obtener datos en forma manual>

- 2.2.1.1.- El sistema cargará las imágenes satelitales en sus diferentes bandas espectrales. En este caso como las imágenes satelitales a utilizar son las proporcionadas por el satélite SPOT-5, sólo se leerán 4 bandas espectrales y se cargaran en memoria <E4>.
- 2.2.1.2.- El sistema mostrará una nueva imagen generada, producto de la combinación de las imágenes leídas en las 4 bandas en donde se obtendrán las clases y muestras de entrenamiento.

- 2.2.1.3.- El sistema mostrará una ventana en donde el usuario ingresará en un cuadro de texto, el nombre que se le asignará a la clase de la cual obtendrá datos de entrenamiento.
- 2.2.1.4.- El sistema mostrará una ventana en donde se le pedirá al usuario que dibuje una región de la cual se obtendrán los datos de entrenamiento de las clases.
- 2.2.1.5.- El usuario dibujará con la ayuda del ratón un área de donde se obtendrán los datos de entrenamiento y se guardarán con su clase correspondiente.
- 2.2.1.6.- El sistema ejecutará el paso 2.2.1.4 tantas veces como se halla seleccionado en el cuadro de texto el número de muestras de la imagen y no mostrará la siguiente ventana hasta que se halla seleccionado un área de entrenamiento dentro de la imagen mostrada.
- 2.2.1.7.- Una vez terminado de ejecutar el paso 2.2.1.6 el sistema regresará a ejecutar el paso 2.2.1.3 y este ciclo se ejecutara tantas veces como se haya elegido en el cuadro de texto el número de clases a considerar.
- 2.2.1.8.- El sistema obtendrá de cada una de las clases, el valor medio producto de los datos de entrenamiento obtenidos anteriormente.
- 2.2.1.9.- Ir al paso 2.1.1.6
  
- 2.2.2.- **FA <2>: < Cargar datos de entrenamiento nuevos y obtener datos en forma automática>**
  - 2.2.2.1.- El sistema cargará las imágenes satelitales en sus diferentes bandas espectrales. En este caso como las imágenes satelitales a utilizar son las proporcionadas por el satélite SPOT-5, sólo se leerán 4 bandas espectrales y se cargarán en memoria <E4>.
  - 2.2.2.2.- El sistema mostrará una nueva imagen generada, producto de la combinación de 4 imágenes tomadas en distintas bandas espectrales de donde se obtendrán las clases y muestras de entrenamiento.
  - 2.2.2.3.- El sistema mostrará una ventana en donde el usuario ingresará en

un cuadro de texto, el nombre que se le asignará a la clase de la cual obtendrá datos de entrenamiento.

- 2.2.2.4.- El sistema mostrará una ventana en donde se le pedirá al usuario que seleccione un píxel de la imagen con ayuda de ratón.
- 2.2.2.5.- El usuario seleccionará un píxel y el sistema de forma automática generará un crecimiento de regiones basándose en sus características del píxel obteniendo sólo los píxeles con características similares. Estos valores de los píxeles se guardarán con su clase correspondiente.
- 2.2.2.6.- El sistema ejecutará el paso 2.2.2.4 tantas veces como se halla seleccionado en el cuadro de texto el número de muestras de la imagen y no mostrará la siguiente ventana hasta que se halla seleccionado un área de entrenamiento dentro de la imagen mostrada.
- 2.2.2.7.- Una vez terminado de ejecutar el paso 2.2.2.6 el sistema regresará a ejecutar el paso 2.2.2.3 y este ciclo se ejecutará tantas veces como se haya elegido en el cuadro de texto el número de clases a tomar.
- 2.2.2.8.- El sistema obtendrá de cada una de las clases, el valor medio producto de los datos de entrenamiento obtenidos anteriormente.
- 2.2.2.9.- Ir al paso 2.1.1.6.

### 2.2.3.- **FA <3>: < Cargar datos de entrenamiento almacenados >**

- 2.2.3.1.- El sistema cargará el valor medio de las clases de entrenamiento, producto de una obtención de datos de entrenamiento en forma manual ejecutada con anterioridad.
- 2.2.3.2.- Ir al paso 2.1.1.6.

## 2.3.- **Flujos de Excepción**

### 2.3.1.- **FE <E1>: < Número de clases a tomar >**

- 2.3.1.1.- Si el número de clases ingresado es inferior a 1 o es superior a 10, se envía un mensaje de error especificando al usuario que el rango

de las clases está entre 1 y 10 por lo que debe ingresar un número entre ese rango.

2.3.1.2.- Se regresa al paso 2.1.1.1.

2.3.2.- **FE <E2>: < Número de muestras a tomar >**

2.3.2.1.- Si el número de muestras ingresado es inferior a 1 y superior a 10, se envía un mensaje de error especificando al usuario que el rango de las muestras está entre 1 y 10 por lo que debe ingresar un número entre ese rango.

