



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

REGULACIÓN ADAPTATIVA DE ESCENAS
DEL SATÉLITE ENVISAT

T E S I S

QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN TELECOMUNICACIONES

PRESENTA:

RAFAEL MORA SÁNCHEZ

DIRECTOR DE TESIS:

DR. MIGUEL MOCTEZUMA FLORES

CIUDAD UNIVERSITARIA, MARZO 2015





UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

REGULACIÓN ADAPTATIVA DE ESCENAS
DEL SATÉLITE ENVISAT

T E S I S

QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN TELECOMUNICACIONES

PRESENTA:

RAFAEL MORA SÁNCHEZ

DIRECTOR DE TESIS:

DR. MIGUEL MOCTEZUMA FLORES

CIUDAD UNIVERSITARIA, MARZO 2015



Agradecimientos

A Dios Padre y Jesucristo

“Hoy es el mejor día para agradecerles por las cosas maravillosas que han hecho en mi vida y guiar mi camino en todo momento, los amo”

A Papá Rafa, mamita Sai, Génesis y mi pequeña Luisa.

“Gracias porque han sido el pilar en mi vida, que han estado conmigo en las buenas y en las malas, que han confiado en mí, hoy me queda agradecerles por todo el gran apoyo que me han brindado”

Al Dr. Miguel Moctezuma Flores

“Gracias a la guía que me brindó pude realizar este trabajo, usted es el pilar de mi tesis”

A mis tíos: Celia, Manolo, Rita, Adolfo, Flavia, Magaly, Mary, Vite, Eliseo, Genoveva y a mi querida abuela María.

“A todos mis familiares que me brindaron su apoyo incondicional, siempre les voy a estar agradecido”

A mis amigos: Paco, Arely, Pantaleón, Anita, Uriel, Marcos y Daniel.

“A mi compañeros y amigos que son más valiosos que cualquier tesoro que exista en esta vida”

| | |
|---|-----------------|
| Índice general..... | <i>i</i> |
| 1. Introducción..... | 1 |
| 1.1 Objetivo..... | 2 |
| 1.2 Fundamentación..... | 2 |
| 1.3 Definición del problema..... | 3 |
| 1.4 Aportación de la tesis..... | 3 |
| 2. Probabilidad y procesos estocásticos..... | 4 |
| 2.1 Axiomas de probabilidad..... | 5 |
| 2.2 Variable aleatoria y funciones de probabilidad..... | 5 |
| 2.3 Distribución gaussiana..... | 7 |
| 2.4 La ecuación de Bayes..... | 12 |
| 3. Campos aleatorios de Markov..... | 13 |
| 3.1 Procesos aleatorios en la percepción remota..... | 13 |
| 3.2 Topologías | 14 |
| 3.2.1 Vecindades..... | 16 |
| 3.2.2 Cliques..... | 18 |
| 3.2.3 Propiedades de Markov..... | 19 |
| 3.3 Modelos de Markov | 21 |
| 3.3.1 Modelo de Ising..... | 21 |
| 3.3.2 Modelo de Potts..... | 23 |
| 4. El satélite ENVISAT..... | 24 |
| 4.1 Objetivos..... | 24 |
| 4.2 Parámetros orbitales..... | 24 |
| 4.3 Instrumentos..... | 26 |
| 4.4 La herramienta NEST [®] | 32 |
| 5. Optimización por Campos de Markov..... | 38 |
| 5.1 Fundamento..... | 38 |
| 5.2 Esquema de Metrópolis..... | 39 |
| 5.3 Recocido simulado..... | 41 |
| 5.4 Muestreador de Gibbs..... | 43 |
| 5.5 Parámetros: vecindades, cliques, temperatura y homogeneidad..... | 44 |
| 6. Determinación de la extensión marina de derrame de petróleo por medio de campos aleatorios de Markov..... | 47 |
| 7. Esquema de regulación adaptativa..... | 46 |
| 7.1 Planteamiento y esquema general..... | 46 |
| 7.2 Funciones de energía..... | 48 |
| 7.3 Algoritmo de recocido simulado..... | 50 |
| 7.4 Matlab [®] como herramienta de procesamiento..... | 53 |
| 8. Resultados..... | 54 |
| 8.1 Imágenes de prueba (esquema general)..... | 54 |
| 8.2 Comparación..... | 64 |
| 8.3 imágenes optimizadas (esquema propuesto)..... | 69 |
| 8.4 Comparación..... | 83 |
| 9. Conclusiones..... | 84 |
| Bibliografía..... | <i>ii</i> |
| Índice de figuras..... | <i>iii</i> |
| Anexos..... | <i>iv</i> |
| | <i>i</i> |

1. Introducción

El procesamiento digital de imágenes hace referencia a la técnica de procesar imágenes del mundo real por medio de métodos ópticos o digitales, sus inicios se remontan a la década de los años 60's y está relacionada directamente con el desarrollo y evolución de las computadoras.

Fue en 1959, en el laboratorio Jet Propulsion donde se comenzó a realizar procesamiento de imágenes, con el propósito de mejorar las imágenes enviadas por los satélites.

Actualmente el desarrollo de diversos lenguajes de programación ha hecho posible el crecimiento de aplicaciones para el procesamiento de imágenes, tales como:

- Imágenes satelitales
- Astronómicas
- Geográficas
- Médicas
- Biológicas

Las imágenes digitales están formadas por píxeles, y éstos al encontrarse muy juntos forman la imagen.

El píxel es la unidad mínima de información de una imagen digital, cada píxel de una imagen almacena información de su luminosidad, donde el color negro es el valor 0 y el blanco el valor más alto (255 en escala de grises) pero en formato binario.

En la herramienta de procesamiento Matlab, una imagen a escala de grises es representada por una matriz bidimensional de m (representa el número de píxeles de ancho) por n (número de píxeles de largo).

1.1 Objetivos

- Con fundamentos probabilísticos, definir un esquema para el procesamiento de datos de radar de apertura sintética.
- Diseñar y aplicar un esquema estocástico de segmentación para la detección de derrames petroleros.

1.2 Fundamentación

La misión del satélite Envisat consistió en recopilar datos y analizarlos para evaluar el grado de contaminación atmosférica y calentamiento global.

Las imágenes fueron capturadas por un instrumento del satélite Envisat llamado “Radar de Apertura Sintética Avanzada (ASAR)”, con estas imágenes se han podido localizar diversas áreas de gran extensión de vertido de combustibles (petróleo y aceites) en costas y mares. Estas imágenes son obtenidas en escala de grises, en donde se puede apreciar la oscura estela a lo largo de la superficie oceánica.

Gracias al procesamiento de imágenes se puede calcular la extensión del combustible derramado (Km^2), de esta manera prever las consecuencias catastróficas, y realizar la remoción del mismo, ya que afecta la vida de los ecosistemas marinos.

El fundamento teórico reside en la teoría de Bayes y la maximización de probabilidades a posteriori.

Un aspecto contextual de análisis es abordado mediante los procesos estocásticos y la teoría de los campos aleatorios de Markov. Como aplicación, se pretende la detección de zonas del derrame petrolero provocado por la plataforma petrolera Deepwater Horizon en el Golfo de México, ocurrido en la primavera de 2010. Se cuenta con un banco de imágenes del satélite ENVISAT/ASAR de alta resolución.

1.3 Definición del problema

En el presente trabajo de tesis, se pretende abordar el estudio de técnicas estocásticas para la segmentación de imágenes adquiridas por el satélite ENVISAT. A pesar de las grandes ventajas en el empleo de instrumentos activos de microondas, diversas problemáticas limitan el análisis de los datos de percepción remota adquiridos por estos medios. En el caso de fenómenos producidos en superficies marítimas, las interacciones de la atmósfera con la capa superficial del agua y la degradación producida por señales altamente coherentes¹ dificultan enormemente el empleo y la aplicación de las señales de radar. El material biológico también puede provocar las falsas detecciones llamadas *look-alikes*², por lo que el procesamiento de datos de radar es aún un campo con enorme potencial de estudio.

1.4 Aportación de la tesis

En datos de alta resolución, un problema a resolver consiste en la definición de un modelo que aporte resultados precisos en las zonas de transición. En base a indicadores matemáticos y de probabilidad, se plantea:

- a) La definición de un esquema estocástico binario tipo máximo a posteriori.
- b) La definición de un esquema adaptativo, capaz de realizar detecciones en datos degradados por el speckle³.

¹Si se habla de señales coherentes quiere decir que existe correlación entre las fuentes, es decir que presentan las frecuencias con la misma fase.

²Falsas detecciones, comúnmente llamados dobles, producidos el mayor de los casos por materiales orgánicos.

³Interferencia mutua entre frentes de onda coherentes que son sujetos a diferencias de fase

2. Probabilidad y procesos estocásticos

El concepto de probabilidad resalta cuando nos encontramos con un evento aleatorio. Existen tres enfoques de la probabilidad:

- Interpretación clásica⁴:

$$P(B) = \frac{n(B)}{N}$$

Dónde:

$P(B)$ = probabilidad de que ocurra el evento B.

$n(B)$ = número de resultados favorables para el evento B.

N = número total de resultados posibles.

- Interpretación frecuentista⁵:

$$P(B) = \lim_{n \rightarrow \infty} \left(\frac{n(B)}{N} \right)$$

Consiste en definir la probabilidad como el límite cuando “n” tiende a infinito de la frecuencia relativa del suceso.

- Interpretación subjetiva:

Es una medida entre 0 y 1 que se asocia de manera subjetiva.

Existen situaciones en donde el experimento cambia con respecto del tiempo, y aquí entra el papel de los procesos estocásticos.

Un proceso estocástico es una colección de variables aleatorias $\{X_t : t \in T\}$ parametrizada por un conjunto T, llamado espacio parametrial, en donde las variables toman valores en un conjunto S, llamado espacio de estados [1].

Algunas aplicaciones de los procesos estocásticos:

- Señales aleatorias de Telecomunicaciones.
- Señales biomédicas.
- Número de manchas solares año tras año.
- Señales sísmicas.

⁴Algunos inconvenientes del uso de este enfoque es que es necesario que todos los resultados del experimento tengan la misma probabilidad. ⁵No es aplicable cuando el evento no se puede repetir un gran número de veces

2.1 Axiomas de probabilidad

La teoría de probabilidad encuentra sus bases en los axiomas de probabilidad, y de ellos se pueden establecer teoremas necesarios para desarrollar una teoría axiomática de probabilidad.

Se llama probabilidad axiomática a la función numérica P , cuyo dominio es A y rango el intervalo $[0,1]$ tal que los valores $P(B)$ en A , cumplen con los siguientes tres axiomas, llamados axiomas de Kolmogorov para familias finitas:

1.- Para cualquier evento B , $P(B) \geq 0$.

2.- $P(S) = 1$ (probabilidad del espacio muestral).

3.1- Si B_1, B_2, \dots, B_k es una colección finita de eventos mutuamente excluyentes, entonces:

$$P(B_1 \cup B_2 \cup \dots \cup B_k) = \sum_{i=1}^k P(B_i).$$

3.2- Si B_1, B_2, B_3, \dots es una colección infinita de eventos mutuamente excluyentes entonces:

$$P(B_1 \cup B_2 \cup B_3 \cup \dots) = \sum_{i=1}^{\infty} P(B_i) \quad (\text{la unión de los eventos es igual a la suma de sus probabilidades})[2].$$

La gran importancia de los axiomas reside en que sientan las bases que debe cumplir una función de probabilidad.

2.2 Variable aleatoria y funciones de probabilidad

El concepto de variable aleatoria proporciona un medio para relacionar cualquier resultado con una medida cuantitativa.

Para un determinado espacio muestral S de un experimento, una variable aleatoria es cualquier regla que relaciona un número con cada resultado en S . En el lenguaje matemático, una variable aleatoria es una cuyo dominio es el espacio muestral y cuyo recorrido es el conjunto de los números reales [3].

Las variables aleatorias se suelen denotar con letras mayúsculas del final del alfabeto como X, Y .

Por ejemplo en la notación $X(s)=x$, significa que “ x ” es el valor relacionado con el resultado “ s ” mediante la variable aleatoria “ X ”, en el caso de la notación de la función de probabilidad ésta quedará como $P(X=0)$, $P(X=1)$, y así sucesivamente, dependiendo del número de variables aleatorias [3].

Existen dos tipos variables aleatorias (como se observa en la figura 1):

- Se dice que una variable aleatoria X es discreta si el número de valores que puede tomar es contable y si éstos pueden arreglarse en una secuencia que corresponde con los enteros positivos.

- Se dice que una variable aleatoria X es continua si sus valores consisten en uno o más intervalos de la recta de los números reales [2].

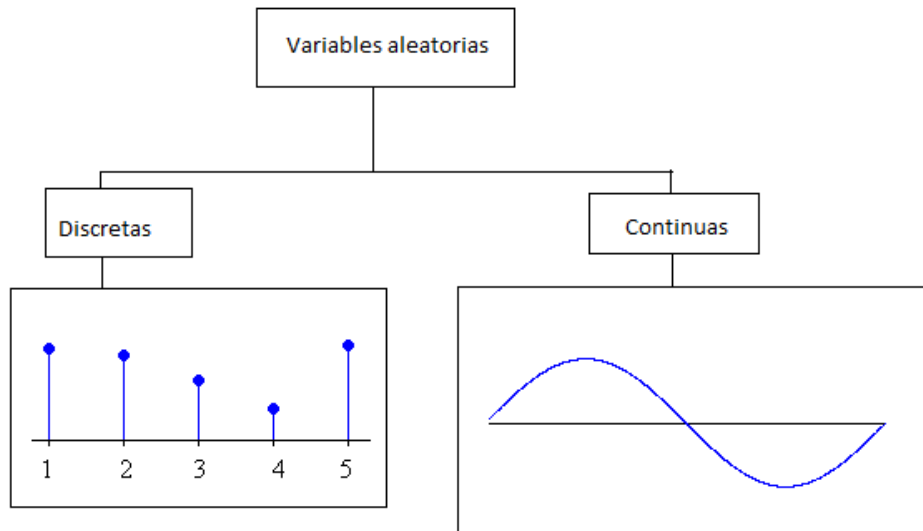


Figura 1: Variables aleatorias

Función de distribución

La función de distribución, también llamada distribución acumulada, muestra el comportamiento acumulado de una variable aleatoria.

Propiedades:

- La función de distribución es creciente.
- Es continua por la derecha⁶.

Si X es una variable aleatoria, entonces su función de distribución $F_x(x)$, se define como una función que asocia a cada valor real, la probabilidad de que la variable aleatoria asuma valores menores o iguales que ella.

$$F_x(x) = P(X \leq x)$$

$$F_x: \mathbb{R} \rightarrow [0,1]$$

⁶Existencia del límite de la función de distribución, cuando x tiende a un valor por la derecha.

Sea X una variable aleatoria continua, su función de distribución quedará definida por:

$$F_x(x) = P(X \leq x) = \int_{-\infty}^x f(x) dx$$

Es decir la función de distribución de probabilidad es la integral de la función densidad⁷.

Propiedades de la función de densidad:

- a) $f_x(x) \geq 0$ para todo x .
- b) $\int_{-\infty}^{+\infty} f_x(x) dx = 1$
- c) $P[a \leq x \leq b] = \int_a^b f(x) dx = F(b) - F(a)$

2.3 Distribución gaussiana

La distribución normal o gaussiana es, con mucho, la distribución de probabilidad de variable aleatoria continua más importante, existen tres razones para considerarla así:

1. Por sus propiedades matemáticas⁸.
2. Diversos experimentos físicos frecuentemente tienen distribuciones que son aproximadamente gaussianas.
3. Por el teorema central del límite⁹.

Se dice que una variable aleatoria continua X tiene una distribución normal con parámetros μ y σ (ó μ y σ^2), donde $-\infty < x < \infty$ y $0 < \sigma$, si la función de probabilidad es [3]:

$$f(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{\left(-\frac{1}{2}\left[\frac{x-\mu}{\sigma}\right]^2\right)} \quad -\infty < x < \infty$$

⁷Corresponde a la función que proporciona información sobre la probabilidad de la variable aleatoria continua.

⁸Su gráfica es simétrica respecto de su media; la moda y la mediana son ambas iguales a la media.

⁹dice que si tenemos un grupo numeroso de variables independientes y todas ellas siguen el mismo modelo de distribución (cualquiera que éste sea), la suma de ellas se distribuye según una distribución normal.

Propiedades de la distribución gaussiana

a) Verifico que $f(x)$ es una función de probabilidad.

Recordando que la función de probabilidad debe cumplir:

$$\int_{-\infty}^{+\infty} f(x) dx = 1, \text{ entonces:}$$

$$\int_{-\infty}^{+\infty} f(x; \mu, \sigma) dx = 1,$$

$$\int_{-\infty}^{+\infty} f(x; \mu, \sigma) dx = \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{\left(-\frac{1}{2}\left[\frac{x-\mu}{\sigma}\right]^2\right)} dx$$

Si defino $y = \frac{x-\mu}{\sigma}$, entonces:

$$R = \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{\left(\frac{y^2}{2}\right)} dy \quad \dots\dots (1)$$

Y también:

$$R = \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{\left(\frac{z^2}{2}\right)} dz \quad \dots\dots (2)$$

Multiplicamos $R \cdot R$ para poder resolver la integral en polares, utilizando R^2 , siendo la multiplicación de las expresiones (1) y (2):

$$R^2 = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{\left(\frac{y^2}{2}\right)} dy \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{\left(\frac{z^2}{2}\right)} dz$$

$$R^2 = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} e^{\left(\frac{-(y^2+z^2)}{2}\right)} dydz$$

Introduciré coordenadas polares para evaluar la integral doble:

$$D = \{(y,z) \mid -\infty \leq y \leq +\infty, -\infty \leq z \leq +\infty\}$$

$$D = \{(r, \theta) \mid 0 \leq r \leq \infty, 0 \leq \theta \leq 2\pi\}$$

$$R^2 = \frac{1}{2\pi} \int_0^{2\pi} \int_0^{\infty} r e^{\left(\frac{-r^2}{2}\right)} dr d\theta$$

Puedo observar que al resolver la transformación de coordenadas rectangulares a polares $dydz$ cambió por $rdrd\theta$.

Resolviendo:

$$R^2 = \frac{1}{2\pi} \int_0^{2\pi} -r e^{\left(\frac{-r^2}{2}\right)} \Big|_0^{\infty} d\theta$$

$$R^2 = \frac{1}{2\pi} \int_0^{2\pi} d\theta = 1$$

Por lo tanto $R = 1$.

Con lo cual queda demostrado que la distribución gaussiana es una función de probabilidad.

b) Forma gráfica de la función:

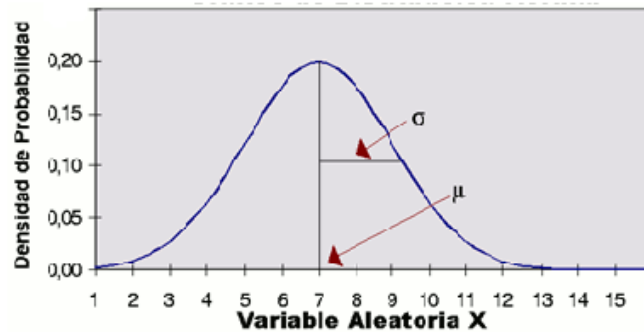


Figura 2: Gráfica de una distribución gaussiana

Observado la figura 2, la función depende sólo de x mediante la expresión $(x - \mu)^2$ y por tanto la gráfica de la función es simétrica respecto a μ . Ahora de $f_x(x) \geq 0$, para valores grandes de x , la gráfica es cóncava hacia arriba. Para conocer el punto de inflexión (punto donde cambia la concavidad) éste se determina al resolver la ecuación $f''(x)$.

El punto de inflexión ocurre en $x = \mu \pm \sigma$.

c) La esperanza y variancia de X son parámetros que caracterizan la distribución normal.

$$E(X) = \mu$$

$$V(X) = E(X^2) - (E(X))^2 = \sigma^2$$

d) Si la variable aleatoria continua tiene una distribución con valores de parámetros $\mu = 0$ y $\sigma = 1$ entonces la variable aleatoria tiene una distribución normal estandarizada, su función de probabilidad sería:

$$f(z) = \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}} \quad -\infty < z < \infty$$

Tabulación de la distribución normal estándar

$$P(a \leq Z \leq b) = \frac{1}{\sqrt{2\pi}} \int_a^b e^{-\frac{z^2}{2}} dz$$

En esta integral no se puede aplicar el teorema fundamental del cálculo, ya que no existe una función cuya derivada sea igual a $e^{-\frac{z^2}{2}}$, a pesar de eso, existen métodos de integración numéricos que pueden usarse para evaluar integrales como la anterior.

La función de probabilidad normal estandarizada se denotará con Φ .

$$\Phi(s) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^s e^{-\frac{z^2}{2}} dz$$

$$P(a \leq Z \leq b) = \Phi(b) - \Phi(a)$$

Puedo usar tablas para evaluar de la función Φ , $P(a \leq Z \leq b)$.

| z | .00 | .01 | .02 | .03 | .04 | .05 | .06 | .07 | .08 | .09 |
|-----|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 0.0 | .0000 | .0040 | .0080 | .0120 | .0160 | .0199 | .0239 | .0279 | .0319 | .0359 |
| 0.1 | .0398 | .0438 | .0478 | .0517 | .0557 | .0596 | .0636 | .0675 | .0714 | .0753 |
| 0.2 | .0793 | .0832 | .0871 | .0910 | .0948 | .0987 | .1026 | .1064 | .1103 | .1141 |
| 0.3 | .1179 | .1217 | .1255 | .1293 | .1331 | .1368 | .1406 | .1443 | .1480 | .1517 |
| 0.4 | .1554 | .1591 | .1628 | .1664 | .1700 | .1736 | .1772 | .1808 | .1844 | .1879 |
| 0.5 | .1915 | .1950 | .1985 | .2019 | .2054 | .2088 | .2123 | .2157 | .2190 | .2224 |
| 0.6 | .2257 | .2291 | .2324 | .2357 | .2389 | .2422 | .2454 | .2486 | .2517 | .2549 |
| 0.7 | .2580 | .2611 | .2642 | .2673 | .2704 | .2734 | .2764 | .2794 | .2823 | .2852 |
| 0.8 | .2881 | .2910 | .2939 | .2967 | .2995 | .3023 | .3051 | .3078 | .3106 | .3133 |
| 0.9 | .3159 | .3186 | .3212 | .3238 | .3264 | .3289 | .3315 | .3340 | .3365 | .3389 |
| 1.0 | .3413 | .3438 | .3461 | .3485 | .3508 | .3531 | .3554 | .3577 | .3599 | .3621 |
| 1.1 | .3643 | .3665 | .3686 | .3708 | .3729 | .3749 | .3770 | .3790 | .3810 | .3830 |
| 1.2 | .3849 | .3869 | .3888 | .3907 | .3925 | .3944 | .3962 | .3980 | .3997 | .4015 |

Tabla 1: Distribución de probabilidad normal estándar

2.4 La ecuación de Bayes

La estructura de la ecuación de Bayes permite el cálculo de probabilidades después de haber sido realizado el experimento (a posteriori).

Este teorema es consecuencia de la definición probabilidad del suceso intersección y del concepto de probabilidad condicional. Así, teniendo en cuenta las fórmulas de la probabilidad condicional, tanto de A sobre B, como de B sobre A.

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

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

Al despejar la probabilidad del suceso intersección:

$$P(A \cap B) = P(B \cap A) = P(A|B) P(B) = P(B|A) P(A)$$

Despejando la última igualdad:

$$P(A|B) = \frac{P(B|A) P(A)}{P(B)}$$

Donde:

A = suceso condicionado y B = suceso condicionante.