2.3.2.2.- Se regresa al paso 2.1.1.1.

2.3.3.- **FE <E3>: < Ingresar ruta de imágenes >**

2.3.3.1.- Si no se ingresa una dirección de imágenes en alguno de los cuatro cuadros de diálogos el sistema muestra un mensaje de error en donde especifica que no se pueden dejar cuadros de textos en blanco en donde debe de especificarse la dirección de alguna de las bandas a clasificar.

2.3.3.2.- Regresar a 2.1.1.2.

2.3.4.- **FE <E4>: < Lectura de imágenes >**

2.3.4.1.- Si ocurrió un error a la hora de leer las imágenes, el sistema muestra un mensaje de error y se cierra el sistema.

2.3.4.2.- Caso de Uso termina.

### 3.- **Precondiciones**

3.1.- **< Cargar imágenes a ser clasificadas >**

Para poder ejecutar este caso de uso previamente se debio haber ejecutado iniciar el programa.

### 4.- **Postcondiciones**

4.1.- **< Caso de uso: Clasificación >**

Una vez ejecutado este caso de uso se podrá solamente ejecutar el caso de

uso Clasificación, en donde se hará la clasificación de una nueva imagen con base a los datos de entrenamiento obtenidos en este caso de uso.

## **Especificación del C.U : CLASIFICACIÓN**

### **1.- Clasificación**

#### **1.1.- Breve Descripción**

Este caso de uso genera una imagen temática producto de la elección de alguno de los clasificadores implementados. En la nueva imagen generada se podrá saber sus coordenadas de mapa  $X$  y  $Y$  pasando el ratón sobre dicha imagen.

### **2.- Flujo de Eventos**

#### **2.1.- Flujo Básico**

##### **2.1.1.- Flujo Básico Principal <Clasificación>**

2.1.1.1.- El sistema mostrará en pantalla opciones en donde el usuario podrá elegir el tipo de clasificador a usar. Estos clasificadores se dividen en dos grupos: clasificadores supervisados y clasificadores no supervisados. En el grupo de clasificadores supervisados el usuario puede elegir: el clasificador de Distancia Mínima a la Media, el clasificador de Distancia Mahalanobis y el clasificador de Máxima Probabilidad. En los clasificadores no supervisados sólo puede elegir el clasificador ISODATA.

2.1.1.2.- El usuario selecciona un tipo de clasificador.

2.1.1.3.- Si la opción seleccionada es Distancia Mínima a la Media ir a 2.2.1.

2.1.1.4.- Si la opción seleccionada es Distancia Mahalanobis ir a 2.2.2.

2.1.1.5.- Si la opción seleccionada es Máxima Probabilidad ir a 2.2.3.

2.1.1.6.- Si la opción seleccionada es ISODATA ir a 2.2.4.

2.1.1.7.- El sistema muestra en una ventana la nueva imagen temática generada.

2.1.1.8.- El sistema muestra las coordenadas de mapa terrestres al pasar el ratón sobre la nueva imagen generada.

2.1.1.9.- El caso de uso termina.

## 2.2.- Flujos Alternos

### 2.2.1.- FA <1>: <Distancia Mínima a la Media>

2.2.1.1.- El sistema ejecuta el algoritmo de clasificación de Distancia Mínima a la Media (DMM).

2.2.1.2.- El sistema asigna cada píxel de la imagen de entrada a una clase definida anteriormente según el algoritmo de distancia mínima a la media y al final se genera una nueva imagen en donde se visualizan las clases encontradas.

2.2.1.3.- Regresar a 2.1.1.7.

### 2.2.2.- FA <2>: <Distancia Mahalanobis>

2.2.2.1.- El sistema ejecuta el algoritmo de clasificación de Distancia Mahalanobis (DMH).

2.2.2.2.- El sistema asigna cada píxel de la imagen de entrada a una clase definida anteriormente según el algoritmo de Distancia Mahalanobis y al final se genera una nueva imagen en donde se visualizan las clases encontradas.

2.2.2.3.- Regresar a 2.1.1.7.

### 2.2.3.- FA <3>: <Máxima Probabilidad>

2.2.3.1.- El sistema ejecuta el algoritmo de clasificación de Máxima Probabilidad.

2.2.3.2.- El sistema asigna cada píxel de la imagen de entrada a una clase definida anteriormente según el algoritmo de Máxima Probabilidad y al final se genera una nueva imagen en donde se visualizan las clases encontradas.

2.2.3.3.- Regresar a 2.1.1.7.