Aplicando en el denominador el Teorema de partición:

$$P(B) = P(A_1)P(B|A_1) + P(A_2)P(B|A_2) + \dots + P(A_n)P(B|A_n)$$

La ecuación de Bayes queda:

$$P(A_1|B) = \frac{P(A_1)P(B|A_1)}{\sum_{i=1}^n P(A_i)P(B|A_i)}$$

3. Campos aleatorios de Markov

3.1 Procesos aleatorios en la percepción remota

La percepción remota permite la observación y obtención de información de nuestro planeta, además engloba los procesos que permiten obtener una imagen y su posterior tratamiento.

Elementos de un sistema de percepción remota (figura 3):

- Fuente de energía que emite la radiación electromagnética que detecta el sensor.
- Cubierta Terrestre (vegetación, suelo, agua, edificios o construcciones), éstas reciben la señal energética y la reflejan de acuerdo a sus características físicas.
- Barredor multiespectral, tiene como misión captar la energía procedente de la cubierta terrestre.
- Sistema de recepción de la información.

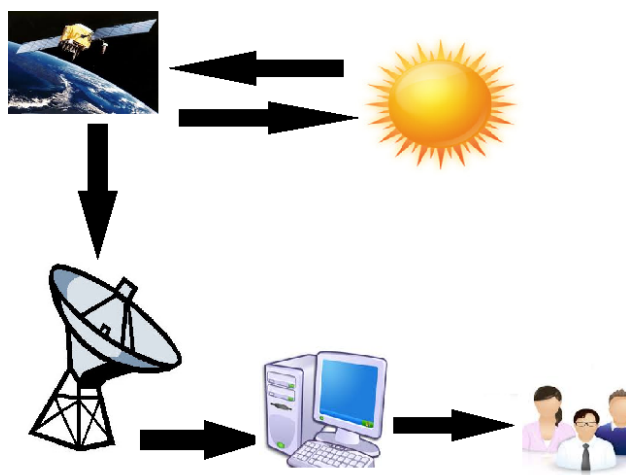


Figura 3: Componentes de un sistema de percepción remota

Existen diversas regiones de difícil acceso para la adquisición de datos de percepción remota, ya sea por condiciones meteorológicas o por ciertos problemas de iluminación en las imágenes.

Es aquí donde intervienen los campos aleatorios, éstos se definen como procesos estocásticos; el caso especial son los campos aleatorios de Markov, los cuales tienen éxito en la restauración de imágenes, segmentación, reconstrucción de superficies, análisis de texturas y en detección de bordes.

3.2 Topologías

Las imágenes desde el punto de vista estocástico pueden modelarse como muestras de un proceso aleatorio bidimensional.

$$Y = \{i, j \mid i, j = 0 \dots N-1\}$$

Donde Y representa el conjunto de posiciones en la imagen, también se tiene que:

$$X = \{X_{i,j} \mid i, j = 0 \dots N-1\}$$

Donde X representa el conjunto de todas las variables aleatorias del campo $N \times N$.

Las topologías son las formas en que se pueden representar las imágenes para su mejor comprensión, existen dos topologías: modelo de textura y modelo de contorno.

Modelo de textura (2D): representa el conjunto de píxeles en una estructura matricial, como se muestra en la figura 4, éste conjunto es llamado conjunto "S" o de espacio de estados para modelo de textura.

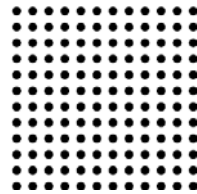


Figura 4: Malla para textura

Modelo de contorno (1D): representa cada uno de los radios trazados desde el centro en una estructura inicial¹⁰ como se muestra en la figura 5, éste conjunto es llamado conjunto "S" o de espacio de estados para modelo de contorno.

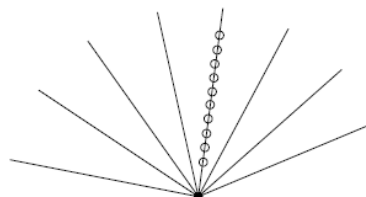


Figura 5: Malla para contorno

¹⁰En sí mismo este modelo no existe en la realidad, solo corresponde a otra forma de representación de las variables aleatorias del campo.

Para cada posición $s \in S$ se va a definir un espacio de estados:

1. Para el modelo de textura corresponde a los niveles de gris, donde "s" es un pixel.
2. Para el modelo de contorno, "s" es un radio y el espacio de estados para cada radio corresponde a cada uno de los puntos donde puede pasar el contorno (puntos de la figura 6).

Así el espacio de todas las configuraciones será el conjunto de todas las texturas o contornos¹¹ posibles que se puedan definir en el conjunto S.

Un espacio es homogéneo si todos los pixeles tienen el mismo rango de niveles de gris (textura) y el mismo rango de posiciones para todos los radios en el caso del modelo de contorno.

La segmentación de imagen es un proceso muy importante que se basa en dividir una escena en un conjunto de regiones y es el primer nivel en la interpretación de imágenes.

Para entender mejor el problema de segmentación de una imagen se considera como un problema de etiquetado.

Para el problema del etiquetado se asigna una etiqueta del conjunto L de etiquetas a cada estado de S y la solución consiste en la asignación de etiquetas a los pixeles de la imagen.

Los campos aleatorios de Markov (CAM) no son un método de segmentación, pero sí un modelo estadístico que puede ser usado dentro de los modelos de segmentación, los CAM modelan las interacciones espaciales entre pixeles vecinos.

¹¹Sí el modelo de textura y el modelo de contorno son campos aleatorios, entonces todas las configuraciones y contornos van a ser posibles.

3.2.1 Vecindades

Las vecindades corresponden al conjunto de pixeles aledaños a un pixel central.

La región donde se define el campo aleatorio se denomina “rejilla o malla” y se denota como el conjunto S .

Los estados¹² de S están relacionados mediante un sistema de vecinos $N = \{N_i \mid \forall i \in S\}$, de donde: N_i es el conjunto de los estados vecinos.

Para una malla regular S , el conjunto de vecinos está definido como:

$$N_i = \{j \in S \mid [\text{dist}(\text{pixel}_j, \text{pixel}_i)]^2 \leq r, j \neq i\} \quad [4]$$

Donde $\text{dist}(\text{pixel}_j, \text{pixel}_i)$ corresponde a la distancia euclidiana entre los dos puntos y r es un número entero.

Tenemos $S = \{1, \dots, m\}$, conjunto ordenado de estados, en los que sus elementos son los índices de los pixeles de una imagen en 1D, un estado interior $i \in \{2, \dots, m-1\}$ tiene dos vecinos más cercanos $N_i = \{i-1, i+1\}$, y los estados de la frontera¹³ (los dos extremos) tienen uno, esto se puede observar al analizar la figura 6..

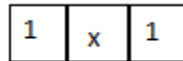


Figura 6: Vecindario de orden 1 para el modelo de contorno.

Propiedades de los vecinos:

- a. Un estado no puede ser vecino de sí mismo.
- b. La relación de vecindad es totalmente mutua.

Para los pixeles de una imagen en dos dimensiones $M \times N$ (modelo de textura), un estado interno (i,j) tiene cuatro vecinos más cercanos (sistema de primer orden) $N_{i,j} = \{(i-1,j), (i+1,j), (i,j-1), (i,j+1)\}$, los estados en la frontera tres y los de las esquinas dos.

¹²De éstos las variables toman sus niveles de gris

¹³Los pixeles de las fronteras tienen menos vecinos.

Sistema de primer orden vecinal

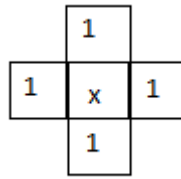


Figura 7: Sistema 4- vecinos (modelo de textura).

Observamos que cada estado interior (x) tiene 4 vecinos de acuerdo a la figura 7.

$$L = \{(i,j): 1 \leq i \leq M, 1 \leq j \leq N\}$$

El alfabeto de la imagen consiste en un conjunto de 256 niveles de gris y se denota desde 0 hasta 255.

El sistema de vecinos en el modelo de textura quedará definido como “ δ ”, en la figura 8 observamos que el sistema de vecinos de x serán:

$$\delta_{i,j}^1 = \{x_{(i-1,j)}, x_{(i,j-1)}, x_{(i+1,j)}, x_{(i,j+1)}\}$$

Que corresponden a los vecinos de lado horizontal y vertical.
Para los vecinos en la frontera (0 y N-1).

Sistemas de vecinos de segundo orden:

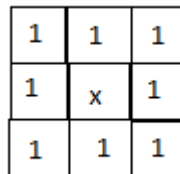


Figura 8: Sistema 8-vecinos (modelo de textura)

En este sistema observamos que cada estado interior (x) tiene 8 vecinos como se muestra en la figura 8.

$$\delta_{i,j}^2 = \{x_{(i-1,j)}, x_{(i,j-1)}, x_{(i+1,j)}, x_{(i,j+1)}, x_{(i-1,j-1)}, x_{(i-1,j+1)}, x_{(i+1,j-1)}, x_{(i+1,j+1)}\}$$

3.2.2 Cliques

Un subconjunto $C \subseteq L^{14}$ es llamado un clique si C consiste de un único pixel o un grupo de píxeles semejantes en un área vecinal [4].

Un clique se define como un subconjunto de S tal que C consiste en un único estado: $C = \{i\}$ o un par de estados vecinos $C = \{i,j\}$, o tres estados vecinos $\{i,j,k\}$; el conjunto de cliques de un solo estado se denota como:

$$C_1 = \{i \mid i \in S\}.$$

Conjunto de cliques de un par de estados:

$$C_2 = \{(i,j) \mid j \in N_i, i \in S\}$$

Para el caso de un sistema de vecinos de primer orden, los cliques para éste sistema serían todas las variables consideradas como conjuntos con un solo elemento y todos los conjuntos de dos variables cuyas ubicaciones en el campo sean contiguas en sentido vertical u horizontal. En la figura 9 y 10 observamos los cliques para un modelo de textura y contorno para sistemas de orden 1.

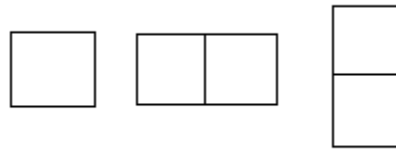


Figura 9: Tipos de cliques para sistemas de orden 1 (modelo de textura)



Figura 10: Tipos de cliques para el vecindario de orden 1 (modelo de contorno).

¹⁴ $A \subseteq B$ significa: cada elemento de A es también elemento de B

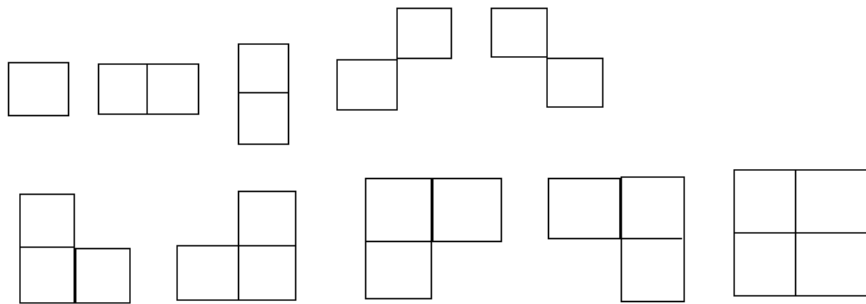


Figura 11: Tipos de cliques para sistemas de orden 2 en el modelo de textura.

3.2.3 Propiedades de Markov

Se dice que un campo aleatorio es de Markov si y sólo sí:

1.- $P(x) > 0$ – positivo.

2.- $p(x_{i,j} = i_{i,j} \mid x_{k,l} = i_{k,l} \forall (k,l) \neq (i,j)) = p(x_{i,j} \mid \delta_{i,j})$, donde $\delta_{i,j}$ son los vecinos de $x_{i,j}$.¹⁵ [5]

Un campo aleatorio de Markov es un campo aleatorio con la propiedad de localidad¹⁶.

Campos aleatorios de Gibbs

Existe otro enfoque para modelar un campo aleatorio, y éste es haciendo uso de las leyes de la Mecánica Estadística, la función de probabilidad que resulta en este caso es la distribución de Gibbs.

Supongamos que el sistema físico bidimensional es la imagen, y las partículas de ese sistema son los pixeles, por consiguiente los posibles estados vendrían siendo los niveles de gris de cada posición de la imagen.

Se define como potencial U sobre el campo aleatorio a la familia de funciones $\{U_A \text{ con } A \subset X\}$ ¹⁷.

Cada U_A toma un valor real y depende solamente de las variables incluidas en A .

La función de energía que induce el campo es [5]:

$$H_u = \sum_{A \subset X} U_A$$

¹⁵En este caso se está hablando del caso discreto, por lo tanto la ecuación 2, está escrita en términos de su función de masa de probabilidad.

¹⁶Probabilidades condicionales, segunda propiedad de los campos aleatorios de Markov.

Un campo aleatorio de Gibbs es un conjunto de variables aleatorias F , en el que sus configuraciones obedecen a una distribución de Gibbs.

Entonces un campo aleatorio de Gibbs con potencial U será aquel con probabilidad:

$$P(x) = \frac{1}{z} e^{-H_u} \quad [5]$$

Debido al tamaño del campo aleatorio, se trabajará en el dominio de las características locales. Una distribución de Gibbs es una probabilidad medida al atributo de un pixel en particular.

Cuando el potencial U es tal que $U_A = 0$, se dice que es un potencial de vecinos y el campo aleatorio será de Gibbs con potencial de vecinos, su probabilidad será:

$$P(x) = \frac{1}{z} e^{-\frac{1}{T} \sum_{c \subset C} U_c} \quad [5]$$

Dónde:

U_c = Energía asociada con el clique c .

T = Temperatura.

z = Constante de normalización.

z se define por:

$$z = \sum e^{-\frac{1}{T} \sum_{c \subset C} U_c}$$

Podemos observar que z debe ser una constante normalizada para que se cumpla:

$$\sum P(x) = 1$$

Equivalencia Markov- Gibbs

La equivalencia entre los campos aleatorios de Markov y los campos aleatorios de Gibbs establece el teorema de Hammersley – Clifford.

Teorema Hammersley – Clifford: F es un campo aleatorio de Markov en S con respecto a N si y sólo si F es un campo aleatorio de Gibbs con respecto a N .

Entonces otra propiedad de los campos aleatorios de Markov es que bajo una condición, el campo aleatorio puede tener una distribución de Gibbs.

¹⁷ $A \subset B$ significa: $A \subseteq B$ pero $A \neq B$

3.3 Modelos de Markov

Un modelo puede definirse como un esquema teórico, generalmente en forma matemática, de un sistema o de una realidad compleja, que se elabora para facilitar su comprensión y el estudio de su comportamiento.

El modelo estocástico más frecuentemente utilizado en el ámbito probabilístico es el denominado modelo de Markov.

Es importante el estudio de algunos modelos de campos aleatorios de Markov, debido a que permiten modelar características importantes de las imágenes como:

- Tipos de región
- Textura

Es importante en el estudio de los campos aleatorios de Markov, la probabilidad condicional, así como sus funciones de energía.

A continuación se presentan algunos potenciales utilizados para modelar imágenes.

3.3.1 Modelo de Ising

Este modelo es un caso particular de un campo aleatorio de Markov, tuvo sus orígenes en la física para explicar el comportamiento de materiales con características ferromagnéticas.

En este modelo se considera un campo aleatorio de Markov binario, donde las variables pueden asumir dos estados opuestos $\delta_i = 1$ (spin up) y $\delta_i = -1$ (spin down) donde δ_i ($i = 1, 2, 3 \dots M$), además el sistema es un arreglo de M puntos fijos y a estos se les asigna una de las dos variables anteriormente descritas, como se puede observar en la figura 12.

La energía del sistema $H_u\{\delta_i\}$ está dada por el estado de los espines y se define como:

$$H_u\{\delta_i\} = - \sum_{\langle i, j \rangle} \epsilon_{i,j} \delta_i \delta_j - H \sum_{i=1}^M \delta_i \quad [5]$$

Donde:

$\langle i j \rangle$ representa pares de espines de vecinos de primer orden (vecinos más cercanos).

$\epsilon_{i,j}$ representa la energía de interacción entre los espines δ_i y δ_j .

H es el campo magnético externo que induce a los espines a polarizarse.

Si:

H > 0, los espines se polarizan positivamente.

H < 0, los espines se polarizan negativamente.

La probabilidad de la configuración se calcula con ayuda de la distribución de Gibbs:

$$P(\delta_i) = \frac{1}{Z} e^{-\frac{1}{kT} \text{Hu}\{\delta_i\}}$$

Donde:

Z: constante de normalización.

$K = 1.3806504 \times 10^{-23} \frac{J}{K}$ Constante de Boltzmann

T: Temperatura del sistema.

En ausencia de un campo magnético externo:

$$P(\delta_i) = \frac{1}{Z} e^{-\beta \sum_{\langle i j \rangle} \{\delta_i \delta_j\}}$$

Se consideran interacciones de vecinos de primer orden, donde $\epsilon_{i,j} = 1$, y con $\beta = \frac{1}{kT}$, donde beta regula las interacciones entre pixeles, este modelo permite representar imágenes en blanco y negro.

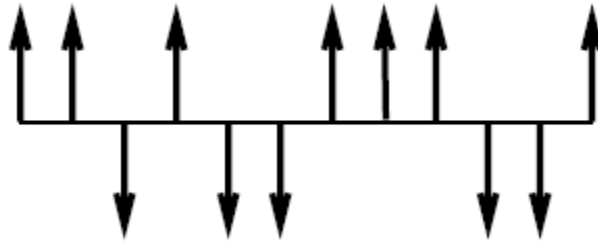


Figura 12: Modelo de Ising en una dimensión.

3.3.2 Modelo de Potts

En las imágenes de niveles de gris es posible modelar las interacciones de pixeles a efecto de segmentar en varias clases.

Este modelo es una extensión del modelo de Ising para más de dos estados por pixel, observar la figura 13.

Considera al modelo de Ising como un sistema de espines interactuando, y pueden ser paralelos o antiparalelos.

Consideremos un sistema de espines, cada espin señalando en una de las direcciones “q” equidistantes del centro. Estos vectores son la combinación lineal de vectores unitarios que apuntan en las direcciones simétricas de un hipertetrahedro en q-1 dimensiones [6].

La energía del modelo de Potts viene dada por:

$$H_u\{\delta_i\} = - \sum_{\langle i,j \rangle} \epsilon_{i,j} \theta_{i,j}$$

Donde:

$\epsilon_{i,j}$: es periódica y $\theta_{i,j}$ es el ángulo entre los espines vecinos en i y j.

El caso q = 2 es equivalente al modelo de Ising [6].

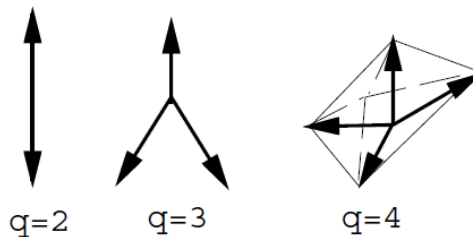


Figura 13: Modelo de Potts.

4. El satélite ENVISAT

El satélite Envisat (*Environmental Satellite*) es un satélite de observación terrestre construido por la Agencia Espacial Europea (ESA). Fue lanzado el 1 de marzo de 2002 en un cohete Ariane 5 en una órbita polar síncrona¹⁸ con el Sol a una altura de 790 km. Orbita la Tierra con un periodo de 101 minutos, con un periodo de repetición de ciclos cada 35 días. Estuvo operativo hasta el día 8 de abril de 2012, día en el que se perdieron todas las comunicaciones con el satélite.

Envisat fue uno de los mayores observatorios de la atmósfera y superficie terrestre lanzados, tiene una masa de 8211 Kg, y dispone de 10 instrumentos para la obtención de información sobre la superficie de la Tierra, los océanos y la atmósfera [7].

4.1 Objetivos

La misión de observación del planeta Tierra apoyó trabajos de teledetección terrestre, que fueron esenciales para investigación climatológica y ambiental. ENVISAT supervisó:

- La tierra.
- Océanos.
- Atmósfera.
- Casquetes polares.

4.2 Parámetros orbitales

La órbita del satélite ENVISAT es del tipo *polar circular heliosíncrona*, en la cual los parámetros de movimiento del satélite están calculados de forma que la superficie terrestre barrida por el mismo se encuentre siempre bajo la misma iluminación solar. Dicho con otras palabras, las posiciones relativas del satélite y del Sol se mantienen constantes, con lo cual se consigue que las fotografías obtenidas lo sean siempre en las mismas condiciones.

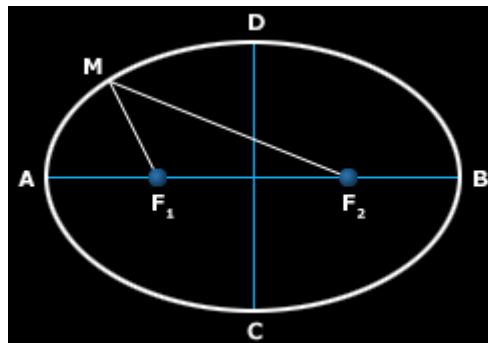


Figura 14: Representación gráfica para el cálculo de la excentricidad.

¹⁸Es una órbita en la que el periodo orbital del cuerpo orbitando (satélite) es igual al periodo de rotación del cuerpo en el cual orbita (planeta) y, además, orbita en el mismo sentido que este cuerpo.

La excentricidad es el grado de aplastamiento de la elipse. Una excentricidad igual a cero representa un círculo perfecto. Cuanto más grande la excentricidad, mayor el aplastamiento de la elipse. Órbitas con excentricidades iguales a uno se denominan parabólicas, y mayores a uno hiperbólicas.

La excentricidad de la elipse puede calcularse de la siguiente manera (observar figura 14):

$$e = \frac{(F_1F_2)}{AB}$$

Donde “e” es la excentricidad, F_1F_2 es la distancia entre los focos y AB es el eje mayor de la elipse. Si la distancia entre los focos F_1F_2 es cero, como en el caso del círculo, la excentricidad da como resultado cero.

La excentricidad de la órbita del satélite Envisat es de 0,001165.

Su periodo orbital es de aproximadamente 101 minutos, éste es el tiempo que le toma al satélite recorrer su órbita.

El punto donde la distancia entre el satélite y la Tierra es mínima se le conoce como periastro (observar figura 16) y para Envisat corresponde a 796 Km.

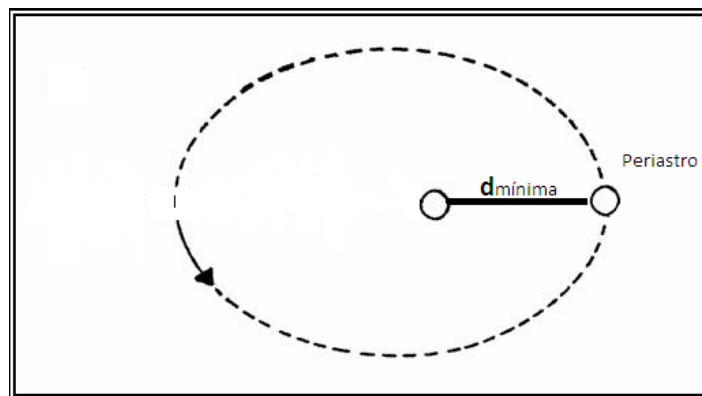


Figura 15: Representación gráfica del periastro.