### 2.2.4.- FA <4>: <ISODATA>

- 2.2.4.1 El sistema muestra una ventana en donde el usuario podrá ingresar el número deseado de clases, los miembros mínimos de cada clase, la desviación estándar máxima, la mínima distancia entre medias, el número máximo de iteraciones y el número máximo de parejas a unir por iteración.
- 2.2.4.2 El usuario ingresa los datos solicitados. <E1> <E2> <E3> <E4> <E5> <E6>
- 2.2.4.3 El sistema ejecuta el algoritmo ISODATA tomando en cuenta los datos ingresados.
- 2.2.4.4 El sistema genera una nueva imagen en donde se muestran las clases encontradas en la imagen de entrada.
- 2.2.4.5 Regresar a 2.1.1.7.

### 2.3.- Flujos de Excepción

#### 2.3.1.- FE <E1>: <Número deseado de clases>

- 2.3.1.1.- Si el número deseado de clases ingresado es inferior a 1 o es superior a 10, se envía un mensaje de error especificando al usuario que el rango de las clases está entre 1 y 10 por lo que debe ingresar un número entre ese rango.
- 2.3.1.2.- Se regresa al paso 2.2.4.1.

#### 2.3.2.- FE <E2>: <Miembros mínimos de cada clase>

- 2.3.2.1.- Si el número de miembros mínimos de una clase ingresado es inferior a 1 y superior a 50,000, se envía un mensaje de error especificando al usuario que el rango de las clases está entre 1 y 50,000 por lo que debe ingresar un número entre ese rango.
- 2.3.2.2.- Se regresa al paso 2.2.4.1.

#### 2.3.3.- FE <E3>: <Desviación estándar máxima>

- 2.3.3.1.- Si el número de desviación estándar máxima ingresado es inferior a 1 y superior a 100, se envía un mensaje de error especificando

al usuario que el rango de las clases está entre 1 y 100 por lo que debe ingresar un número entre ese rango.

2.3.3.2.- Se regresa al paso 2.2.4.1.

**2.3.4.- FE <E4>: <Mínima distancia entre medias>**

2.3.4.1.- Si el número de mínima distancia entre medias ingresado es inferior a 1 y superior a 100, se envía un mensaje de error especificando al usuario que el rango de las clases está entre 1 y 100 por lo que debe ingresar un número entre ese rango.

2.3.4.2.- Se regresa al paso 2.2.4.1.

**2.3.5.- FE <E5>: <Máximo de Iteraciones>**

2.3.5.1.- Si el número máximo de iteraciones ingresado es inferior a 1 y superior a 10, se envía un mensaje de error especificando al usuario que el rango de las clases está entre 1 y 10 por lo que debe ingresar un número entre ese rango.

2.3.5.2.- Se regresa al paso 2.2.4.1.

**2.3.6.- FE <E6>: <Parejas a unir>**

2.3.6.1.- Si el número máximo de parejas a unir ingresado es inferior a 1 y superior a 5 se envía un mensaje de error especificando al usuario que el rango de las clases está entre 1 y 5 por lo que debe ingresar un número entre ese rango.

2.3.6.2.- Se regresa al paso 2.2.4.1.

### **3.- Precondiciones**

**3.1.- < Caso de uso: Obtención de datos de entrenamiento >**

Para ejecutar este caso de uso se debió haber previamente ejecutado el caso de uso obtención de datos de entrenamiento. Esto solo aplica para clasificadores supervisados, ya que para el clasificador no supervisado ISODATA

solo es necesario haber ingresado las imágenes a ser procesadas.

#### **4.- Postcondiciones**

##### **4.1.- < Caso de uso: Validación de Resultados >**

Una vez ejecutado este caso de uso se podrá ejecutar el caso de uso de validación de resultados ya que se tomarán los datos obtenidos por el caso de uso clasificación para obtener el desempeño del clasificador.

#### **Especificación del C.U : VALIDACIÓN DE RESULTADOS**

##### **1.- Validación de Resultados.**

###### **1.1.- Breve Descripción**

Este caso de uso genera un reporte en donde se mostrarán los resultados obtenidos con respecto a la exactitud de cada clasificador. Estos datos mostrados serán el desempeño global del clasificador ejecutado con respecto a la imagen procesada, el error por omisión, el error por comisión y la estadística Kappa.

##### **2.- Flujo de Eventos.**

###### **2.1.- Flujo Básico**

###### **2.1.1.- Flujo Básico Principal <Validación>**

2.1.1.1.- Una vez ejecutado el caso de uso clasificación, el sistema activará un cuadro de texto en donde el usuario ingresará el número de partes en las que se dividirán los datos generados de la clasificación y también se activará un botón para generar una matriz de confusión en donde se mostrarán los datos producto de la validación cruzada.

2.1.1.2.- El usuario ingresa el número de subconjuntos deseado para dividir el archivo generado por el caso de uso clasificación.