4.3 Instrumentos

Envisat lleva a bordo diez instrumentos desarrollados por la Agencia Espacial Europea [8], como se muestra en la figura 16:

| ASAR (Advanced Synthetic Aperture Radar) |
|--|
| <p>Descripción: Opera en la banda C, cuenta con gran capacidad en términos de cobertura, gama de ángulos de incidencia, polarización y modos de operación. Esta capacidad mejorada es proporcionada por diferencias significativas en el diseño del instrumento: un arreglo de antenas, equipado con módulos de transmisión/recepción que proporcionan distintos haces de transmisión y recepción.</p> <p>Utilizado para la observación de los océanos, tierras emergidas y casquetes polares con una resolución espacial de 30 a 1000 m.</p> |
| <p>Tipo: Procesamiento de imágenes (Radares de Microondas).</p> |
| <p>Características Técnicas:</p> <ul style="list-style-type: none">• <i>Precisión:</i> Resolución radiométrica¹⁹ en el rango: 1.5-3.5 dB, precisión radiométrica²⁰: 0.65 dB.• <i>Resolución espacial:</i> Modos de polarización de la onda: aprox. 30 m x 30 m, modo de franja ancha: aprox. 150 m x 150 m, modo de seguimiento global: aprox. 1000 m x 1000 m.• <i>Ancho de franja:</i> Imagen y polarización en modos alternos: hasta 100 km, modo de onda: 5 km, y amplia franja y modos de seguimiento global: 400 km o más.• <i>Bandas:</i> Microondas: banda C, con opción de 5 modos de polarización (VV, HH, VV / HH, HV / HH, o VH / VV) |
| MERIS (Medium Resolution Imaging Spectrometer) |
| <p>Descripción: Espectrómetro de imágenes que operan en el rango espectral de reflexión solar. Quince bandas espectrales pueden ser seleccionadas. Matrices CCD lineales proporcionan muestreo espacial en la dirección transversal a la trayectoria, mientras que el movimiento del satélite ofrece escaneo.</p> <p>MERIS está diseñado de modo que puede adquirir datos sobre la Tierra cuando las condiciones de iluminación son las adecuadas. Los 68.5° del campo de vista del instrumento en torno al nadir²¹ cubre un ancho de barrido de 1.150 Kilómetros.</p> <p>Básicamente dedicado a la observación del color del océano, sus objetivos se amplían con estudios atmosféricos y terrestres. Presenta una resolución espacial de 300 y 1200 m, en 15 bandas del espectro, programable en anchura y posición.</p> |
| <p>Tipo: Procesamiento de imágenes (Radiómetros multiespectrales).</p> |

¹⁹El término hace referencia al número de niveles digitales utilizados para expresar los datos recogidos por el sensor.

²⁰Correspondencia entre la temperatura de brillo medida y la real, definiendo a la temperatura de brillo como la intensidad de emisión de un cuerpo en una escala de temperaturas.

Características Técnicas:

- *Precisión:*
Bandas típicas de color del océano.
- *Resolución espacial:*
Océano: 1040 m x 1200 m, Tierra y costa: 260 m x 300 m.
- *Ancho de franja:*
1150 km, cobertura global cada 3 días.
- *Bandas:*
15 bandas seleccionables en todo el rango: 390 nm a 1040 nm (programable ancho de banda entre 2,5 y 30 nm)

RA-2 (Radar Altimeter 2)

Descripción: Es un instrumento para determinar el retraso bidireccional del eco del radar desde la superficie de la Tierra a un nivel muy alto de precisión: menos de un nanosegundo. También mide la potencia y la forma de los impulsos del radar reflejados.

Es un altímetro de radar de impulsos funcionando a la frecuencia principal nominal de 13.575 GHz (banda Ku).

Permite el estudio de la topografía de la superficie de los océanos, tierras emergidas y casquetes polares, con una precisión que podría alcanzar los 5 cm.

Tipo: Altímetro²² de radar.

Características Técnicas:

- *Bandas:*
Microondas: banda C, 13.575 Ghz (banda Ku) y 3,2 GHz (Banda S).

MWR (Micro Wave Radiometer)

Descripción: Su objetivo principal es la medida del vapor del agua de la atmósfera, siendo sus datos además empleados en la corrección de la señal en su recorrido por la troposfera.

Tipo: Procesamiento de imágenes, radiómetros multiespectrales.

Características Técnicas:

- *Resolución espacial:*
Temperatura: 2.6k
- *Resolución espacial:*
20 kilómetros.
- *Ancho de franja:*
20 Kilómetros
- *Bandas:*
Microondas: 23.8 y 36.5 GHz

²¹Intersección entre la vertical del observador y la esfera celeste.

²²Es un instrumento de medición que indica la diferencia de altitud entre el punto donde se encuentra localizado y un punto de referencia.

| LRR (Laser Retro-Reflector) |
|--|
| Descripción: Instrumento pasivo que permite determinar con precisión la órbita del satélite. |
| Tipo: Instrumento óptico pasivo. |
| Características Técnicas: <ul style="list-style-type: none"> • Eficiencia: menor o igual a 0.15 al final de su vida, coeficiente de reflexión: menor o igual a 0.8 al final de su vida, Diámetro: mayor o igual a 20 cm. • <i>Bandas:</i> 350-800 nm optimizado para 532 nm |

| GOMOS (Global Ozone Monitor by Occultation of stars) |
|---|
| Descripción: Es un espectrómetro de resolución media que cubre el rango de longitud de onda de 250 nm a 950 nm. Su objetivo es conocer los procesos que ocurren en la alta atmósfera, permitiendo el estudio de la distribución del ozono durante largos periodos de tiempo a varias latitudes y altitudes, así como la medida de distribución de aerosoles estratosféricos. |
| Tipo: Química Atmosférica. |
| Características Técnicas: <ul style="list-style-type: none"> • <i>Precisión:</i> Relativa exactitud espectral: 0.17 / 0.20 nm de UV VIS (oscuro/ brillante típico) • <i>Ancho de franja:</i> No aplicable. • <i>Bandas:</i> Espectrómetros: UV-Vis: 248-371nm y 387-693 nm, NIR: 750-776 nm y 915-956 nm, fotómetros: 644-705 nm y 466-528nm. |

| MIPAS (Michelson Interferometer for Passive Atmospheric Sounding) |
|---|
| Descripción: Interferómetro de Michelson para el sondeo atmosférico pasivo (MIPAS) es un espectrómetro de transformada de Fourier para la detección de espectros de emisión en la atmósfera media y alta. Opera en un rango de longitudes de onda de 4.15 micras a 14.6 micras. En sí es una sonda atmosférica de gran resolución espectral para observaciones globales de elementos traza atmosféricos. |
| Tipo: Química atmosférica y temperatura. |
| Características Técnicas: <ul style="list-style-type: none"> • <i>Precisión:</i> Precisión radiométrica: 685-970cm-1: 1%, 2410 cm-1: 3% • <i>Resolución espacial:</i> Resolución vertical: 3 kilómetros, 5-150 km, Rango de escaneado vertical, horizontal: 3 kilómetros x 30 kilómetros, la resolución espectral: 0.035 líneas / cm, |

- *Ancho de franja:*
3 x 30 km.
- *Bandas:*
MWIR-TIR (espectrómetro de onda corta infrarroja): entre 4.15 y 14.6 m.

AATSR (Advanced Along Track Scanning Radiometer)

Descripción: Es la más reciente de una serie de instrumentos diseñados principalmente para medir la temperatura superficial del mar. Permite conocer la temperatura superficial del mar y de los continentes, el crecimiento de la vegetación, la humedad del suelo y la composición de las nubes.

Tipo: Procesamiento de imágenes (Radiómetros multiespectrales y múltiple dirección/radiómetros de polarización).

Características Técnicas:

- *Precisión:*
Temperatura superficial del mar: <0,5 K por encima de 0,5 grados x 0,5 grados (latitud / longitud) de área con un 80%, temperatura superficial del terreno cubierto de nubes: 0.1 K (relativa).
- *Resolución espacial:*
IR²³ canales oceánicos: 1 km x 1 kilómetro, canales terrestres visibles: 1 km x 1 kilómetro.
- *Ancho de franja:*
500 km.
- *Bandas:*
VIS - NIR: 0.555, 0.659, 0.865 micrómetros, SWIR: 1,6 micrómetros, MWIR: 3,7 micrómetros, TIR: 10,85, 12 micrómetros

DORIS (Doppler Orbitography and Radiopositioning Integrated by Satellite)

Descripción: La orbitografía Doppler y radio-posicionamiento integrado en el satélite, constituyen un sistema de seguimiento por medio de microondas, que puede ser utilizado para determinar la ubicación precisa del satélite ENVISAT. Sistema de navegación que con una resolución espacial de 3 cm es clave para medidas de altimetría

Tipo: Precisión orbital.

Características Técnicas:

- *Precisión:*
Error orbital alrededor de 1 cm.
- *Resolución espacial:*
5 cm de altitud.

²³El término hace referencia a la emisión de señales infrarrojas.

- *Ancho de franja:*
²⁴FOV: 130 grados
- *Bandas:*
401.25MHz, 2036.25MHz

| SCIAMACHY (Scanning Imaging Absorption Spectrometer for Atmospheric Cartography) |
|--|
| Descripción: Es un espectrómetro de imágenes cuyo objetivo principal es llevar a cabo las mediciones de gases traza ²⁵ en la troposfera y en la estratosfera. La radiación solar transmitida, reflejada y retrodispersada desde la atmósfera se registra a una resolución relativamente alta (0.2 nm a 0.5 nm). |
| Tipo: Química atmosférica. |
| Características Técnicas: <ul style="list-style-type: none"> • <i>Precisión:</i> Radiométrica: menor al 4%. • <i>Resolución espacial:</i> Brazo vertical 3 x 132 km, Nadir horizontal 32 x 215 km • <i>Ancho de franja:</i> 1000 km (máx.) • <i>Bandas:</i> UV-SWIR: 240-314, 309-3405, 394-620, 604-805, 785-1050, 1000-1750, 1940-2040 y 2265-2380 nm |

²⁴ campo de visión de un instrumento sensor o ángulo de apertura.

²⁵ Llamados así, porque se encuentran en muy pequeñas cantidades.

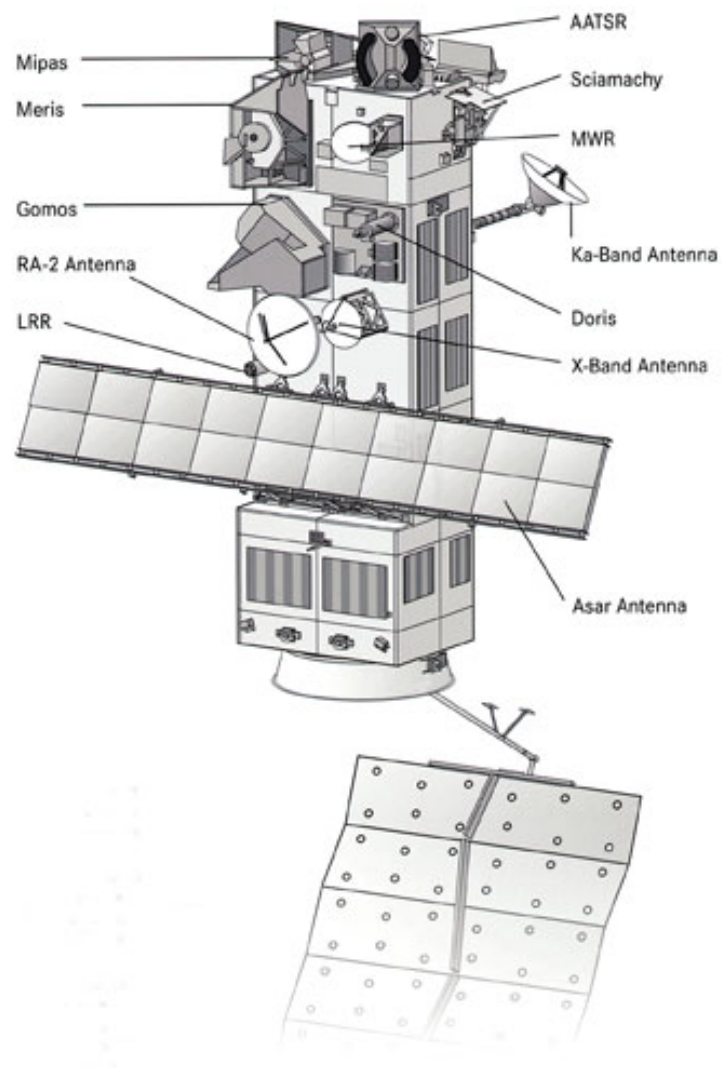


Figura 16: Instrumentos del satélite ENVISAT

4.4 La herramienta NEST®

Nest (Next ESA SAR Toolbox) es una herramienta de código abierto de la Agencia Espacial Europea bajo la licencia GNU GPL (figura 18) para leer, procesar, analizar y visualizar información de los satélites (ERS-1/2, ENVISAT) y otros vehículos espaciales (TerraSAR-X, RADARSAT 1-2, COSMO-SkyMed, JERS-1 Y ALOS PALSAR).

Nest (figura 17) está siendo desarrollado bajo contrato de la Agencia Espacial Europea número 20698/07/LG por Array Systems Computing Inc., PPO. Labs, Delf University of Technology y Brockmann consult.



Figura 17: Logotipo de la herramienta Nest.

Principales características

- Análisis de datos y estadísticas.
- Manejo de metadatos.
- Subconjuntos²⁶ de datos y muestreo.
- Exportación de formatos Geo TIFF, HDF 4 & 5, NetCDF, Bynary, Envi y Kmz.
- LUT²⁷ y gestión de capas.
- Herramienta ROI.
- Calibración absoluta²⁸ (Envisat ASAR, ERS 1 & 2, ACOS, RADARSAT2, TerraSAR-X, Cosmo Sky Med).
- Multilooking²⁹ e interferometría Speckle³⁰ (simple y multitemporal).
- ERS-ASAR manejo preciso de la órbita.
- Normalización radiométrica durante la corrección del terreno.
- Simulación SAR.
- Corrección del terreno SAR simulado.
- Herramientas básicas para uso en océanos: detección de derrames de petróleo, detección de barcos y estimación de los campos de viento (SAR).

²⁶Base de datos específicos y de tamaño reducido. ²⁷Mapulación del color. ²⁸ Con esto se obtiene una mejor precisión en los resultados. ²⁹Proceso que se realiza con la finalidad de generar una imagen con un tamaño de pixel aproximadamente cuadrado. ³⁰Consiste en el análisis de patrones de intensidad producidos por la interferencia mutua entre frentes de onda coherentes que son sujetos a diferencias de fase.

- Implementación Java.
- Herramienta de análisis, visualización e interfaz gráfica integrada.
- Ejecución gráfica y líneas de comandos.
- Gestión de memoria para trabajar con volúmenes de datos muy grandes.
- Diseño modular para fáciles modificaciones y actualizaciones.
- Los usuarios pueden añadir sus propios módulos.
- Compatibilidad con el procesador multiprocesos.

Uso de la herramienta Nest

Primero se debe descargar el software en el siguiente enlace:

<https://earth.esa.int/web/nest/downloads>.

Interfaz del programa en la figura 18:

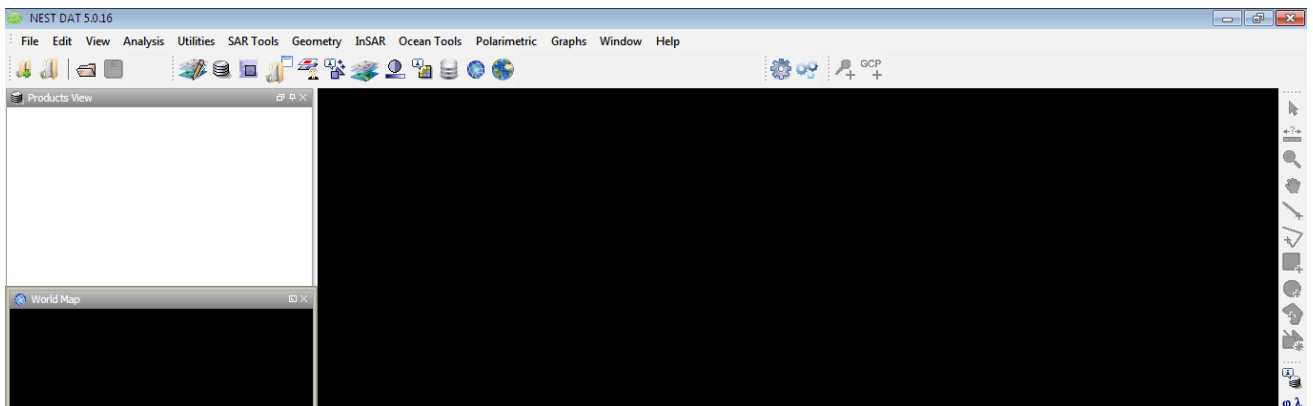


Figura 18: Interfaz gráfica de la herramienta Nest.

En la página oficial de la Agencia Espacial Europea:
<https://earth.esa.int/web/guest/data-access>.

Pude encontrar información completa sobre la observación del planeta Tierra (datos disponibles observar figura 19). busqué diversos productos por misión e instrumentos, por tema, tipología y procesamiento a nivel de Tierra.

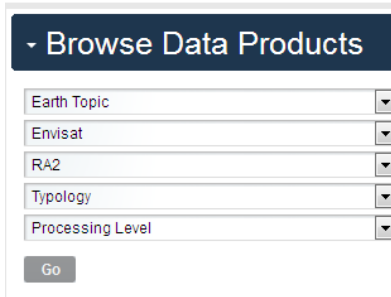


Figura 19: Búsqueda de datos del producto a elegir.

En el caso de los datos del satélite Envisat, éste dejó de operar (fuera de servicio) y la noticia fue dada el 9 de Mayo de 2012.

Los datos de dicho satélite artificial siguen disponibles, en este caso se utilizó el programa EOLI “ESA’s Link to Earth Observation”, en donde podemos encontrar imágenes que fueron tomadas por Envisat y otros satélites.

Descargado de la página oficial: <http://earth.esa.int/EOLi/EOLi.html> .

Una vez descargado procedemos a instalarlo, y seleccionamos la misión Envisat, en la sección del área se eligió “México” ver figura 20.



Figura 20: Búsqueda de datos del Instrumento “ENVISAT ASAR”

Los datos arrojados fueron los siguientes:

```

Mission          ENVISAT-1
Sensor          ASAR/IM
Product         ASA_IM_OP
Orbit           2643
Track           84
Swath           I4
Start           2002-09-01 17:50:25.08
Stop           2002-09-01 17:50:41.88
FOOTPRINT       33.01 -117.45 32.86 -116.47 31.92 -116.7 32.07 -117.66 33.01 -117.45
ACQUISITIONDESCRIPTOR ENVSAT-1-02090117502588-1600.ASA_IM_OP
THUMBNAI_URL    http://eoliserv.eo.esa.int/rowse/ASARz0IM/ASA_IM_OP/ENVSAT_1/2002090117502588
Scene center
Latitude        32.47
Longitude       -117.07
Status          Archived
Pass            D
ProductCodeId   ASA_IM_OP
PassEquatorLongitude -119.51
MissionPhaseId  A
Dataset ID      ASA_IM_OPNPK20020901_174843_000001192009_00084_02643_0579.NI
ArchivingCentre UP
AcquisitionStation KS
Polarisation    VV

```

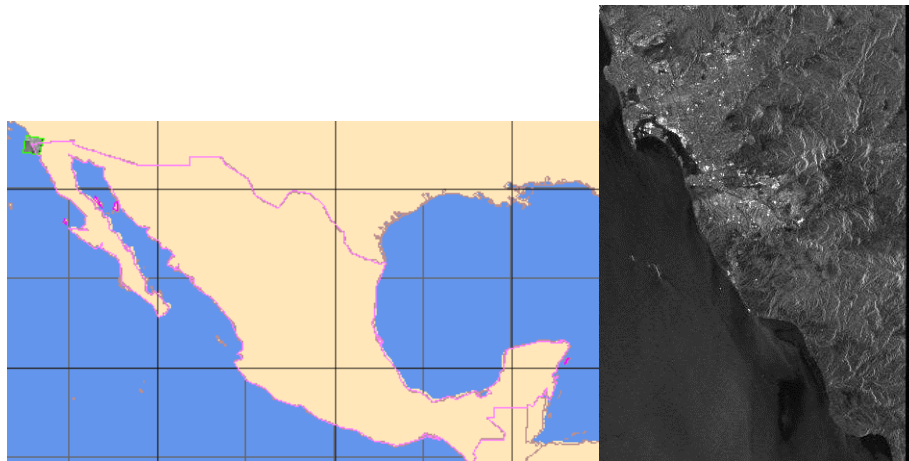
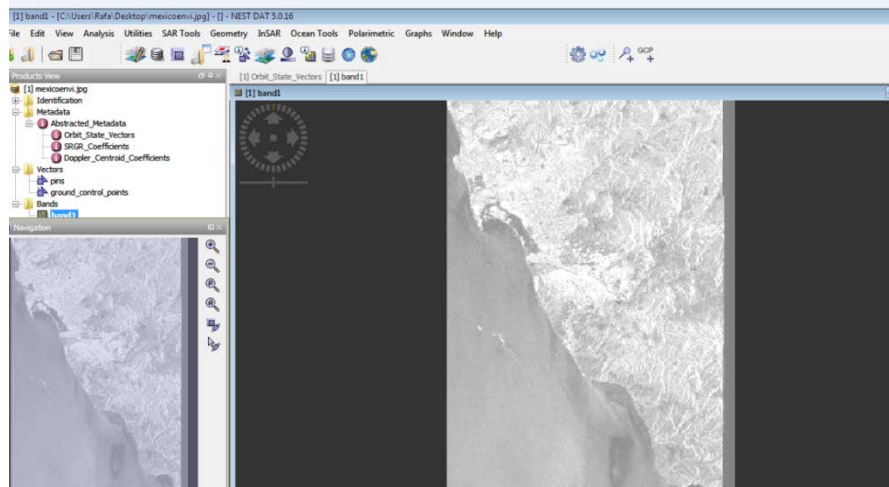


Figura 21: Imagen³¹ desplegada con el programa EOLI-SA

Además proporcionó información complementaria:

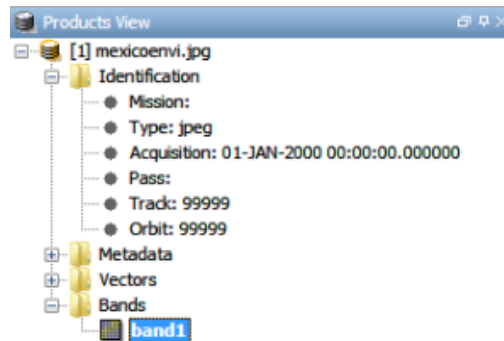
| Display | Mosaic | Id | Mission | Sensor | Product | Status | Start | Stop |
|-------------------------------------|-------------------------------------|-------|-----------|---------|-----------|----------|------------------------|------------------------|
| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | 2 | ENVISAT-1 | ASAR/IM | ASA IM OP | Archived | 2002-09-01 17:50:25.88 | 2002-09-01 17:50:41.88 |
| Orbit | Track | Swath | Pass | | | | | |
| 2643 | 84 | I4 | D | | | | | |

Ahora abrí esa imagen con el programa NEST:



³¹Podemos observar que la imagen representa una parte de Baja California Norte.

Algunas aplicaciones que posee la herramienta (NEST):



- **Identification:** Información básica sobre el producto (Misión, tipo de producto, el tiempo de adquisición, de paso y órbita).
- **Metadata:** Incluye todos los metadatos originales dentro del producto, la historia gráfica de procesamiento, entre otros datos.
- **Vectors.**
- **Bands:** Las bandas actuales dentro del producto y bandas virtuales son creadas por Nest. Se utilizan iconos diferentes para distinguir estas bandas.

Ventana de navegación (figura 22):

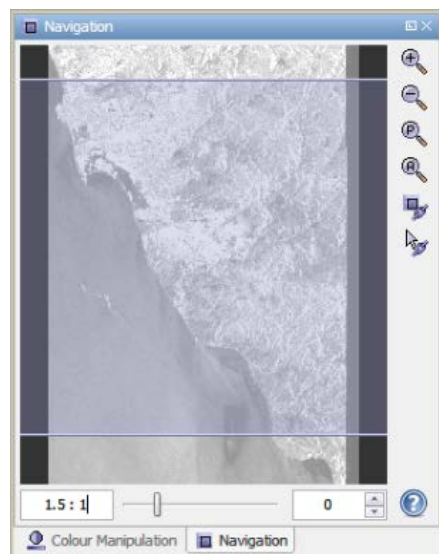
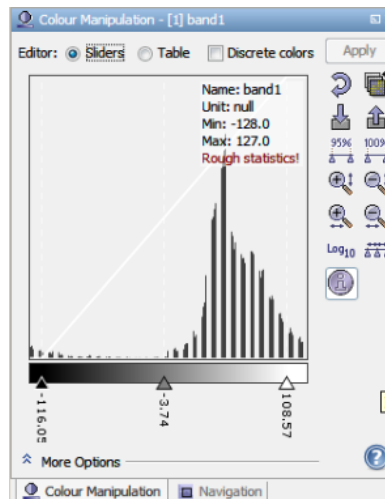


Figura 22: Vista de navegación

- **Zoom In:** incrementa la magnitud.
- **Zoom out:** Decrementa la magnitud.
- **Zoom actual pixel:** Establece el factor de zoom en el valor predeterminado de forma que el tamaño de un pixel de la imagen tiene el mismo tamaño de un pixel de la pantalla.
- **Zoom All:** Ajusta la ampliación de manera que toda la imagen entera quepa en la visión de la imagen (vista).

LUT manipulación del color:



Información general:

| | |
|----------------------------------|--------------------------------------|
| Information - [1] mexicoenvi.jpg | |
| Product name: | mexicoenvi.jpg |
| Product type: | jpeg |
| Product description: | |
| Product format: | ImageIO |
| Product reader: | ImageIO Products |
| Product file location: | C:\Users\Rafa\Desktop\mexicoenvi.jpg |
| Product scene width: | 336 pixels |
| Product scene height: | 476 pixels |
| Product start time (UTC): | Not available |
| Product end time (UTC): | Not available |

Vista del Pixel:

Muestra información de los pixeles mientras mueve el ratón sobre la imagen.

The screenshot shows a "Pixel Info View" window with several sections:

- Geo-location:**

| | | |
|---------|-----|-------|
| Image-X | 227 | pixel |
| Image-Y | 64 | pixel |
- Time Info:**

| | | |
|------------|---------------------|--------------------|
| Date | No date information | YYYY-MM-DD |
| Time (UTC) | No time information | HH:MM:SS:mm [A...] |
- Tie Point Grids:** (Empty section)
- Bands:**

| | |
|--------|----|
| band 1 | 68 |
|--------|----|

At the bottom, there are icons for "Products", "Pixel Info", and "Mask Manager", along with a "Snap to selected pin" checkbox.

5 Optimización por Campos de Markov

5.1 Fundamento

Los métodos Bayesianos acoplados con los modelos Markovianos por lo general resultan en una función (costo en temas de optimización). Con el fin de encontrar una estimación, debemos optimizar esta función.

Sea $S = \{1, 2, \dots, r\}$ un espacio de estados finito, y π una distribución de probabilidad sobre S :
¿Cómo obtener un valor de la variable X con distribución π ?

Si r , es el cardinal (conjunto de elementos del conjunto), el cual no es muy grande (del orden del millar), se simula π por el método clásico, se forma la sucesión creciente en $[0, 1]$ que define la función de distribución de $X \sim \pi$,

1. $F_0 = 0$ y para $0 < i \leq r$, $F_i = \sum_{j \leq i} \pi_j$;
2. Se simula una distribución uniforme³⁰ U en $[0, 1]$
3. Se guarda el valor $x = x(U)$ si $F_{x-1} < U \leq F_x$.

A pesar de que este método de simulación es directo y exacto, sólo es aplicable cuando el cardinal de S no es muy grande.

Desafortunadamente este es un problema computacional muy difícil de resolver y es conocido como optimización combinatoria.

Por ejemplo considerando una imagen 16×16 con sólo dos etiquetas posibles por pixel, obtendremos un espacio de configuración de 2^{256} elementos, teniendo en cuenta que este ejemplo es de tamaño pequeño, un problema real en imágenes hará intervenir un valor de $256^{512 \times 512}$! (256 niveles de gris, imagen con 512×512 pixeles).

Es imposible entonces encontrar el grado óptimo calculando los valores posibles de la función.

La idea de la solución viene de la física estadística³² y fue en 1953 cuando Nicholas Metrópolis propuso una simulación Monte-Carlo para encontrar los estados de equilibrio de los sistemas termodinámicos. Esto fue realizado a principios de los años 80s, por Cerny y Kirkpatrick.

Existe una analogía entre minimizar la función de coste de un problema de optimización combinatoria y la búsqueda de mínimos de energía de los sistemas termodinámicos enfriando lentamente un sólido hasta que se alcanza el equilibrio.

Ellos sustituyeron la función de energía del sólido por la función de coste y ejecutaron el algoritmo de Metrópolis en una secuencia lenta de disminución de la temperatura. El algoritmo de optimización combinatoria definido se denominó como "recocido simulado".

El Recocido Simulado es un método de optimización inspirado en el proceso de templado de metales usado desde alrededor de los 5000 años antes de Cristo. El proceso de templado de metales consiste de tres fases: una fase de calentamiento a una temperatura determinada; en la segunda fase se sostiene la temperatura alta lo cual permite a las moléculas acomodarse en estados de mínima energía; y se sigue de una fase de enfriamiento controlado para aumentar el tamaño de sus cristales y reducir sus efectos.

La investigación en el campo ha crecido rápidamente resultando en una variedad de contribuciones al original recocido simulado. La más importante es probablemente el muestreador de Gibbs.

³²Es una rama de la física que mediante la Teoría de la probabilidad es capaz de deducir el comportamiento de los sistemas físicos macroscópicos a partir de ciertas hipótesis sobre los elementos o *partículas* que los conforman. ³⁰Es una distribución de probabilidad continua donde todos los valores de un intervalo específico tienen la misma probabilidad

5.2 Esquema de Metrópolis

Iniciaremos este capítulo con la formulación inicial del algoritmo de metrópolis, supongamos que tenemos N partículas en un cuadrado, si conocemos las posiciones de éstas, podremos calcular fácilmente la energía del sistema.

$$E = \frac{1}{2} \sum_{i=1}^N \sum_{j=1, j \neq i}^N V(d_{i,j});$$

Donde V es el potencial entre las moléculas, y $d_{i,j}$, es la distancia entre las partículas i y j . Para calcular el valor de la fuerza de equilibrio, usaremos la siguiente integral [9]:

$$F = \frac{\int F e^{-\frac{E}{kT}} d^{2N} p d^{2N} q}{\int e^{-\frac{E}{kT}} d^{2N} p d^{2N} q}$$

Como resulta imposible resolver cientos de integrales dimensionales por los métodos numéricos usuales, recurrimos a los métodos de Monte Carlo.

El método de Monte Carlo para muchas integrales dimensionales consiste en la integración sobre un muestreo de puntos al azar en vez de realizarla sobre una serie regular de puntos.

La descripción del método es la siguiente:

1. Colocar cada una de las N partículas en una posición arbitraria (un punto arbitrario en el espacio dimensional de configuración $2N$).
2. Estimar la energía de sistema:

$$E = \frac{1}{2} \sum_{i=1}^N \sum_{j=1, j \neq i}^N V(d_{i,j});$$

3. Asignar al estado de la configuración una probabilidad:

$$e^{-E/KT}$$

Cabe hacer la aclaración que este método no es práctico para configuraciones del sistema que se encuentren dentro de un rango relativamente angosto de energía, por ello utilizaremos un esquema de Monte Carlo modificado, en lugar de elegir las configuraciones al azar.

Definiendo el método mejorado (observar figura 23):

1. Colocamos las N partículas en cualquier configuración, después mediante un mecanismo de perturbación (pequeña distorsión del estado) movemos cada una de las partículas en la sucesión dentro de un cuadrado³⁴ de lado 2α centrado en sus coordenadas originales de acuerdo a :

Respecto a X: $x + \alpha \varepsilon_1$

Respecto a Y: $y + \alpha \varepsilon_2$

³⁴De acuerdo con la hipótesis de periodicidad, si se mueve la partícula fuera del cuadrado, esto solo significa que vuelve a entrar al cuadrado por el lado opuesto

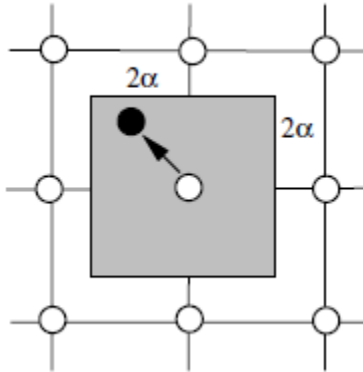


Figura 23: Movimiento de partículas de acuerdo al algoritmo de metrópolis

Donde α es el máximo desplazamiento permitido (arbitrario) y $\varepsilon_1, \varepsilon_2$ son números aleatorios entre -1 y 1.

2. Calculando el cambio de energía del sistema ΔE , el cual es causado por el movimiento, si:
 - 2.2. $\Delta E < 0$, esto quiere decir que el movimiento llevará al sistema a un estado de menor energía, permitir el movimiento y la partícula en su nueva posición.
 - 2.3. $\Delta E > 0$, tomar el movimiento con probabilidad:

$$e^{-E/KT}$$

Tomamos un número aleatorio ε_3 entre 0 y 1.

Si $\varepsilon_3 < e^{-E/KT}$ moviendo la partícula a su nueva posición.

Si $\varepsilon_3 > e^{-E/KT}$ la regresando a su vieja posición.

Se puede observar que el movimiento podrá ser permitido o no, de esta forma estaremos en una posición diferente o igual a la original; para la toma de promedios, considerando que estamos en una nueva configuración:

$$F = \frac{1}{M} \sum_{i=1}^M F_i$$

F_i es valor de la propiedad Fuerza del sistema después de cierta cantidad de movimientos llevados a cabo de acuerdo a lo anterior descrito [9].

Lo que se haría en términos de procesamiento de imágenes con el algoritmo de Metrópolis es generar un vecino, calcularle su energía (función de costo en problemas de optimización) y aceptar ese vecino si tiene menor energía o aceptarlo con mayor energía pero con cierta probabilidad que depende de la temperatura (T). Se puede probar que si se realiza este proceso durante muchas transiciones se puede llegar a lo que se conoce como un equilibrio térmico. El equilibrio térmico está caracterizado por la distribución de Boltzmann.

5.3 Recocido simulado

Como se explicó anteriormente el recocido consiste en el calentamiento de un sólido y después en un enfriamiento, a cada temperatura "T" el sólido se permite alcanzar un equilibrio térmico que es caracterizado por la distribución de Boltzmann.

$$P(w) = \frac{1}{Z(T)} e^{-U(w)/KT}$$

Donde:

U (w) es la energía de los estados,

Z (T) es la función de normalización dependiente de T,

K es la constante de Boltzmann

Para un "T" fijo, la ecuación anterior no es otra cosa que una distribución de Gibbs, claramente, cuando la temperatura disminuye, la distribución se concentra en los estados con baja energía y cuando la temperatura se acerca a cero, sólo los estados mínimos tienen una probabilidad no cero.

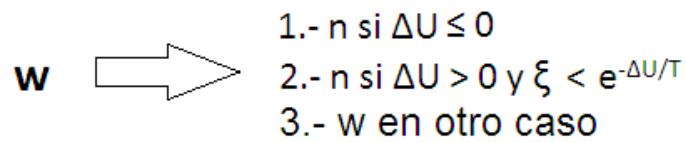
La disminución de la temperatura es crucial: si el enfriamiento es demasiado rápido y el sistema no permite que se alcance el equilibrio térmico para cada temperatura, un mínimo global no se puede alcanzar. Para una temperatura fija, la transición al equilibrio térmico es simulada por el algoritmo de Metrópolis.

El algoritmo de recocido simulado se puede ver como una iteración de algoritmos de Metrópolis, Si se baja la temperatura suficientemente lento se puede alcanzar el equilibrio térmico en cada temperatura.

Denotando como n y w a las configuraciones de un problema de optimización combinatoria y la función coste de energía como U(w) de la configuración W, que corresponde a la energía de los estados en un sistema termodinámico.

Algoritmo de Recocido Simulado:

1. Utilizando un contador i=0 e inicializando "w" aleatoriamente, elegimos una temperatura lo suficientemente alta $T = T_0$.
2. Crear una perturbación de prueba "n" de la configuración actual "w".
3. (Criterio de metrópolis) Calculamos $\Delta U = U(n) - U(w)$ y aceptamos "n" si $\Delta U < 0$ además aceptamos con probabilidad $e^{-\Delta U/T}$ (analogía con termodinámica):



Donde:

Si se cumple la condición 2 (mostrada arriba), se genera un número ξ , el cual es un número aleatorio uniforme entre [0, 1).

4. Ir a paso 2 hasta que se alcanza el equilibrio.
5. Disminuya la temperatura: $T = T_{k+1}$ e ir al paso 2 con $k = k+1$ hasta que el sistema se enfríe.

El algoritmo también es llamado recocido homogéneo ya que es descrito por medio de una secuencia de cadenas de Markov homogéneas. ***Si la temperatura disminuye después de cada transición, el algoritmo es descrito por cadenas de Markov no homogéneas, el cual es referido como recocido no homogéneo y es la forma de recocido más utilizado [10].***

5.4 Muestreador de Gibbs

El muestreador de Gibbs es el algoritmo de recocido simulado más popular, consiste en un matriz de aceptación acoplada junto con un esquema de recocido no homogéneo.

Algoritmo:

1. Hacer $K = 0$, asignando una configuración inicial w y $T = T_0$ a modo que se encuentre a una temperatura lo suficientemente alta.
2. Para cada configuración la cual difiere en al menos un elemento de la configuración actual w (los elementos son denotados como N_w), calcular la energía $U(n)$ ($n \in N_w$).

NOTA:

El algoritmo de recocido simulado genera una secuencia de configuraciones las cuales constituyen una cadena de Markov.

$P_{w,n}(k-1, k)$ es la probabilidad de la configuración después de k transiciones, de n dado la configuración previa w . Además $X(k)$ denota el estado al que se llegó después de k transiciones.

La probabilidad del evento es dada por [11]:

$$P(X(k) = w) = \sum_{\zeta} P(X(k-1) = \zeta) P_{\zeta,w}(k-1,k) \quad k = 1, 2, \dots$$

3. (Muestreador de Gibbs) se extrae una muestra de tal manera que se acepta con probabilidad:

$$\frac{e^{-U(n)}}{\sum_{\zeta \in N_w} e^{-U(\zeta)}}$$

Como la nueva configuración.

4. Decrementar la temperatura $T = T_{k+1}$ e ir al paso 2 con $K = K+1$ hasta que el sistema se enfríe.

En el caso de dos sistemas de estados como en el caso del modelo de Ising, el muestreador de Gibbs es equivalente al algoritmo de Metrópolis. Haciendo la observación que la matriz de generación es simplemente $G_{w,n} = 1$ para todo w y $n \in N_w$, 0 en otro caso.

5.5 Parámetros: vecindades, cliques, temperatura y homogeneidad

Los elementos que serán necesarios y de vital importancia en el proceso de segmentación estocástica son:

- Vecindades:

Corresponden a las regiones alrededor de un pixel, que resultan de dividir la imagen (explicación más completa en la sección 3.2.1).

- Cliques:

Corresponde a un indicativo de la relación que guardan entre sí los pixeles vecinos (explicación más detallada en la sección 3.2.2).

- Temperatura:

Calculada a partir de los parámetros de Markov (conviene elegir una temperatura elevada).

- Homogeneidad:

En la textura homogénea se puede observar que los niveles de gris de los pixeles son muy semejantes, en estos casos el nivel de gris de un pixel no depende de los niveles de gris de los vecinos³⁴. Además cuentan una gran peculiaridad que es la de no presentar un origen espacial definido.

³⁴ Esto es debido a la semejanza que existe en los niveles de gris.

6. Determinación de la extensión marina de derrame de petróleo por medio de campos aleatorios de Markov

El derrame de petróleo de la plataforma Deepwater Horizon en las profundidades del Golfo de México ocurrido el 22 de abril de 2010, fue el accidente marino más grande en la historia de la industria petrolera.

Poco después del accidente, todos los satélites de observación terrestre centraron su atención en la adquisición de imágenes sobre el Golfo de México.

El radar de apertura sintética es sin duda el sensor más potente para la observación de los fenómenos oceánicos, como las olas, los vientos, derrames de petróleo, hielos marinos, etc.

Gracias al proyecto de la Agencia Espacial Europea, se encuentran disponibles los datos recabados por la herramienta Envisat/ASAR, esta herramienta fue elegida para el estudio de los accidentes de derrame de petróleo.

Lo que prosiguió fue detectar la capa de aceite y medir su superficie. La presente tesis presenta un esquema de procesamiento basado en la teoría de campos aleatorios de Markov (CAM), ésta describe la información global de los términos de probabilidad involucrando representaciones de vecinos locales de los datos de dispersión retrógrada³⁵.

Los atributos espaciales están estructurados por la ecuación de Bayes derivado de un máximo, una estimación a posteriori.

El problema de segmentación markoviano es resuelto con un método numérico de optimización. Este esquema fue aplicado a imágenes del Satélite Envisat usando la herramienta ASAR sobre el golfo de México el día 9 de Mayo de 2010, cuándo el derrame del petróleo alcanzó su máxima extensión.

Para el análisis de la imagen, se desarrolló un esquema específico de segmentación, este enfoque sólo utiliza información radiométrica de las escenas del radar.

El proceso de segmentación es modelo por:

- La formulación de Bayes.
- La optimización de las funciones de probabilidad por medio de la teoría de campos aleatorios de Markov como definición de un criterio de máximo a posteriori.
- Modificación del algoritmo de minimización estocástico para actualizar los parámetros del modelo a priori y la vecindad del sistema.

³⁵ Desviar (radiación o partículas) a través de un ángulo de 180°.

³⁶ Filtro utilizado para la recuperación máxima de la señal

7. Esquema de regulación adaptativa

La regulación adaptativa que utilizaremos en esta presente tesis consistirá en un proceso de segmentación de imágenes, el cual consiste en dividir la imagen en sus partes que la constituyen hasta llegar a un nivel de subdivisión en el que se pueden aislar las regiones de interés. Existen dos propiedades básicas de los valores del nivel de gris: discontinuidad o similitud entre los niveles de los píxeles vecinos.

Discontinuidades: Se realiza la división de la imagen basándose específicamente en los cambios bruscos del nivel de gris:

- ✓ Detección de puntos aislados.
- ✓ Detección de líneas.
- ✓ Detección de bordes.

Similitud: En este caso se divide la imagen pero ahora se basa en la búsqueda de zonas que tengan valores similares, conforme a criterios fijados:

- ✓ Crecimiento de región.
- ✓ Umbralización.

Para éste algoritmo se basa en la propiedad básica de similitud entre los niveles de gris de los píxeles vecinos.

7.1 Planteamiento y esquema general

El esquema que se utilizará para este proceso de regulación adaptativa se dividirá en dos partes, la primera será empleando procesos aleatorios (Bayes) en un proceso de segmentación y la segunda consistirá en emplear el recocido simulado como método de optimización.

A continuación se explicará en qué consiste el esquema general.

Empezando con los procesos aleatorios aplicados en el proceso de segmentación (figura 25):

1.- Definición de clases:

Considerando la regla de Bayes $P(W_i|X) = \frac{P(X|W_i)P(W_i)}{P(X)}$, Siendo X la imagen original y W_i las clases, con $i = 1, 2, \dots, 5$, donde:

$P(W_i|X)$ representa la probabilidad a posteriori.

$P(W_i)$ es la probabilidad a priori de la clase W_i .

$P(X|W_i)$ es la probabilidad conjunta de X dado W_i .

$P(X)$ es la probabilidad total de X.

Para este caso utilizaremos una imagen de prueba titulada parís.bmp (figura 24), la cual fue proporcionada por el Dr. Miguel Moctezuma.

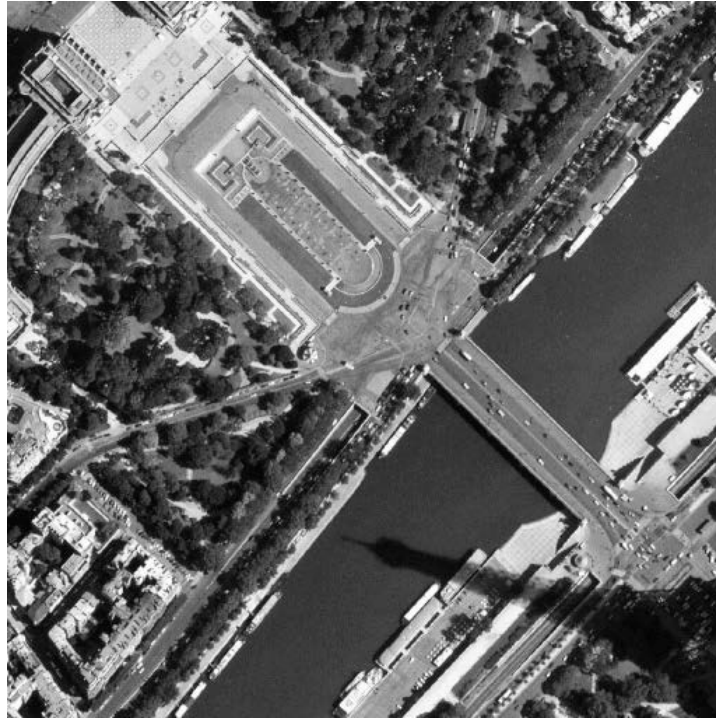


Figura 24: Imagen original X “parís.bmp”

A partir de esta imagen:

- ✓ Definiremos 5 ventanas que identifiquen a 5 clases típicas (agua, sombra, vegetación, carretera clara y oscura).
- ✓ Sobre cada ventana, calculamos las medias m_i , con $i = 1, 2, 3, 4, 5$.

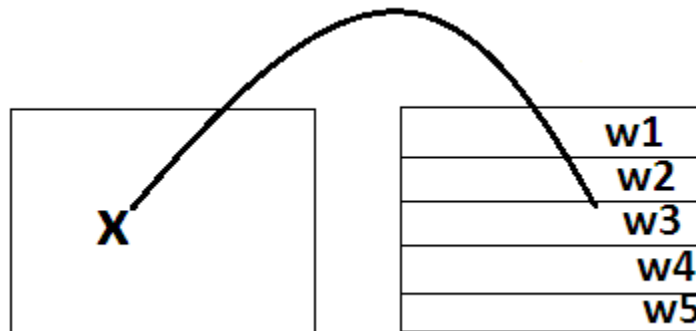


Figura 25: Imagen en grises (izquierda) e imagen dividida en clases (derecha).

2.- Funciones de energía:

Estas se definirán en el siguiente subtema.

3.- Clasificar la imagen X aplicando 50 iteraciones.

- Empleando la aproximación de la regla de Bayes³⁷ $P(W_i|X) = P(X|W_i)P(W_i)$ para calcular la energía a posteriori:

³⁷ Podemos observar que en esta aproximación ya no se encuentra normalizada, es decir la probabilidad ya no estará entre [0,1], pero la proporción se mantiene entre valores grandes y valores muy pequeños.

$$U(W_i|X)_{min} \approx \arg \max(\log P(X|W_i) + \log P(W_i))$$

Un campo aleatorio de Markov discreto es aplicado al modelo de segmentación.

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

- Analizando pixel por pixel la imagen X al aplicar el algoritmo de recocido simulado propiamente aplicado a pixeles, que se explicará en el subtema 7.3.