2.1.1.3.- El usuario ejecuta el botón de generar matriz de confusión.

- 2.1.1.4.- El sistema verifica que el dato ingresado sea correcto y se encuentre en el rango especificado <E1>.
- 2.1.1.5.- El sistema verifica las opciones seleccionadas por los casos de uso anteriores y ejecuta la validación del clasificador seleccionado.
- 2.1.1.6.- Si la opción de clasificación seleccionada es la de Distancia Mínima a la Media ir a 2.2.1.
- 2.1.1.7.- Si la opción de clasificación seleccionada es la de Distancia Mahalanobis ir a 2.2.2.
- 2.1.1.8.- Si la opción de clasificación seleccionada es la de Máxima Probabilidad ir a 2.2.3.
- 2.1.1.9.- El sistema muestra una pantalla en donde se visualizan los datos generados por la validación cruzada: el error por omisión, comisión, desempeño global y estadística Kappa.
- 2.1.1.10.- El caso de uso termina.

## 2.2.- Flujos Alternos

### 2.2.1.- FA <1>: <Validación Distancia Mínima a la Media>

- 2.2.1.1.- El sistema lee el archivo generado de la clasificación de Distancia Mínima a la Media y lo divide en los  $k$  subconjuntos ingresados por el usuario.
- 2.2.1.2.- El sistema ejecuta nuevamente el clasificador de Distancia Mínima a la Media para realizar la validación cruzada del mismo.
- 2.2.1.3.- El sistema obtiene los valores de la matriz de confusión.
- 2.2.1.4.- El sistema regresa a 2.1.1.9.

### 2.2.2.- FA <2>: <Validación Distancia Mahalanobis>

- 2.2.2.1.- El sistema lee el archivo generado de la clasificación de Distancia Mahalanobis y lo divide en los  $k$  subconjuntos ingresados por el usuario.
- 2.2.2.2.- El sistema ejecuta el clasificador de Distancia Mahalanobis para realizar la validación cruzada del mismo.

2.2.2.3.- El sistema obtiene los valores de la matriz de confusión.

2.2.2.4.- El sistema regresa a 2.1.1.9.

### 2.2.3.- **FA <3>: < Validación Máxima Probabilidad >**

2.2.3.1.- El sistema lee el archivo generado de la clasificación de Máxima Probabilidad y lo divide en  $k$  subconjuntos ingresados por el usuario.

2.2.3.2.- El sistema ejecuta el clasificador de Máxima Probabilidad para realizar la validación cruzada del mismo.

2.2.3.3.- El sistema obtiene los valores de la matriz de confusión.

2.2.3.4.- El sistema regresa a 2.1.1.9.

## 2.3.- **Flujos de Excepción**

### 2.3.1.- **FE <E1>: <Número de partes a dividir >**

2.3.1.1.- Si el número deseado de clases ingresado es inferior a 1 o es superior a 10, se envía un mensaje de error especificando al usuario que el rango de las clases está entre 1 y 10 por lo que debe ingresar un número entre ese rango.

2.3.1.2.- Se regresa al paso 2.1.1.2.

## 3.- **Precondiciones**

### 3.1.- **< Caso de uso: Clasificación >**

Para ejecutar este caso de uso se debió haber previamente ejecutado el caso de uso Clasificación.

# Apéndice B

## Descripción de las clases

### **CLASE: CLASIFICADORESDLG.CPP**

La función principal de esta clase es mostrar el ambiente gráfico de la interfaz y efectuar las acciones de los botones ejecutados por el usuario.

Esta clase contiene tres botones principales, los cuales son: un botón para cerrar el sistema, otro para iniciar la ejecución del clasificador deseado y otro para obtener la matriz de confusión.

Entre sus principales funciones se encuentran las siguientes:

- **Función: Inicio ()**. Es la función principal del botón que ejecuta el clasificador seleccionado, en la cual se valida toda la información ingresada por el usuario, se ejecuta el clasificador elegido y se obtiene información de georreferenciación.

Las principales validaciones realizadas por esta función son las siguientes:

- Valida que el número de clases y muestras de entrenamiento elegidas para cada clase estén dentro de un rango especificado por defecto. El rango para el número de clases y para las muestras de entrenamiento tiene que ser mayor a 0 y menor o igual a 10.
- Valida que los campos en donde se ingresa la ruta de los archivos a ser leídos no estén vacíos.
- Verifica qué tipo de muestras de entrenamiento se desean obtener: si se desean obtener muestras de entrenamiento nuevas o si se desea leer las muestras de entrenamiento almacenadas en el sistema.