4.- Repetir el proceso completo con $\beta = 0.1$.

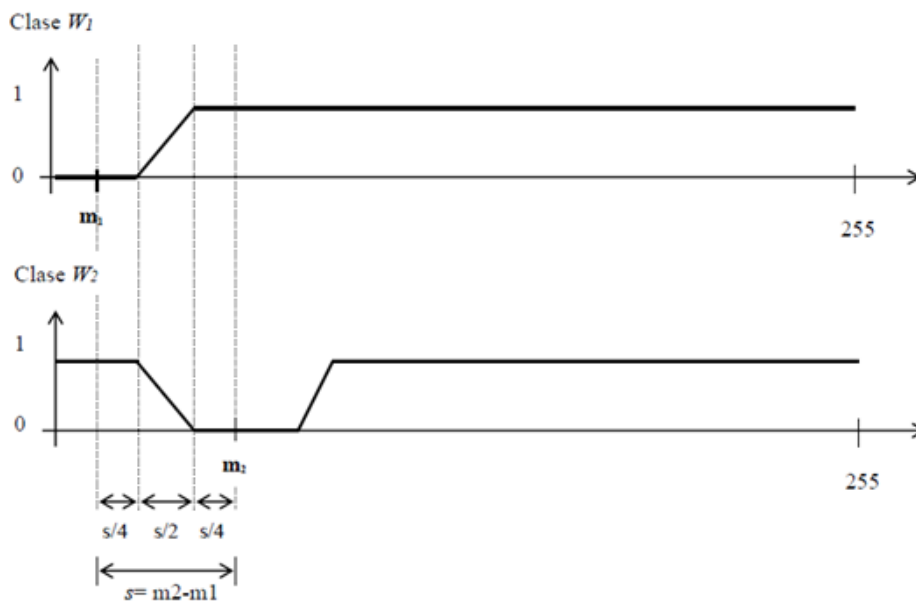
5.- Repetir el proceso completo con $\beta = 0.9$.

6.- Resultados de las iteraciones 1, 10, 20, 40 y 50.

7.2 Funciones de energía

Aproximando las probabilidades conjuntas del numerador de la regla de Bayes mediante las funciones de energía mostradas en la figura 26, éstas gráficas indican la energía condicional para que un pixel dado pertenezca a la clase W_i .

TERMINOS DE ENERGIA $U(X|w_i)$



llegar a la última clase:

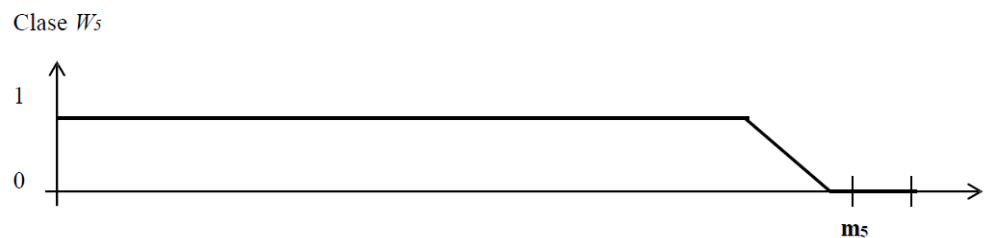


Figura 26: Funciones de energía

Una forma más clara para entender el cálculo de la energía $U(X|W_i)$ es a través de la distribución gaussiana (fig. 27)

$$f(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{1}{2}\left[\frac{x-\mu}{\sigma}\right]^2} \quad -\infty < x < \infty$$

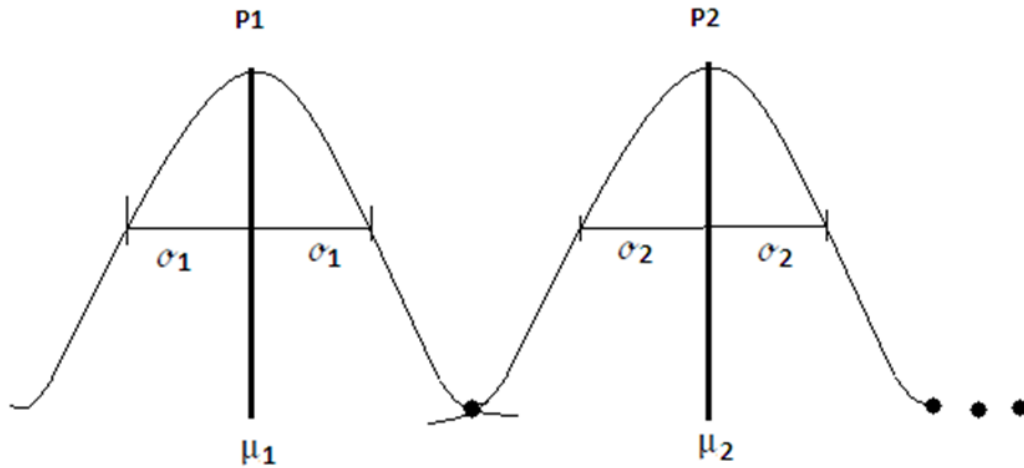


Figura 27: Cálculo de la energía $U(X|W_i)$ a través de la distribución gaussiana

Recapitulando, lo primero que se realizó en el esquema general fue segmentar la imagen X en cinco clases, después se calculó sus medias, una vez teniendo las medias es posible calcular la desviación estándar de una de ellas.

Procediendo a calcular la energía $U(X|W_i)$ con ayuda de las funciones gaussianas de cada clase creada, dependiendo del valor del píxel analizado x , se elegirá a cual clase pertenece (gaussiana 1,2..5).

Nota: Todo lo anterior expuesto se encuentra programado con la ayuda de la herramienta de procesamiento de "Matlab" y el código (programa) se encuentra en el anexo 1.

7.3 Algoritmo de recocido simulado

Después de haber aplicado el proceso de segmentación de la imagen en clases, utilizaremos el método de recocido simulado.

Sea X la imagen original (en niveles de gris) e Y la imagen segmentada (por ejemplo, en cinco clases), la simulación toma como datos iniciales a X y a la imagen de clases originada en la inicialización, $W^{(1)}$. La figura 28 muestra la esquematización del algoritmo de simulación, para el caso de n iteraciones.

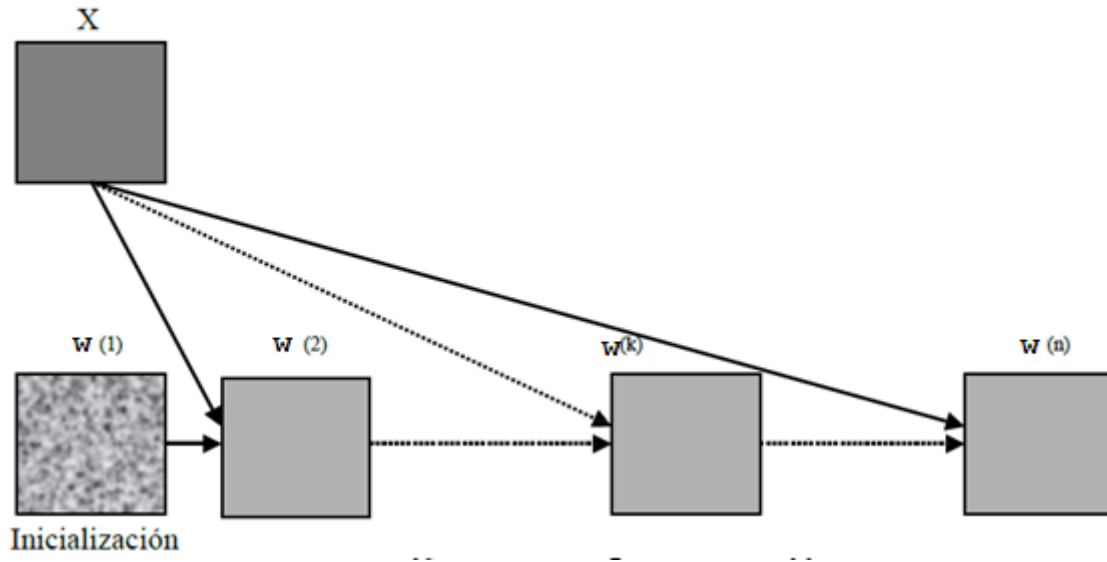


Figura 28: Esquema iterativo. Para la energía $U(W_i)$ se considera al modelo de Potts, el cual se aplica en vecindades en 8-conexidad. La energía $U(X|W_i)$ se evalúa sobre la imagen original X .

Pasos del algoritmo:

1.- Inicialización:

- Determinación aleatoria de la solución $W^{(1)}$ (la clase de cada pixel debe determinarse mediante un generador de números aleatorios, con distribución uniforme, en el rango del número de clases).
- Escoger una temperatura inicial $T^{(1)}$ (el superíndice indica que es la temperatura 1 o inicial) lo suficientemente elevada ($T^{(1)} = 2.5$).

2.- Hacer $N_{iter}=1$, $N_{iter}= n$.

2.1.- Barrer secuencialmente la imagen, en cada etapa (k):

- Escoger un pixel g de coordenadas $[i][j]$, su valor es $W_g^{(k)}$ y su configuración de vecindad es $V_g^{(k)}$.
- Efectuar el sorteo de una variable aleatoria (clase "elección") con una distribución uniforme: $\xi \in \Omega$.

2.2.- Criterio de Metrópolis y Modelo de Potts (Cálculo de la energía de clase o a priori)

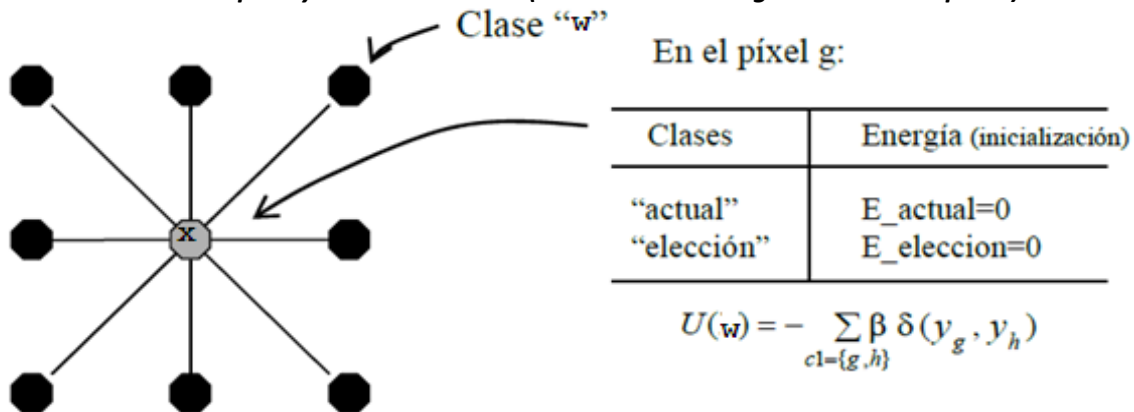
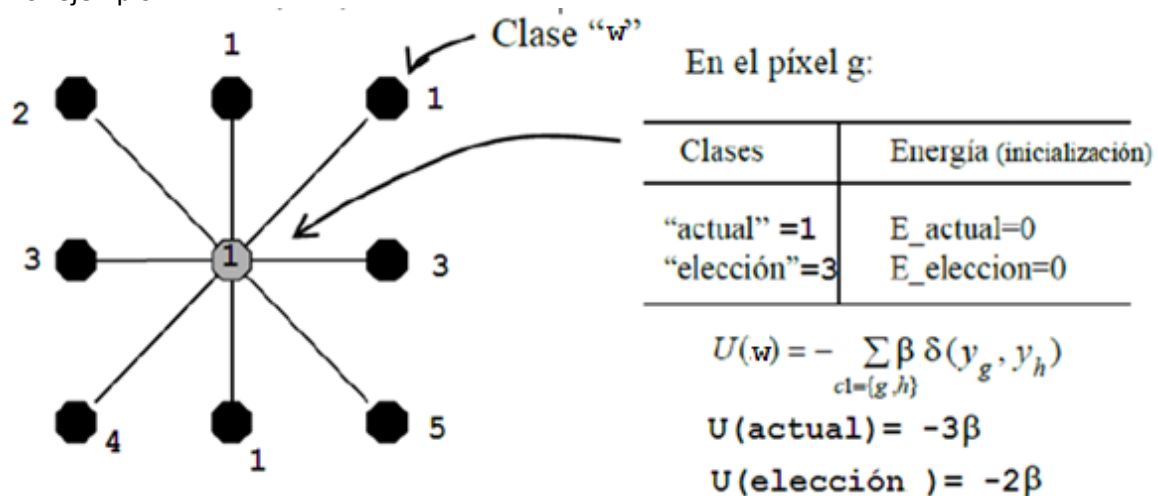


Figura 28: Vecindad $V_g^{(k)}$ asociada al píxel g (al centro de la ventana en 8-conexidad).

De la figura 28 análisis en 8-conexidad (modelo de Potts) para el píxel $g[i][j]$:

- Si "actual" = "x": $E_{\text{actual}} = -\beta E_{\text{actual}}$
- Si "elección" = "w": $E_{\text{actual}} = -\beta E_{\text{elección}}$

Por ejemplo:



2.3.- Calcular la variación de la energía local asociada al píxel g.

En la imagen X, en las coordenadas $[i][j]$, tomar el nivel de gris del píxel analizado g y proyectarlo en las funciones de energía mostradas en la figura 26, en las clases "actual" y "elección". Sumar la contribución a la de Potts:

- $E_{\text{actual}} = E_{\text{actual}} + \text{potencial}[\text{actual}][i][j]$.
- $E_{\text{actual}} = E_{\text{elección}} + \text{potencial}[\text{elección}][i][j]$.

Donde potencial [actual] [i] [j] representa la energía condicional $U(X|W_i)$ e $[i][j]$ son las coordenadas del píxel analizado.

- Si $E_{\text{elección}} \leq E_{\text{actual}}$, entonces $W[i][j] = \text{"elección"}$

En caso contrario:

- $P = e^{[(E_{\text{elección}} - E_{\text{actual}})/T(n)]}$.
- Sorteo de una variable aleatoria ξ_2 , uniformemente distribuida en el rango [0,1].
- Si $\xi_2 < P$, entonces $W[i][j] = \text{"elección"}$.

3.- Regresar al punto 2.1 hasta terminar un barrido de toda la imagen.

4.- Decrementar la temperatura:

$$T(n) = T(n-1) * 0.95$$

5.-Regresar al punto 2 hasta terminar el número de iteraciones fijadas.

Nota: Todo lo anterior expuesto se encuentra programado con la ayuda de la herramienta de procesamiento de "Matlab" y el código (programa) se encuentra en el anexo 1.

7.4 Matlab® como herramienta de procesamiento.

A lo largo de la vida del ingeniero las herramientas computacionales han sido de gran ayuda para realizar operaciones complejas en poco tiempo y así tener una mejor eficiencia.

En esta presente tesis se utilizará Matlab (Fig. 29 y 30) como herramienta de procesamiento, ésta fue creada por la compañía Mathworks, Inc.

Los usos más comunes son para cálculos numéricos, matriciales, visualización de análisis numéricos, procesamiento de gráficos y de señales, otros usos son:

- Desarrollo de algoritmos.
- Modelado y simulación.
- Cálculos matemáticos.
- Gráficas científicas y de ingeniería.
- Análisis de datos, exploración y visualización.

El motivo de selección de Matlab, es que éste ofrece una gran ventaja sobre otros programas, su uso es práctico y muy fácil, no requiere el uso de programación tradicional ya que los problemas y sus soluciones respectivas son expresados de la misma manera en que se escriben matemáticamente.

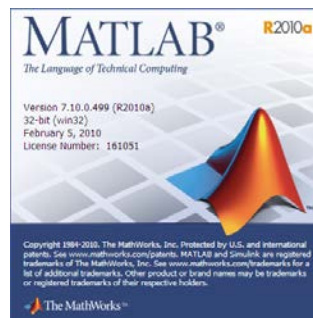


Figura: 29 Logo de Matlab.

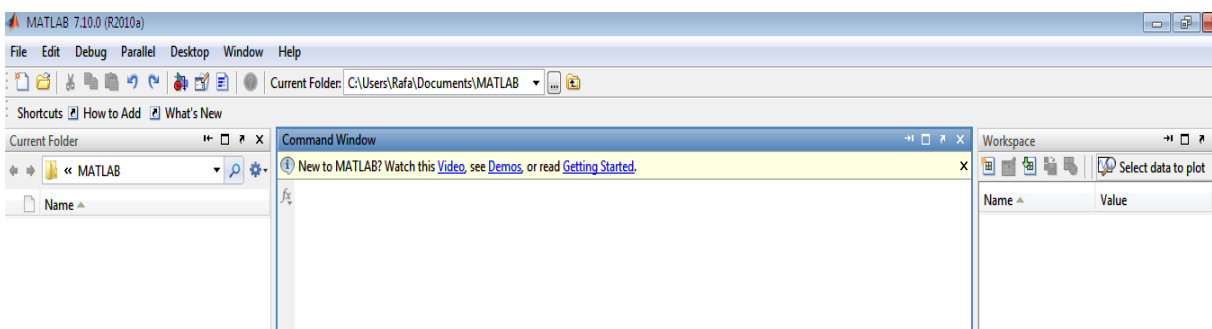


Figura 30: Interfaz gráfica de la herramienta de trabajo Matlab

8. Resultados

8.1 Imágenes de prueba (esquema general)

Para iniciar se retomará la actividad del punto 7.1, definiendo las 5 ventanas que identifiquen a 5 clases, y después calculando sus medias.

Pero antes de realizar la programación se debe fijar las regiones que se utilizarán para cada clase (figuras 31,32,33,34 y 35), para esto se retomará la ayuda de la herramienta NEST, insertando la imagen y con la vista del pixel, determinando las regiones que ocuparán cada clase.

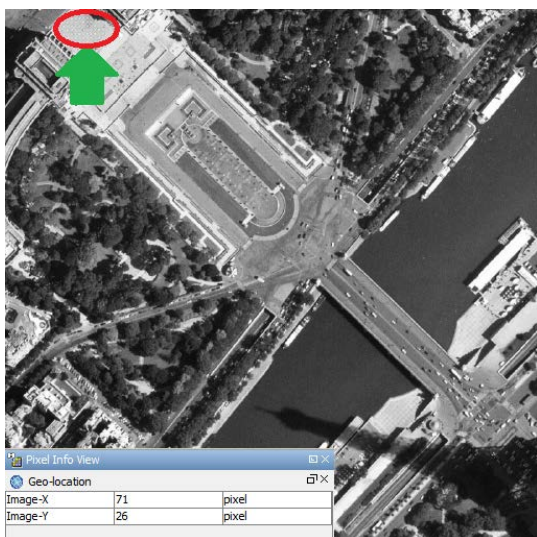


Figura 31: Selección de clase (Construcción)



Figura 32: Selección de la clase (vegetación)

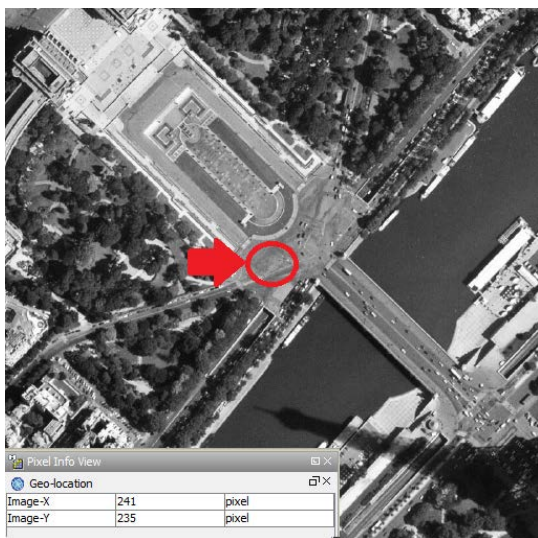


Figura 33: Selección de la clase (Carretera)

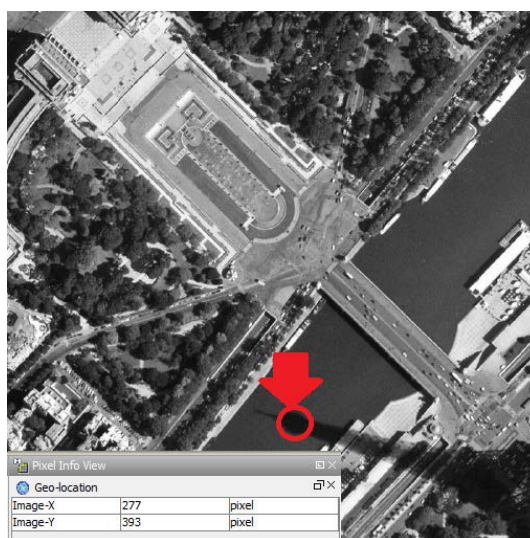


Figura 34: Selección de la clase (Sombra)



Figura 35: Selección de la clase (agua)

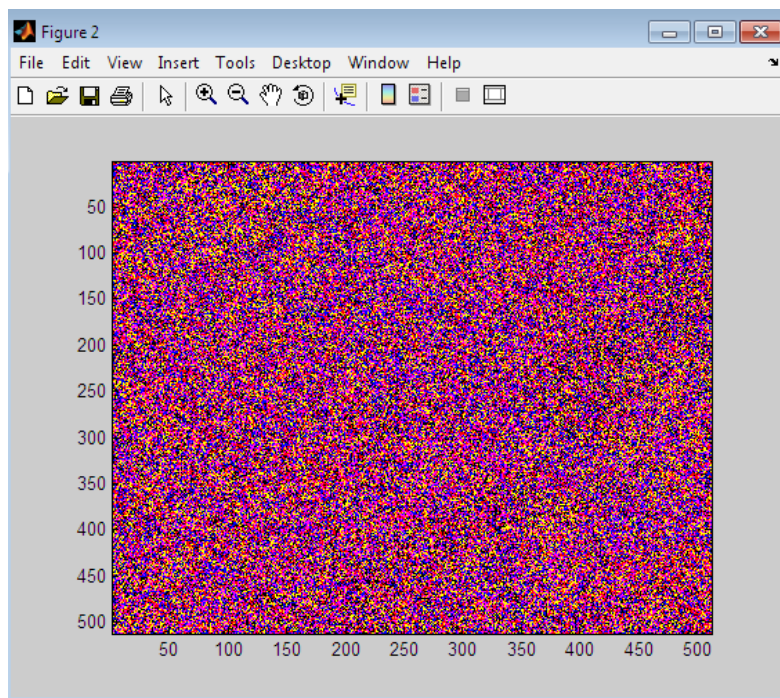
Código utilizado siguiendo el esquema general [Anexo 1].

Resultados de aplicar la segmentación y el recodo simulado como método de optimización:

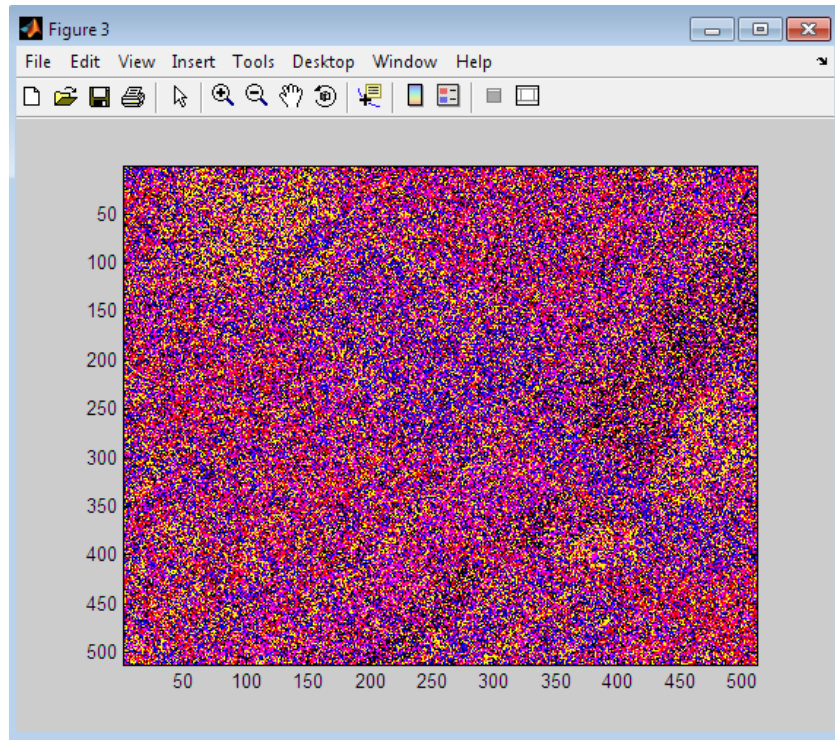
con $\beta=0.1$.

Iteraciones:

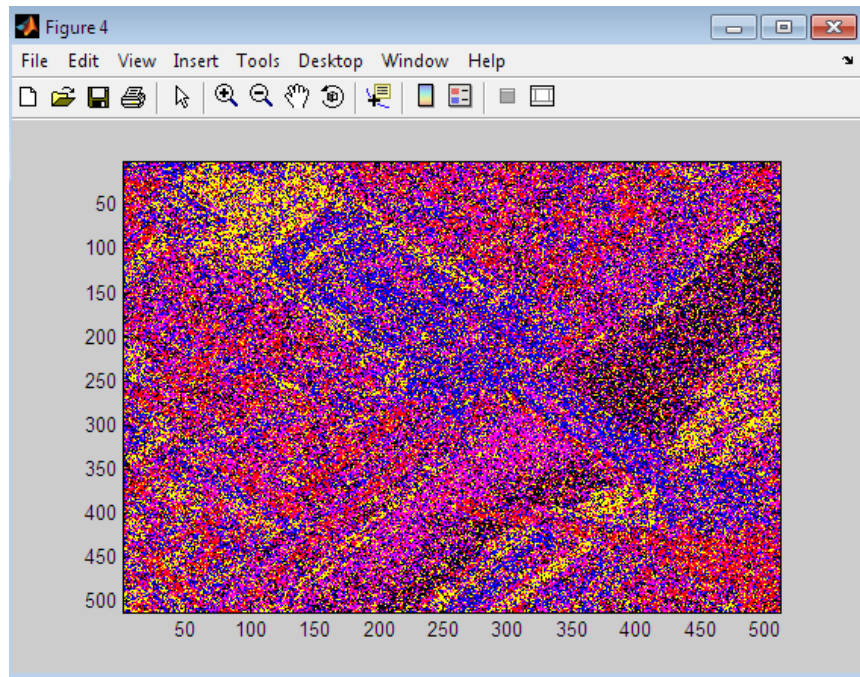
n=1



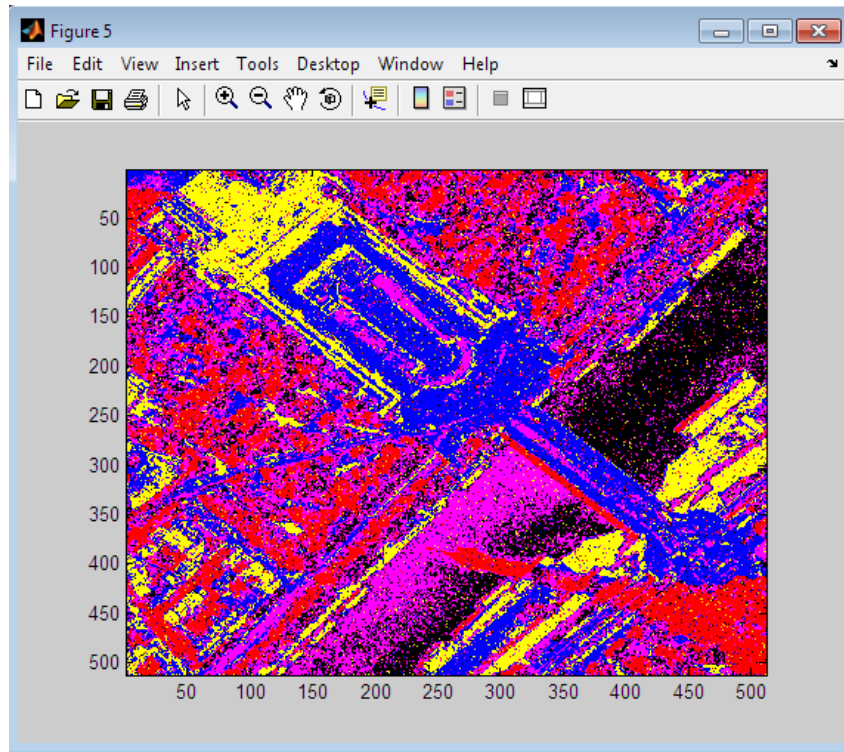
n=10



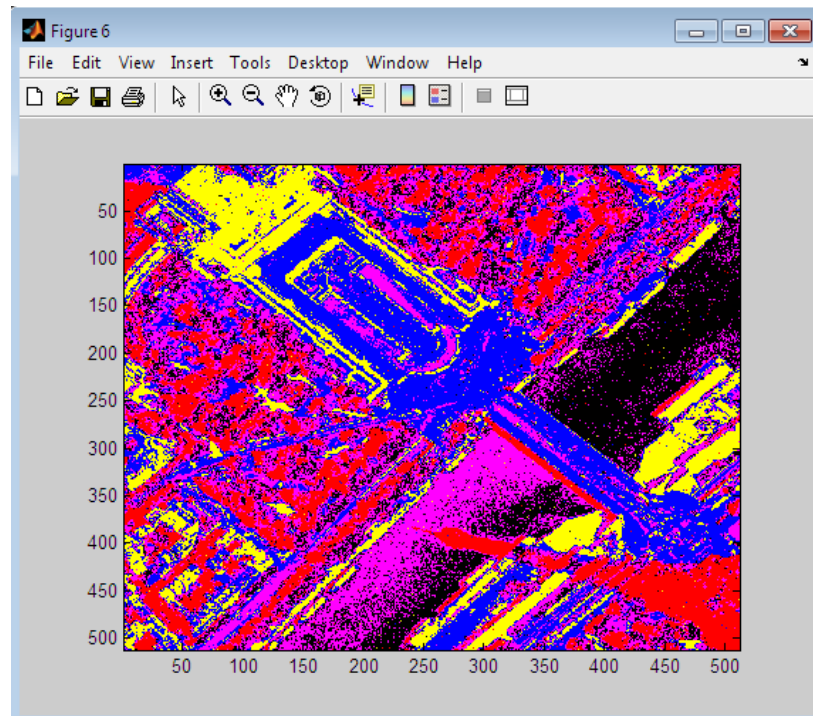
n= 20



n= 40



n=50



Valores de las medias y desviaciones estándar de cada región.

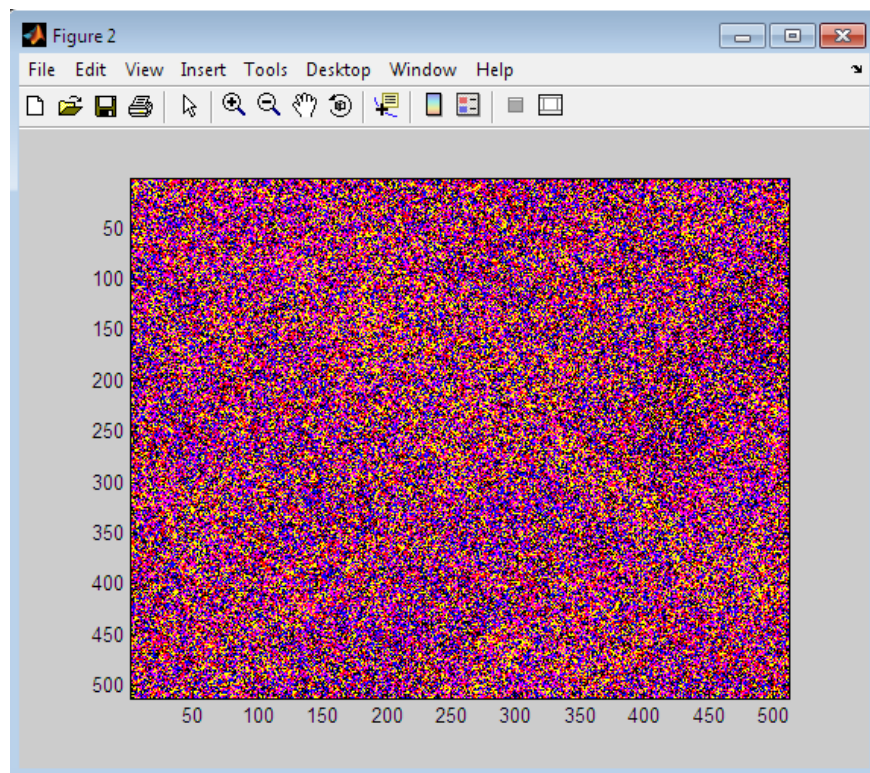
| | | | | | |
|------------|--------|--------|--------|--------|--------|
| m = | | | | | |
| | 0.2764 | 0.2161 | 0.0854 | 0.6107 | 0.7984 |
| v = | | | | | |
| | 0.0175 | 0.0867 | 0.0350 | 0.1231 | 0.0553 |

Se puede observar que para $\beta=0.1$ el mejor resultado se encuentra en la iteración 50.

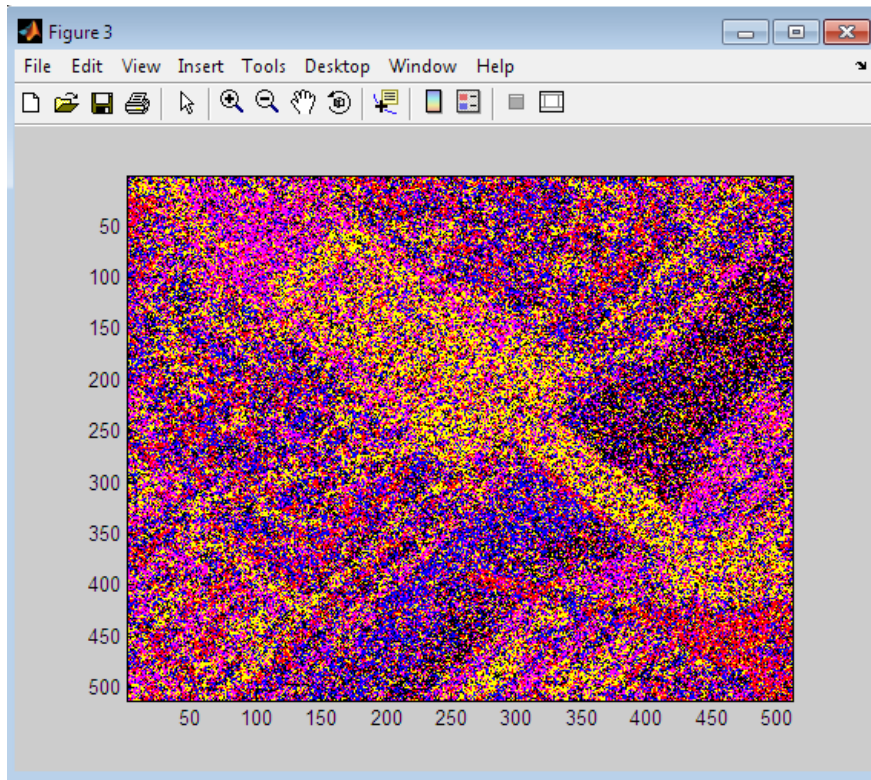
Ahora para $\beta=0.35$:

| | | | | | |
|------------|--------|--------|--------|--------|--------|
| m = | | | | | |
| | 0.2772 | 0.0945 | 0.2118 | 0.7816 | 0.6712 |
| v = | | | | | |
| | 0.0163 | 0.0523 | 0.0763 | 0.1042 | 0.1321 |

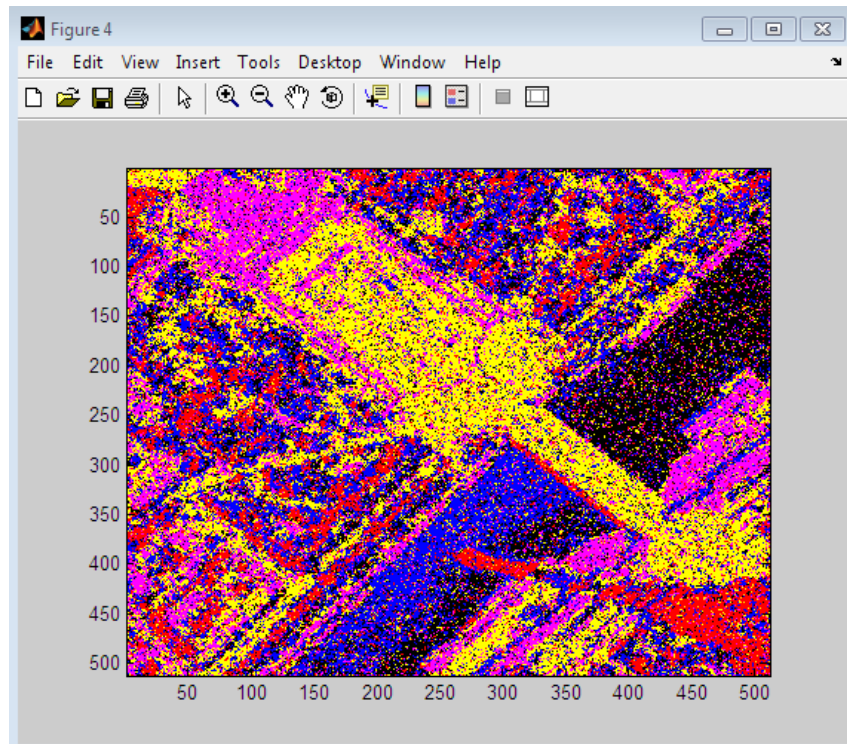
n=1



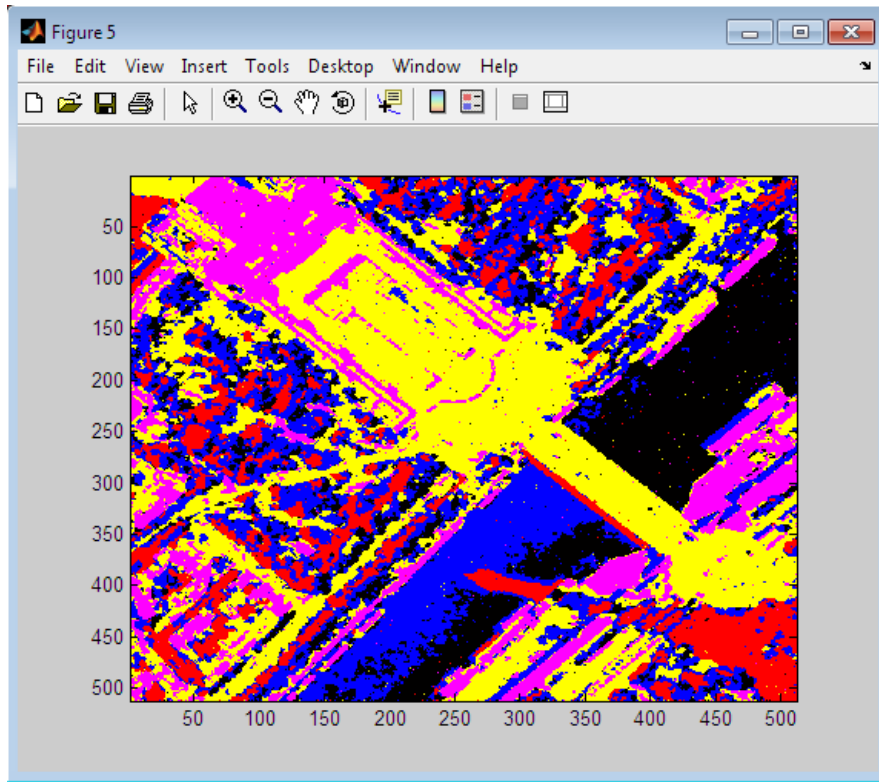
n=10



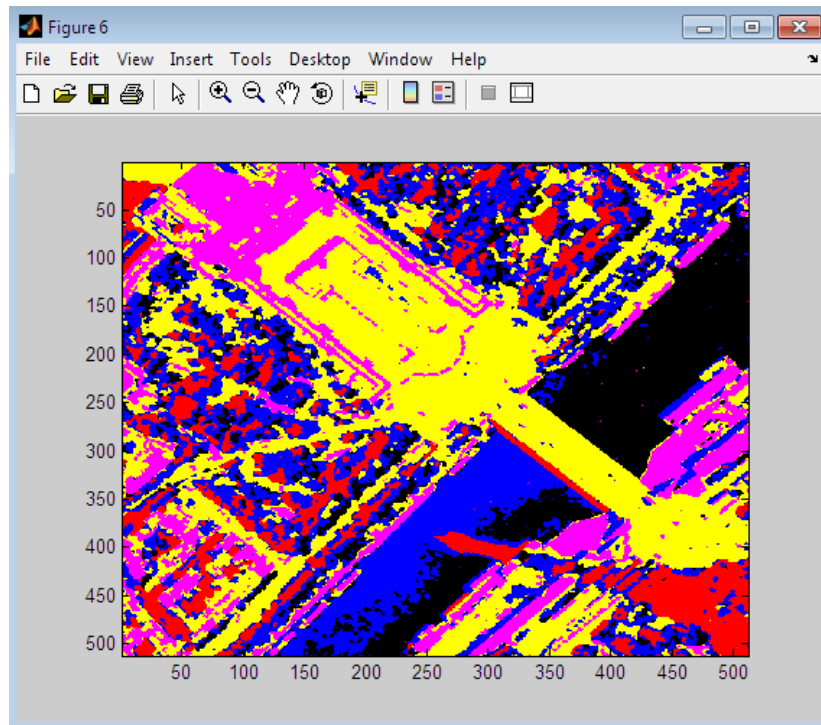
n=20



n=40



n=50



Observando que para $\beta= 0.35$ el mejor resultado se encuentra en la iteración 50, no se incluyen más iteraciones porque se llega a resultados muy parecidos a los de la iteración 50.

Ahora para $\beta= 0.9$:

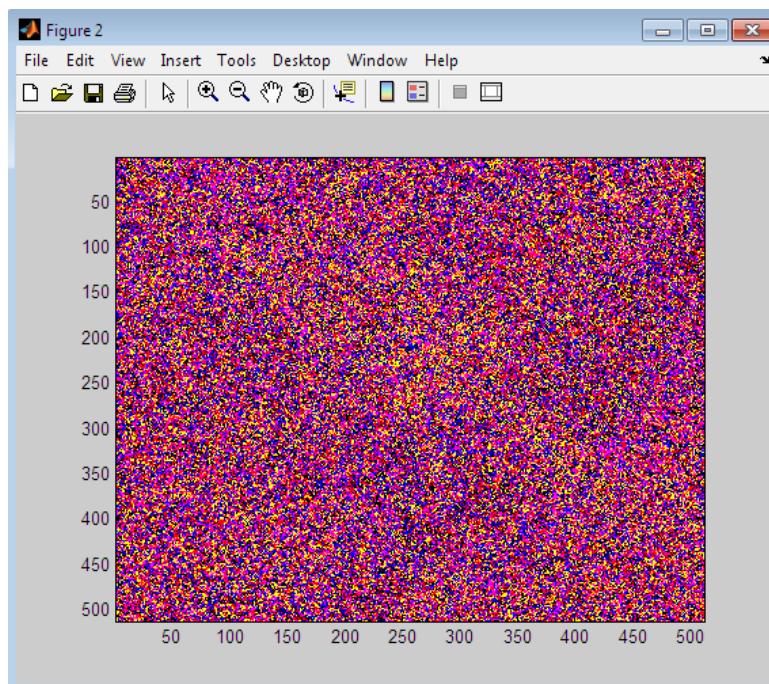
$m =$

0.2762 0.1078 0.2080 0.6291 0.8119

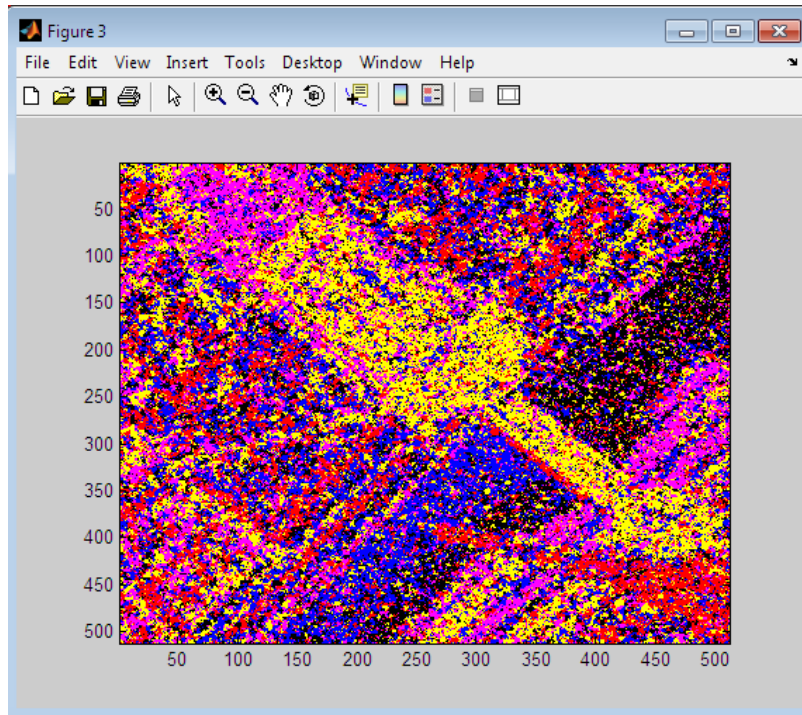
$v =$

0.0198 0.0692 0.0759 0.0809 0.0842

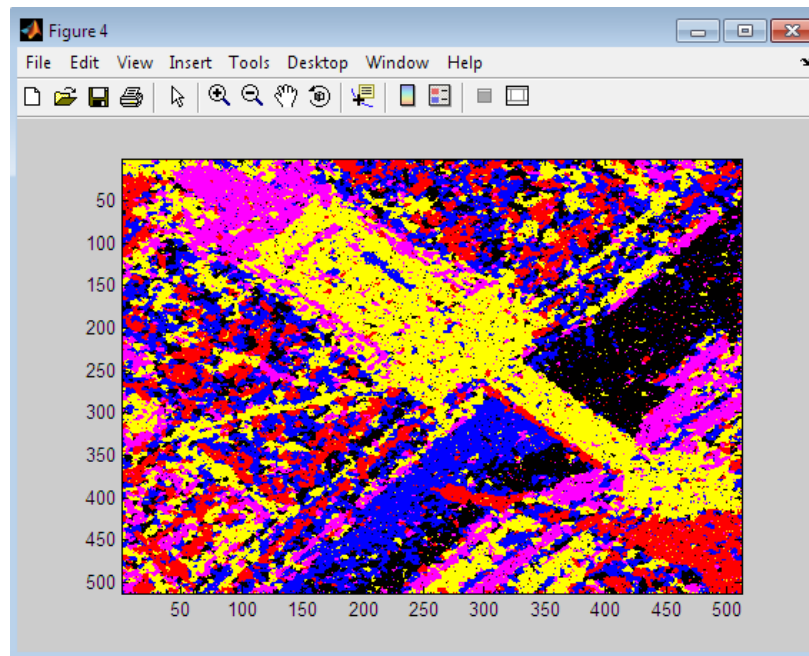
$n=1$



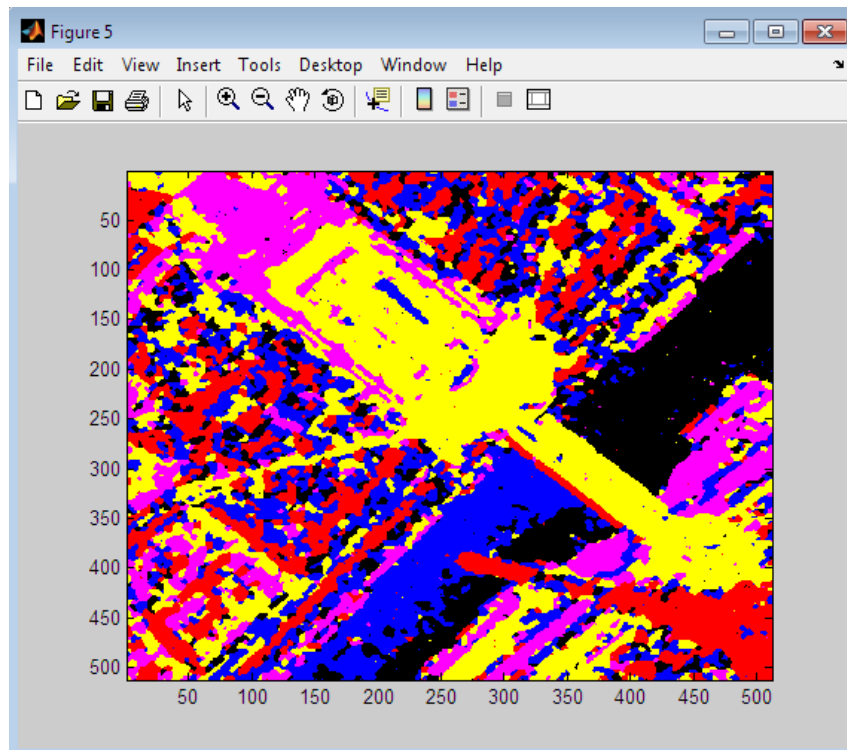
n=10



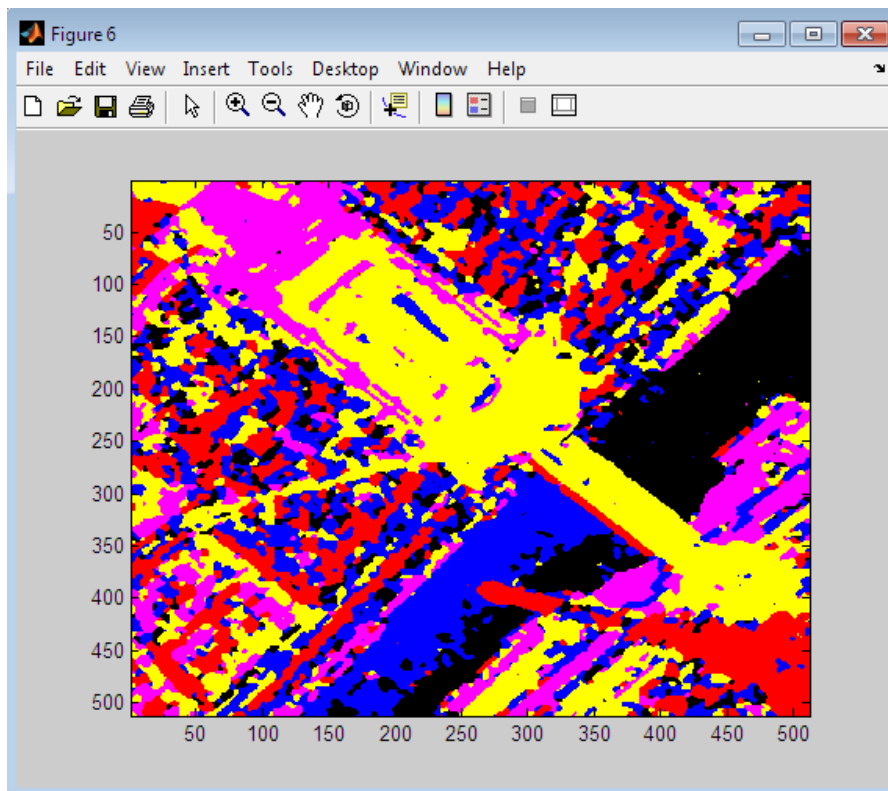
n=20



n=40

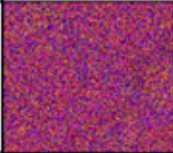
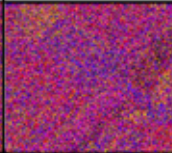
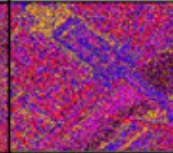
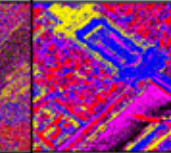
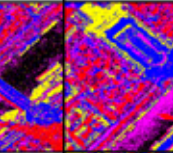
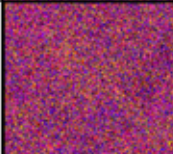
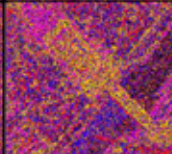
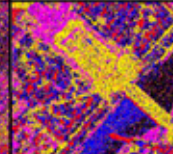
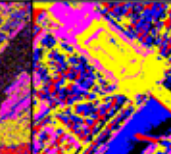
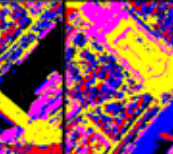

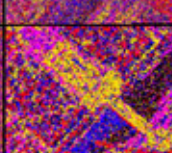
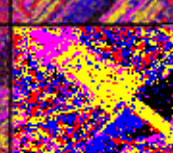
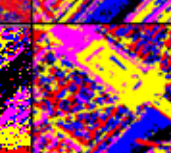
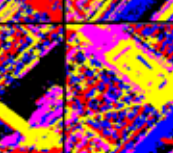


n=50

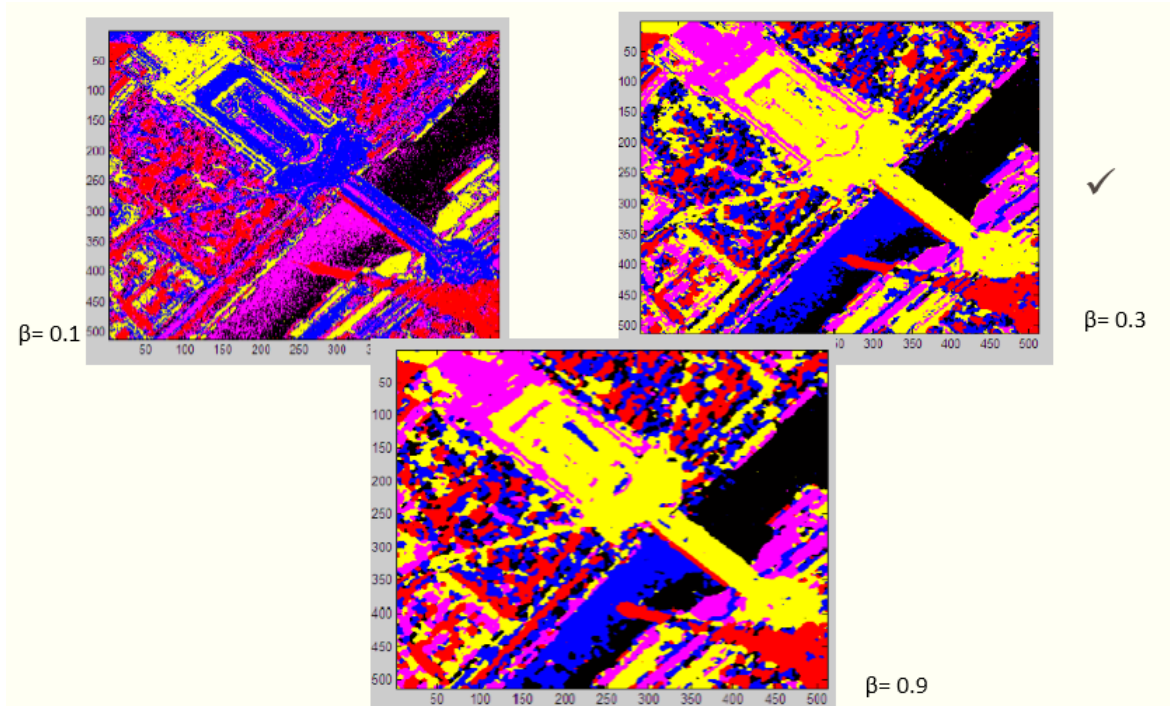


Observando que para $\beta=0.9$ el mejor resultado se encuentra en la iteración 50.

8.2 Comparación

| Beta/Iteración | 1 | 10 | 20 | 40 | 50 |
|----------------|---|---|---|--|---|
| 0.1 |  |  |  |  |  |
| 0.35 |  |  |  |  |  |
| 0.9 |  |  |  |  |  |

Comparando resultados de la iteración 50:



Observando que el mejor resultado se obtuvo con $\beta = 0.35$, debido a que con $\beta = 0.1$ las interacciones entre los píxeles es muy débil y se requiere realizar muchas iteraciones, después de la iteración número 50 los resultados no cambian significativamente.

En el caso de utilizar una $\beta = 0.95$, se tienden a saturar las regiones de los píxeles.

Por lo tanto la mejor iteración fue la número 50, con $\beta = 0.35$ y temperatura de 2.5, por ello se utilizarán estos parámetros.

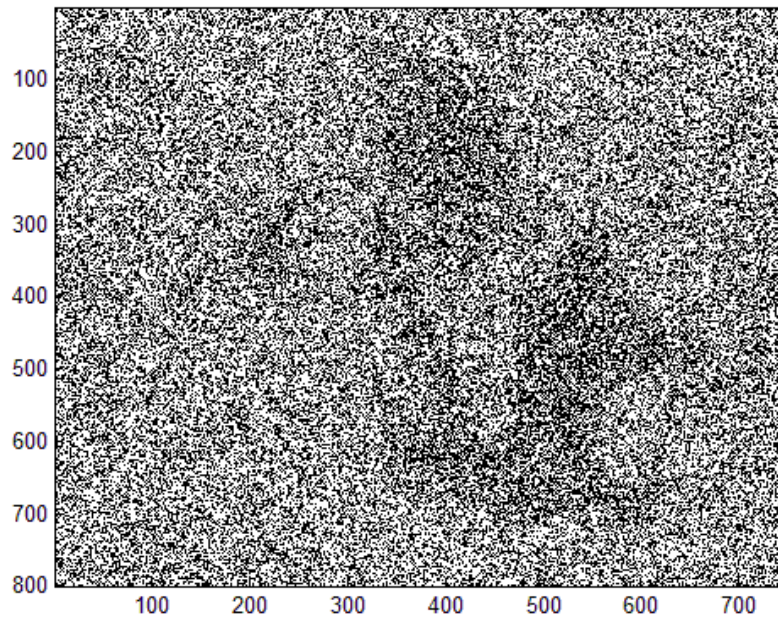
Aplicando el *esquema general* sobre una de las imágenes de la mancha de petróleo utilizando solo dos clases (petróleo, agua) [Anexo 2].

Imagen original:

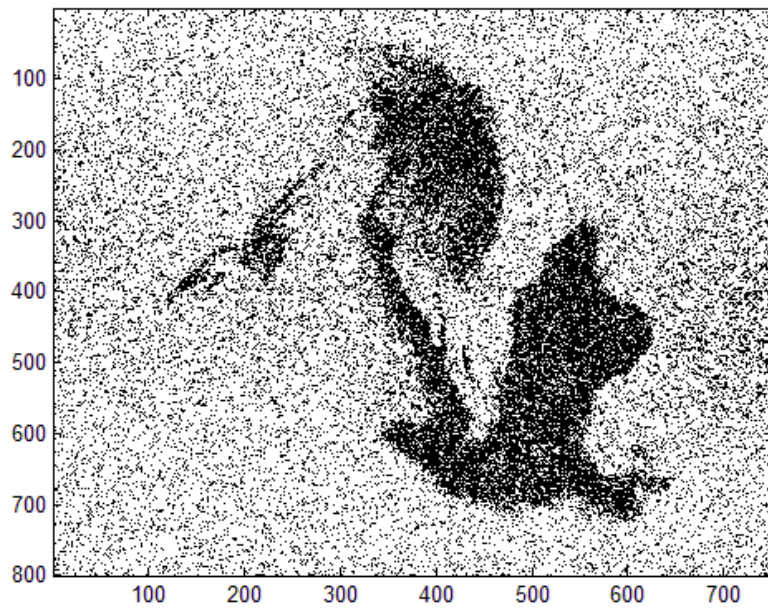


Figura 36: Imagen 0426

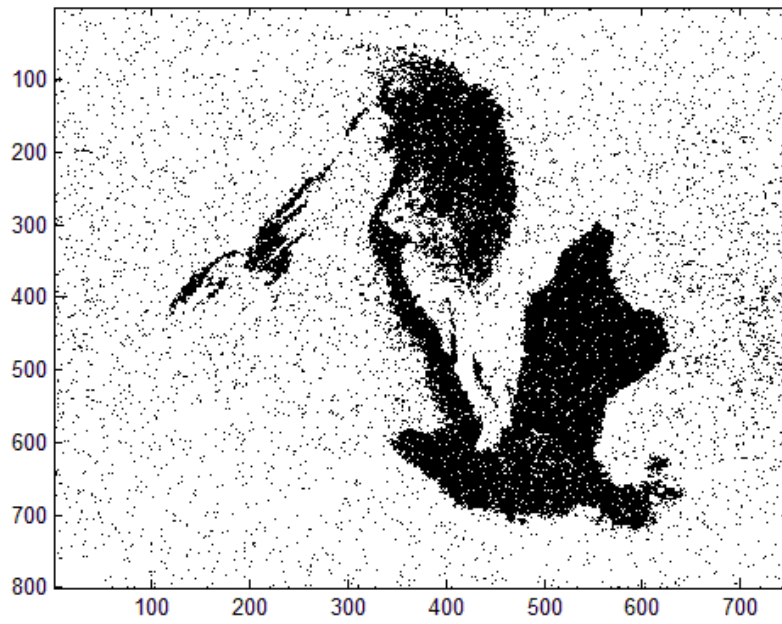
Iteración 1:



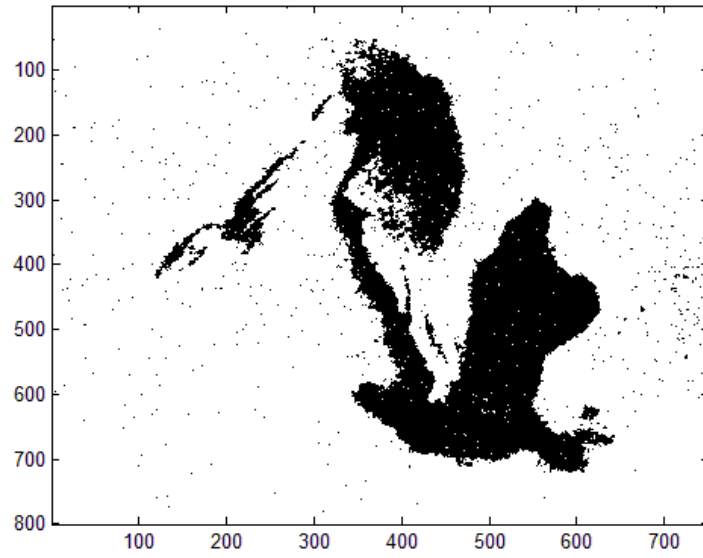
Iteración =10



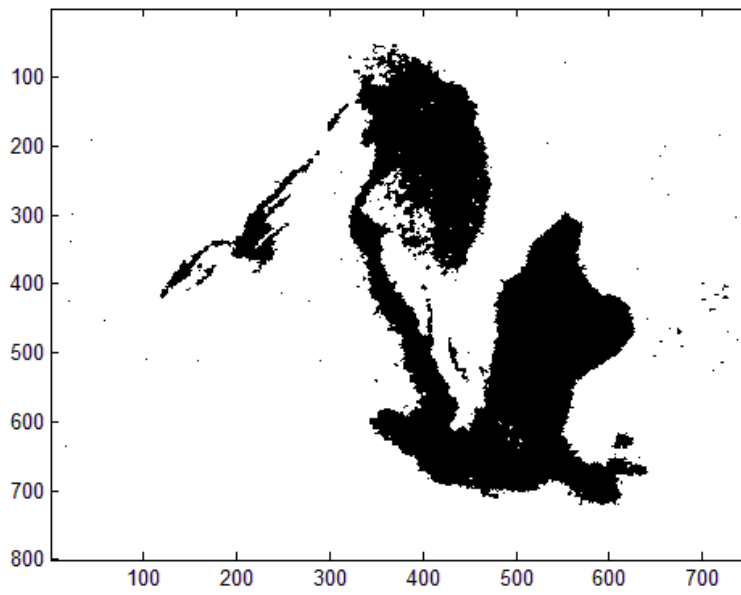
Iteración= 20



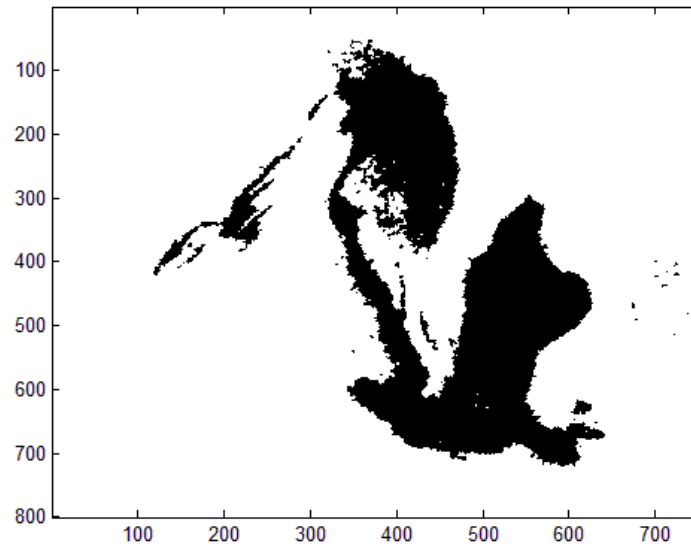
Iteración=30



Iteración= 40.



Iteración= 50



Observando que se logra un resultado aceptable con el *esquema general*, pero aún se puede determinar con mayor exactitud la superficie del petróleo y esto es gracias al *esquema propuesto* que se explicará a continuación.

8.3 imágenes optimizadas (esquema propuesto)

La primera parte del esquema propuesto, consiste en crear una imagen Y de desviaciones estándar del mismo tamaño que la imagen original X, barriendo secuencialmente la imagen X con ventanas de píxeles: 3x3, 5x5 y 7x7, la desviación estándar de cada ventana se coloca en el pixel central de la ventana (en Y).

Consecutivamente moviendo la ventana una columna y repitiendo el cálculo; al terminar el renglón de la imagen, bajar un renglón y colocarse en la primera columna y así repetir el cálculo hasta barrer toda la imagen completa (X).

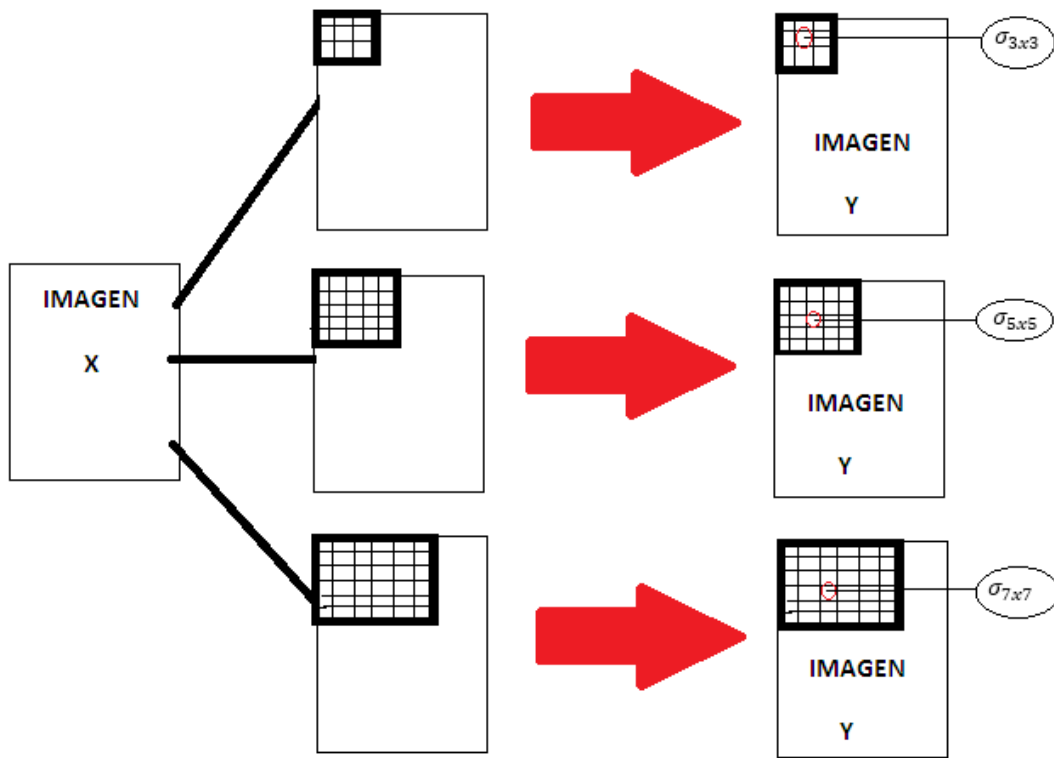


Figura 37: Visualización del esquema propuesto.

Ahora la diferencia entre el esquema general y el propuesto, es que en el propuesto:

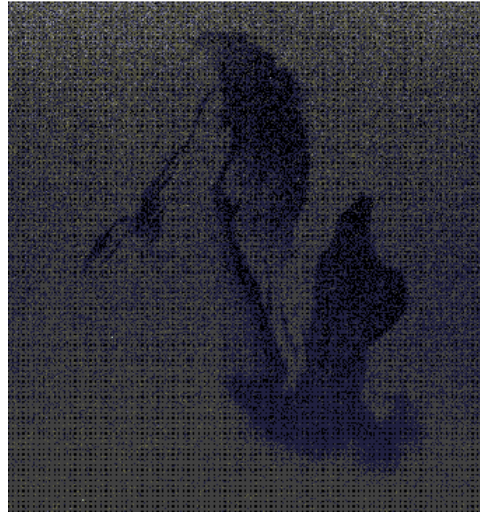
$$\begin{aligned}
 U(W_i | \sigma_{3 \times 3}) &\approx U(W_i) + U(\sigma_{3 \times 3} | W_i) \\
 U(W_i | \sigma_{5 \times 5}) &\approx U(W_i) + U(\sigma_{5 \times 5} | W_i) \\
 U(W_i | \sigma_{7 \times 7}) &\approx U(W_i) + U(\sigma_{7 \times 7} | W_i)
 \end{aligned}$$

De estas imágenes, se elegirá la óptima.

Creando las tres imágenes de desviación estándar 3x3, 5x5 y 7x7 [Anexo 3] y aplicamos el método propuesto sobre la imagen 0426.