- Verifica la forma de obtener las muestras de entrenamiento: si es en forma manual, que es en donde el usuario ingresa una región de entrenamiento dentro de la imagen ó de forma automática, que es en donde el usuario solo ingresa un único punto dentro de la imagen.
- Verifica qué clasificador es el seleccionado y lo ejecuta. Existen cuatro clasificadores los cuales son: El clasificador de Distancia Mínima a la Media (DMM), el clasificador de Distancia Mahalanobis (DMH), el clasificador de Máxima Probabilidad (MP) y el clasificador ISODATA.
- Valida que el número de folds(partes) para realizar la validación cruzada sea mayor a 0 y menor o igual a 10.

Esta clase utiliza varias clases auxiliares las cuales sirven de apoyo para obtener las muestras de entrenamiento, para llamar a ejecutar el clasificador deseado, para mostrar la imagen temática y para obtener nueva información del clasificador ISODATA. Todas estas clases y las funciones pertenecientes a las mismas las detallaremos mas adelante.

- **Función: OnCrossValidation ()**. Esta es la función principal del botón de validación cruzada, en la cual se obtiene una matriz de confusión, que nos indica el desempeño global del clasificador ejecutado. Además muestra el porcentaje de errores por comisión y omisión de cada clase, así también como la estadística Kappa.

Toda esta información es mostrada en una nueva ventana para la visualización de resultados. Para realizar el proyecto se requiere de clases auxiliares las cuales son: una clase que valida el clasificador y una clase que muestra en pantalla los resultados obtenidos.

#### **CLASE: OBTENCIONDATOS.CPP**

Esta clase proporciona funciones que ayudarán a la manipulación y obtención de datos de entrenamiento. También proporciona funciones para leer las imágenes dadas por el usuario y mostrarlas en pantalla como una sola imagen.

Tiene 4 funciones principales las cuales son:

- **Función: Cargar-imagenes (var1, var2, var3, var4, var5)**. El objetivo de esta función es leer las imágenes ingresadas por el usuario y cargarlas en memoria.

Estas imágenes son leídas de los primeros 4 parámetros los cuales son las rutas de cada una de las imágenes.

Una vez leídas las imágenes, la función crea una nueva imagen producto de la composición de 3 imágenes y la muestra en una nueva ventana.

- **Función: void Obtencion-manual (var1, var2).** El objetivo de esta función es obtener para cada clase identificada por el usuario, una o varias muestras de entrenamiento dadas. El número de clases y el número de muestras de entrenamiento dependen del primer y segundo parámetro de la función. Se crearán tantas clases como el usuario desee y para cada clase se tomarán tantas muestras como también el usuario elija.

Para cada clase definida por el usuario, éste ingresará un nombre de cómo desea llamar a la clase, y para la obtención de las muestras de entrenamiento el usuario elegirá con el ratón los datos de entrenamiento dibujando un área sobre la imagen mostrada. Esta área servirá para obtener los valores de gris de cada una de las imágenes y guardarlos como datos de entrenamiento en un archivo para su posterior procesamiento.

- **Función: void Obtencion-Automatica (var1, var2).** El objetivo de esta función es el mismo que el de la función de Obtencion-manual, la diferencia está en la forma en que se obtienen los datos de entrenamiento.

Para obtener los datos de entrenamiento el usuario sólo ingresa un punto en la imagen mostrada con la ayuda del ratón dando clic en un punto. El sistema toma el valor de gris de cada una de las imágenes leídas y realiza un crecimiento de regiones, es decir toma los píxeles similares que están cerca del píxel ingresado y los almacena en la muestra de entrenamiento de dicha clase.

Al final se obtiene un archivo que contiene las clases con sus respectivas muestras de entrenamiento.

- **Función: ObtenMatrizMedias(var1, var2, var3).** Este método obtiene una matriz que contiene todos los valores de gris de las imágenes leídas de cada una de las 4 bandas de una clase específica.

## **CLASE: TIPOSCLASIFICADORES.CPP**

Esta clase es una de las más importantes, ya que cuenta con los métodos que clasifican a las imágenes satelitales para generar la imagen temática.

Cuenta con 4 funciones que son:

- **Función: Clasificador-DMM (var 1, var 2, var 3, var 4, var 5, var 6).**

Este método genera una nueva imagen, con base en las clases elegidas y en los valores de las muestras de entrenamiento. Al final se genera un archivo en donde se tiene a que clase perteneció cada píxel de la imagen y la imagen temática correspondiente.

Su procedimiento es el siguiente:

Se obtiene una matriz que contiene los valores de gris de las muestras de entrenamiento de cada clase. Por cada clase es una matriz de valores de gris. La dimensión de la matriz es el número de bandas ingresadas por el usuario, en este caso son 4 imágenes lo que equivale a 4 bandas por el número de píxeles obtenidos en las muestras de entrenamiento de dicha clase.

Después por cada matriz que contiene los valores de gris de cada clase se obtiene el promedio de cada banda de la matriz (cuatro bandas entonces son 4 promedios, uno por cada banda o imagen). Se comienza a hacer un barrido de las imágenes leídas para obtener los valores de gris de toda la imagen y se calcula la distancia espectral entre cada clase con respecto al valor de gris leído. Después, la clase que esté más cerca al píxel es la que se asigna. Este procedimiento se repite hasta que todos los píxeles estén clasificados. Al final se genera la imagen temática.

- **Función: DistanciaMahalanobis (var 1, var 2, var 3, var4, var 4, var 6).**

La función principal de este método es la misma que la del método de DMM, obtener una imagen temática.

El algoritmo es diferente y consiste en lo siguiente:

Además de sacar la matriz de valores de gris de las clases y los promedios de cada matriz como en el método anterior, también se obtiene la matriz de covarianza de las clases a clasificar.

Con estos datos obtenidos se aplica el algoritmo que realiza un barrido de toda la imagen y obtiene el valor de gris de cada banda en cada punto y determina mediante la fórmula de la distancia Mahalanobis la similitud que hay entre el

valor de gris de cada banda y las clases muestra de clasificación. La clase que tenga una mayor similitud ó menor distancia será la clase con la cual se clasifique a el píxel. Este procedimiento se realiza para cada píxel de la imagen. Al final se obtiene una imagen temática con las clases elegidas por el usuario.

- **Función: MaximaProbabilidad(var 1, var 2, var 3, var 4, var 5, var 6).**

Este método como los dos anteriores consiste en generar una imagen temática en donde se muestren las clases de interés para el usuario.

Este método obtiene la matriz de valores de gris para cada clase, el promedio de valores de gris de cada banda presente en la matriz, la matriz de covarianza de cada clase y el determinante de cada matriz de covarianza. Con estos datos el método realiza un barrido de toda la imagen y obtiene el valor de gris de cada punto. Mediante probabilidad determina que píxel tiene una mayor probabilidad de pertenecer a una clase. Esto se hace para todos los píxeles de la imagen. Al final se obtiene una imagen temática producto de la clasificación del algoritmo.

#### **CLASE: ISODATA.CPP**

Esta clase muestra una ventana grafica en donde se solicita al usuario que ingrese los parametros necesarios para la ejecución del algoritmo ISODATA. Además, tiene funciones que facilitan la ejecución de dicho algoritmo.

Entre sus principales métodos se encuentran los siguientes:

- **Función: obtenArchivoDatos(var 1, var 2, var 3, var 4, var 5, var 6, var 7).**

Esta función obtiene todos los valores de gris de toda la imagen y los guarda en un archivo de texto.

- **Función: ObtenClaseCercana(var 1, var 2, var 3).** Esta función obtiene la clase que esta más cerca de una clase especifica, por medio de su distancia espectral.

- **Función: ObtenPromedioCentroides(var 1, var 2, var 3, var 4, var 5, var 6...).** Esta función obtiene de manera aleatoria un punto de la imagen y obtiene sus valores de gris en las 4 bandas de la imagen.

- **Función: recalculaPromedio(var 1, var 2).** Esta función calcula el promedio de valores de gris de cada banda en cada clase.

- **Función: ActualizaArchivoClusters(var 1, var 2, var 3).** Esta función actualiza la clase que va a fusionarse con otra clase por no cumplir con los requerimientos necesarios para poder ser definida como una clase.
- **Función: ordenaArchivo(var 1).** Esta función ordena todos los valores de grises que estén separados a lo largo de todo un archivo y que pertenezcan a una clase en específico de forma consecutiva para su rápido manejo de información.
- **Función: ObtenDMMcadaClase(var 1, var 2, var 3, var 4).** Esta función obtiene la distancia euclidiana que hay entre el centro de cada clase y un punto específico de la imagen.
- **Función: ObtenDistanciaMediaTotal(var 1, var 2, var 3).** Este método obtiene la distancia total que existe combinando las distancias de las 4 bandas en una sola distancia.
- **Función: CalculaDesviacionStandar(var 1, var 2, var 3).** Esta función calcula todas las desviaciones estándar de todos los valores de gris de todas las bandas.
- **Función: ObtenDesviacionStandarMaxima(var 1, var 2).** Este método calcula la desviación estándar máxima que existe entre las 4 bandas para un mismo punto.
- **Función: redimensionaPromedio(var 1, var 2, var 3, var 4, var 5)** Esta función ingresa los nuevos promedios obtenidos producto de dividir una nueva clase en dos, es decir, redimensiona el vector de promedios de las clases.
- **Función: reAgrupaClases(var 1, var 2, var 3, var 4)** Esta función asigna los píxeles de una imagen a una nueva clase dependiendo de la distancia euclidiana que existe entre ellos y el centro de la clase.

#### **CLASE: VALIDACIONCLASIFICADORES.cpp**

- **Función: ObtenNumTotalPixeles(var 1).** Como su nombre lo dice este método obtiene el número total de píxeles presentes en una imagen.
- **Función: OrdenaDatosAleatorios(var 1, var 2, var 3).** Esta función divide el número total de píxeles en  $n$  partes iguales de forma aleatoria. Al final se tienen  $n$  archivos con el mismo número de píxeles cada uno.

- **Función: ConcatenarArchivos(var 1, var 2).** Este método concatena dos archivos.
- **Función: ObtenTamClases(var 1 var 2 var 3 var 4).** Función que ordena los píxeles de un archivo de forma que los píxeles que estén en una clase menor los ingresa al inicio del archivo y los mayores a esta clase los va poniendo en su lugar correspondiente de forma ascendente.
- **Función: k-FoldCrossValidation(var 1, var 2, var 3, var 4, var 5).** Función que obtiene el número de píxeles de cada clase.
- **Función: ObtenPocentajeOnek-fold(var 1, var 2).** Este método recibe dos archivos y obtiene los píxeles que están clasificados correctamente y los que no están clasificados correctamente.
- **Función: ObtenMatrizConfusion(var 1, var 2, var 3).** Este método obtiene una matriz en donde se indica los píxeles que se clasificaron correctamente y los píxeles que no, indicando en qué clase fueron asignados.
- **Función: ObtenDesempeñoGlobal(var 1).** Esta función obtiene el desempeño global del clasificador, es decir, divide el total de píxeles que fueron clasificados correctamente entre el número total de píxeles de la imagen procesada. Al final se muestra un porcentaje de desempeño del clasificador.
- **Función: ObtenErrorOmission(var 1, var 2).** Esta función obtiene un porcentaje de error para cada clase con respecto a las columnas de la matriz de confusión obtenida.
- **Función: ObtenErrorComision(var 1, var 2).** Esta función obtiene un porcentaje de error para cada clase con respecto a las filas de la matriz de confusión obtenida.
- **Función: ObtenPíxelesClasificadasCorrectamente(var 1, var 2).** Este método obtiene la suma de la diagonal principal de la matriz de confusión.
- **Funcion: ObtenSumFilasColumns(var 1, var 2).** Función que multiplica las filas por todas las columnas de una matriz y al final suma esos resultados obtenidos.

- **Función: ObtenEstadisticaKappa(var 1, var 2, var 3).** Este método obtiene la estadística kappa.

#### **CLASE: MATRIZCOV.CPP**

Esta clase cuenta con métodos matemáticos los cuales sirven de apoyo a las clases de tiposclasificadores.cpp y a la clase isodata.cpp.

Entre los principales encontramos los siguientes:

- **Función: ObtenPromedio(var 1, var 2, var 3).** Esta función obtiene el promedio de cada una de las bandas en una clase específica.
- **Función: ObtenMatrizResta(var 1, var 2, var 3, var 4).** Esta función recibe de parámetro una matriz que contiene todos los valores de grises de una clase en sus 4 bandas y resta a cada valor de gris su promedio correspondiente a su banda.
- **Función: ObtenMatrizTrans(var 1, var 2, var 3).** Esta función obtiene la matriz transpuesta de una matriz dada.
- **Función: ObtenMatrizCov(var 1, var 2, var 3, var 4).** Esta función obtiene una matriz de covarianza.
- **Función: ObtenDeterminante(var 1).** Esta función obtiene el determinante de una matriz de 3 x 3.
- **Función: matAdj(var 1, var 2, var 3, var 4).** Esta función obtiene una matriz adjunta.
- **Función: Obten-PrediccionMaHala(var 1, var 2, var 3, var 4, var 5, var 6).** Función que calcula la Distancia Mahalanobis de una clase dada, con respecto a un punto específico.
- **Función: Obten-PrediccionMP(var 1, var 2, var 3, var 4, var 5, var 6, var 7).** Función que calcula la probabilidad de un píxel de la imagen de pertenecer a una clase dada.
- **Función: ObtenClaseDMM-Mahalanobis(var 1, var 2).** Función que obtiene la clase con la distancia espectral mínima en un punto dado.

- **Función: ObtenClaseMaximaProbabilidad(var 1, var 2).** Función que obtiene la clase con la mayor probabilidad de pertenecer a una clase dada.
- **Función: DistanciaSpectral(var 1, var 2).** Función que obtiene la distancia espectral entre dos puntos dados.

#### **CLASE: MATRIZCONFUSION.CPP**

Esta clase muestra una ventana en donde se visualizan todos los resultados producto de la validación cruzada, estos resultados muestran los datos como son: el error por comisión, el error por omisión, el desempeño Global y la estadística Kappa. Tiene un único método el cual es:

- **Función: CargarDatos().** Esta función lo que realiza es leer todos los datos de la matriz de confusión y cargarlos en la ventana que se mostrará al usuario.

#### **CLASE: NAMECLASS.CPP**

Es una clase en donde se muestra una ventana grafica en donde se le indica al usuario que ingrese el nombre de cada clase.

Cuenta con un solo método el cual es:

- **Función: Onacceptar().** Este método lee el nombre ingresado por el usuario y lo guarda para su posterior uso en donde sea requerido.

#### **CLASE: MOSTRARIMGTIFF.CPP**

Esta clase muestra una ventana con la imagen temática obtenida de la clasificación de coberturas y sus coordenadas de referencia.

Sus métodos son los siguientes:

- **Función: showImage(var 1, var 2).** Esta función carga la imagen generada de la clasificación de coberturas y la muestra en la ventana para su visualización en pantalla.
- **Funcion: OnMouseMoveScribblectrl1(var 1, var 2, var 3, var 4).** Esta función obtiene las coordenadas de referencia geográfica dependiendo de la posición en que se encuentre el ratón sobre la imagen temática.

### **CLASE: ARCHIVOCLASIFICADO.CPP**

La función principal de este archivo es ir pintado píxel por píxel cada punto de la nueva imagen generada para que al final se forme la imagen temática producto del clasificador elegido.

Su función principal es la siguiente:

- **Funcion: PintaPixel(var 1, var 2, var 3, var 4, var 5, var 6).** Este método pinta de un color el píxel de acuerdo a su clase.

# Referencias

- [1] Dr. Manuel Arbelo. *Introducción a la Teledetección*. Universidad de la Laguna, [http://webpages.ull.es/users/marbelo/rs.html](http://webpages ull.es/users/marbelo/rs.html), 2005.
- [2] Uriel Pérez G. *La Percepción Remota*. Universidad del Tolima, <http://www.ut.edu.co/profesores/upg>, Febrero de 2005.
- [3] Julian Eugene Boggess. A genetic algorithm approach to identifying roads in satellite images. *Compute Science Department, Mississippi State University*, 2002.
- [4] J. Liu, G. Shao, H. Zhu, and S. Liu. A neuronal network approach for information extraction from remotely sensed data. *Proceedings of the 12th International Conference on Geoinformatics*, pages 655–662, June 2004.
- [5] J. Shulcloper, A. Guzmán, and J.F. Martínez. *Enfoque Lógico Combinatorio al Reconocimiento de Patrones*. Editorial Politécnica, 1999.
- [6] Zoltan Kato Home Page. Supervised image segmentation using markov random fields, <http://www.inf.u-szeged.hu/kato/software/mrfdemo.html>.
- [7] Gerhard Winkler. *Image Analysis, Random Fields and Markov Chain Monte Carlo Methods - A Mathematical Introduction*. Springer, 2006.
- [8] M. Oruc, A.M. Marangoz, and G. Buyuksalih. Comparison of pixel-based and object-oriented classification approaches using landsat-7 etm spectral bands. *Proceedings of ISPRS (International Society for Photogrammetry and Remote Sensing) Conference, XXth Congress*, 35:1118–1123, July 2004.
- [9] E. Chuvieco. *Fundamentos de teledetección espacial*. Madrid, España, 1995.
- [10] Spot 5. Página SPOT 5, <http://spot5.cnes.fr/gb/index2.htm>.

- [11] J. P. Marques de Sá. *Pattern Recognition. Concepts, Methods and Applications*. May, 2001.
- [12] Ariel Carrasco O. Reconocimiento de patrones. *INAOE*, <http://ccc.inaoep.mx/~ariel/recpat.pdf>.
- [13] John A. Richards and Xiuping Jia. *Remote Sensing Digital Image Analysis: an Introduction*. Third edition, 1999.
- [14] David A. Landgrebe. *Signal Theory Methods in Multispectral Remote Sensing*. Wiley-Interscience, 2003.
- [15] A. Skidmore and H. Prins. *Environmental Modelling with GIS and Remote Sensing*. Taylor & Francis London and New York, 2003.
- [16] Sing-Tze Bow. *Pattern Recognition and Image Preprocessing*. 2002.
- [17] Ian Sommerville. *Ingeniería de Software*. sexta edición, 2002.
- [18] Terry Quatrani. *Visual Modeling with Rational Rose 2002 and UML*. October 09 2002.
- [19] Mvtec halcon. Página de Halcon, <http://www.mvtec.com/halcon/>.
- [20] Juan Francisco Robles Camacho. Extracción de mapas temáticos a partir de la clasificación en imágenes satelitales. Tesis de Maestría, INAOE, Febrero 2007.