Las imágenes de desviación estándar obtenidas por cada ventana son:

3 por 3:

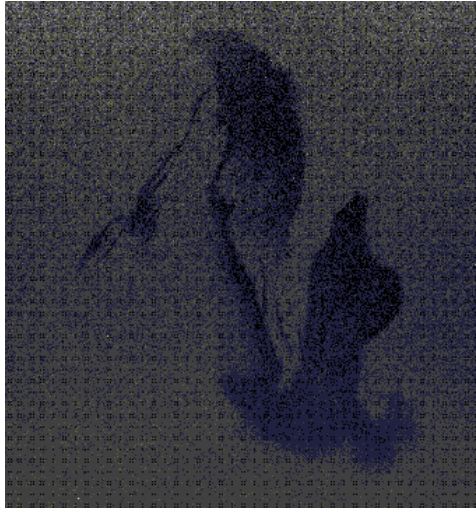


Observando como el valor de la desviación estándar se coloca en el pixel central (en este caso para la ventana 3 por 3):

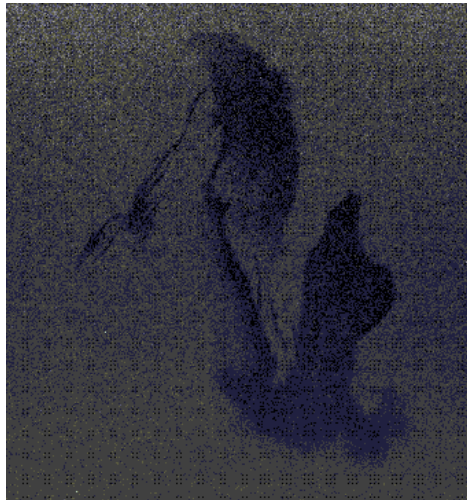
| | 1 | 2 | 3 | 4 | 5 | 6 |
|----|--------|--------|--------|--------|--------|--------|
| 1 | 0.2078 | 0.4196 | 0.2745 | 0.3333 | 0.5137 | 0.3333 |
| 2 | 0.4941 | 0.4078 | 0.5098 | 0.4706 | 0.4314 | 0.4078 |
| 3 | 0.2980 | 0.3255 | 0.3843 | 0.3451 | 0.3255 | 0.2431 |
| 4 | 0.2784 | 0.2353 | 0.4510 | 0.4824 | 0.2118 | 0.2549 |
| 5 | 0.3608 | 0.3843 | 0.3333 | 0.4431 | 0.1569 | 0.4353 |
| 6 | 0.4157 | 0.4196 | 0.5333 | 0.2902 | 0.2039 | 0.3098 |
| 7 | 0.4353 | 0.4431 | 0.3176 | 0.4549 | 0.3137 | 0.3176 |
| 8 | 0.3569 | 0.2863 | 0.3569 | 0.4039 | 0.2235 | 0.1768 |
| 9 | 0.3412 | 0.4235 | 0.3451 | 0.5843 | 0.3020 | 0.3568 |
| 10 | 0.2510 | 0.2471 | 0.2980 | 0.3451 | 0.4784 | 0.5843 |

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|--------|--------|--------|--------|--------|--------|
| 1 | 0.2078 | 0.4196 | 0.2745 | 0.3333 | 0.5137 | 0.3333 |
| 2 | 0.4941 | 0.0833 | 0.5098 | 0.4706 | 0.1038 | 0.4078 |
| 3 | 0.2980 | 0.3255 | 0.3843 | 0.3451 | 0.3255 | 0.2431 |
| 4 | 0.2784 | 0.2353 | 0.4510 | 0.4824 | 0.2118 | 0.2549 |
| 5 | 0.3608 | 0.0733 | 0.3333 | 0.4431 | 0.0788 | 0.4353 |
| 6 | 0.4157 | 0.4196 | 0.5333 | 0.2902 | 0.2039 | 0.3098 |
| 7 | 0.4353 | 0.4431 | 0.3176 | 0.4549 | 0.3137 | 0.3176 |
| 8 | 0.3569 | 0.0934 | 0.3569 | 0.4039 | 0.1237 | 0.1768 |
| 9 | 0.3412 | 0.4235 | 0.3451 | 0.5843 | 0.3020 | 0.3568 |

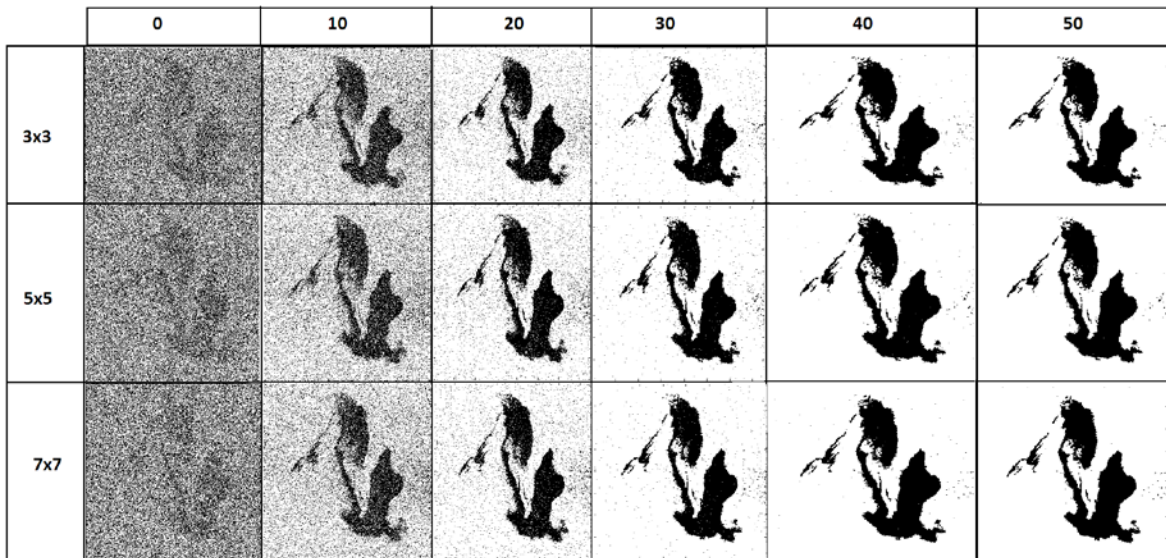
5 por 5:



7 por 7:



A continuación se resumen las corridas del programa de acuerdo a las ventanas de desviación estándar y el número de iteraciones.



El criterio de decisión de la mejor ventana consistirá en sobreponer el borde de cada una de ellas (solo la iteración 50) sobre la imagen del petróleo original (imagen 0426) [Anexo 4].

Para la ventana de 3 por 3:



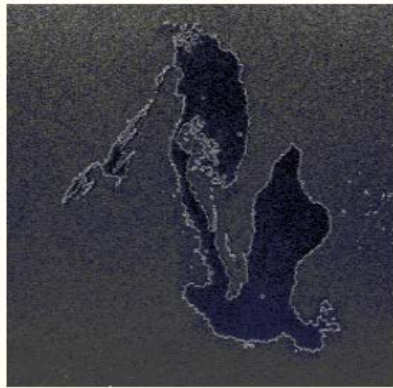
Para la ventana de 5 por 5:



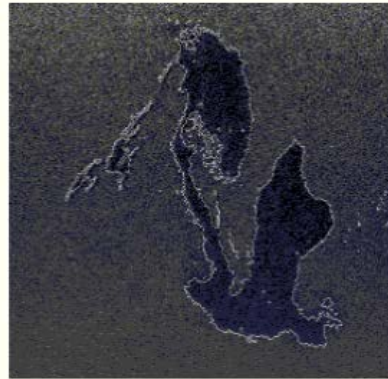
Para la ventana de 7 por 7:



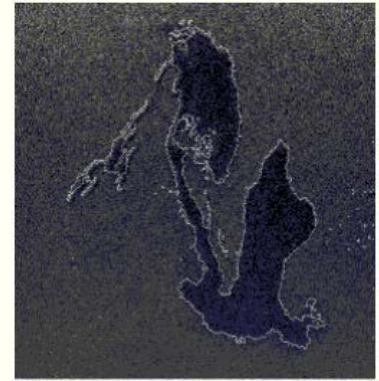
Sobreponiendo las imágenes sobre la imagen del derrame de petróleo original [Anexo 6]:



3 X 3

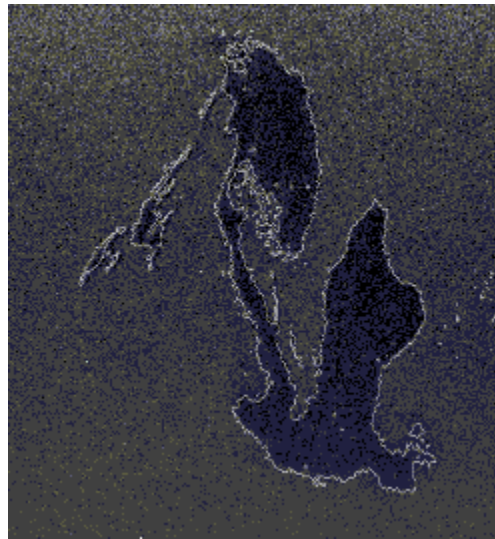


5 X 5



7 X 7

A simple vista parecen ser los mismos contornos, pero si se le aplica "zoom" a cada una de las imágenes se comprobará que la más óptima (el mejor trazado y detección) corresponde a la ventana de 5 por 5.



Procedemos a calcular el número de píxeles en color blanco (mar) y negro (mancha de petróleo) y así calcular la extensión aproximada del derrame [Anexo 7].

Se cuentan los pixeles blancos y negros de la imagen utilizada, los blancos corresponden al mar y los negros a la región de vertido de petróleo.

blanco =

487670

negro =

112330

La imagen 0426 comprende un área de 150 Km x 150 Km= 22500 km².

| | Número de Pixeles | Extensión (kilometros cuadrados) |
|------------------------------|-------------------|----------------------------------|
| pixeles blancos (Mar) | 487,670 | 18,287.625 |
| Pixeles negros (Petróleo) | 112,330 | 4,212.375 |

Así el número de pixeles totales de la imagen es $487,670 + 112,330 = 600000$, de éstos se contó la parte de pixeles que pertenece al mar (487,670) y la parte de la región de petróleo (112,330); como la extensión de la imagen en tamaño real es de 22500 km², se realiza una comparación concluyendo que:

Para la imagen 0426 la extensión de la mancha de petróleo es de aproximadamente 4,212.375 km².

Con esto concluimos el procesamiento para una de las tres imágenes a procesar.

Procedemos a realizar el procesamiento anterior a la imagen 0502:

1. Aplicar el recocido simulado (esquema general) a la imagen [Anexo 8].
2. Creando una nueva imagen Y de desviaciones estándar, construida por el recorrido de ventanas de 5 x 5 pixeles y colocando el valor de la desviación estándar al pixel central de la ventana [Anexo 9].
3. Recortando el borde de la imagen creada y sobreponiéndolo sobre la imagen original [Anexo 5 y 10].
4. Comparar el método propuesto con el método de recocido simulado general.

Las siguientes imágenes que se utilizaron corresponden al igual que la anterior al derrame petrolero de 2010.

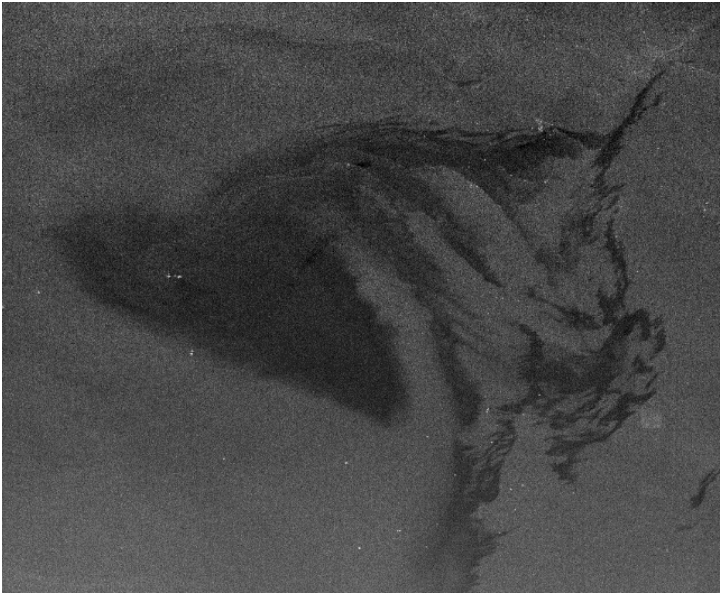


Figura 38: imagen 0502

Resultado obtenido a la iteración 50 (esquema general):

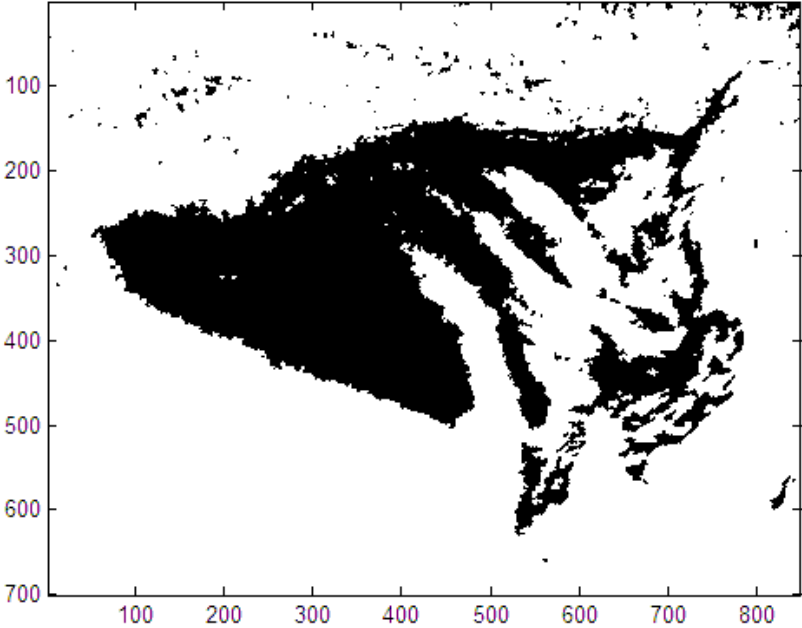
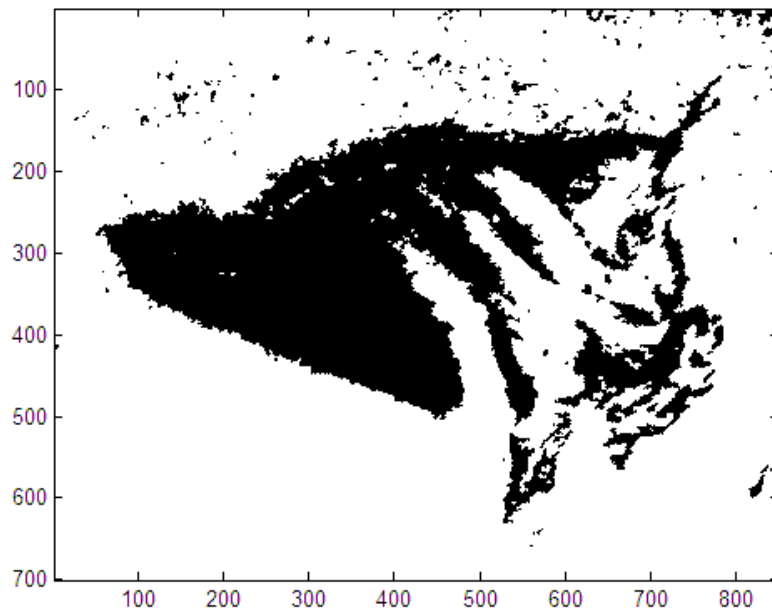


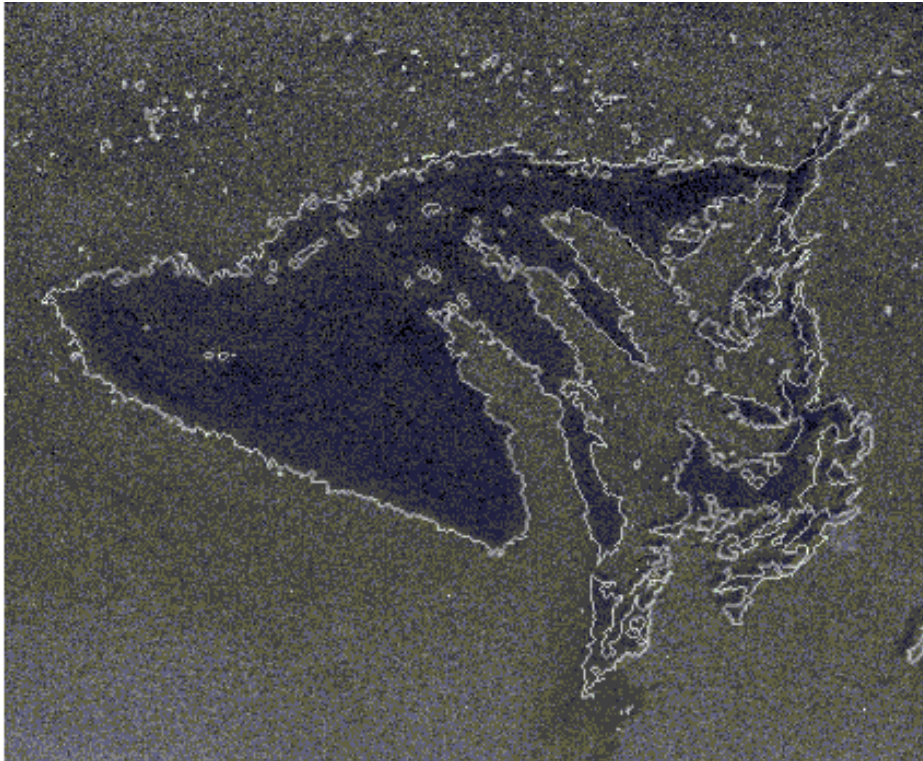
Imagen optimizada, iteración 50 (esquema propuesto):



Creación del borde:



Resultado de la sobreposición:



Contador de pixeles [Anexo 11].

```
blanco =  
429051
```

```
negro =  
165949
```

La imagen 0502 comprende un área de 150 Km x 150 Km= 22500 km².

| | Número de Pixeles | Extensión (kilometros cuadrados) |
|------------------------------|-------------------|----------------------------------|
| pixeles blancos (Mar) | 429,051 | 16,224.61 |
| Pixeles negros (Petróleo) | 165,949 | 6,275.38 |

Así el número de pixeles totales de la imagen es $429051 + 165949 = 595000$, de éstos se contó la parte de pixeles que pertenece al mar (429051) y la parte de la región de petróleo (165949); como la extensión de la imagen en tamaño real es de 22500 km², se realiza una comparación concluyendo que:

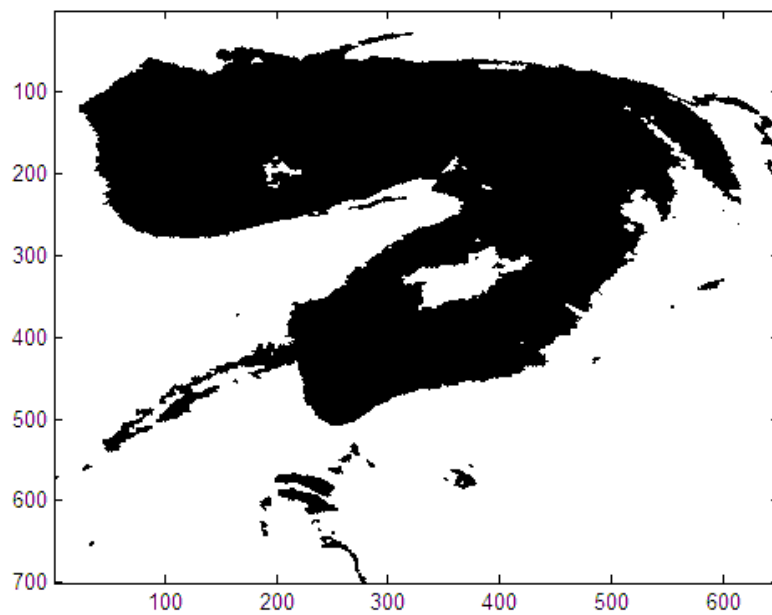
La extensión de la mancha de petróleo de la imagen 0502 es de aproximadamente 6,275.38 km².

Y por último realizaremos el procesamiento para la imagen 0509.

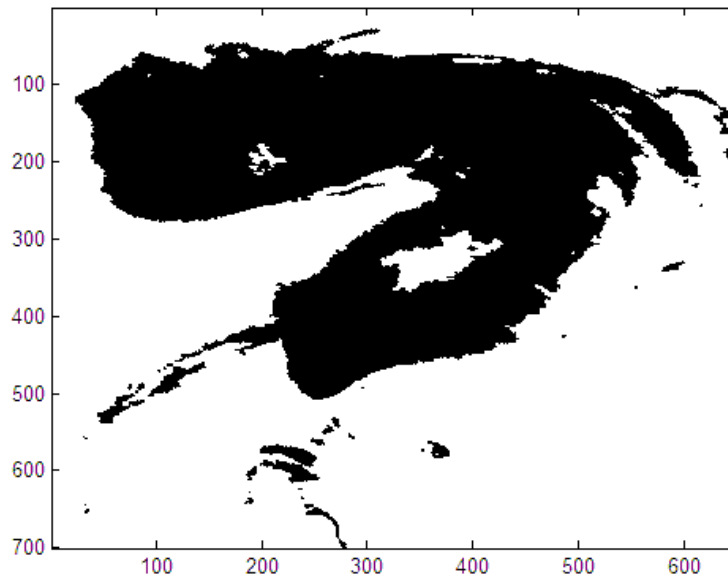


Figura 39: imagen 0509

Resultado de binarizar la imagen (iteración 50) esquema general [Anexo 12]:



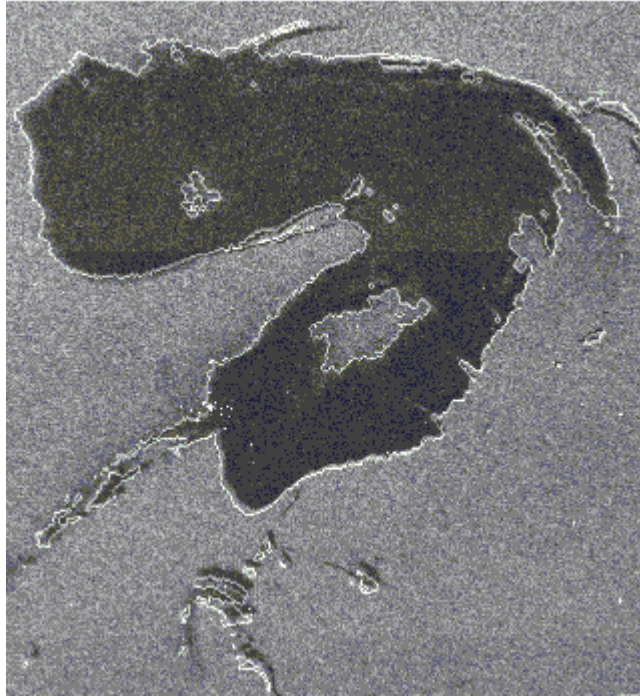
Resultado de la imagen optimizada (esquema propuesto) [Anexo 13]:



Detección de los bordes:



Sobreponiendo las imágenes [Anexo 14]:



Contando los pixeles [Anexo 15]:

```
blanco =  
      289374
```

```
negro =  
      165626
```

La imagen 0509 comprende un área de 150 Km x 150 Km= 22500 km².

| | Número de Pixeles | Extensión (kilometros cuadrados) |
|--------------------------------------|-------------------|----------------------------------|
| pixeles blancos (Mar) | 289,374 | 14,309.70 |
| Pixeles negros (Petróleo) | 165,626 | 8,190.29 |

Así el número de pixeles totales de la imagen es $289374 + 164626 = 455000$, de éstos se contó la parte de pixeles que pertenece al mar (289,374) y la parte de la región de petróleo (165,626); como la extensión de la imagen en tamaño real es de 22500 km^2 , se realiza una comparación concluyendo que:

La extensión de la mancha de petróleo es de aproximadamente $8,190.29 \text{ km}^2$.

8.4 Comparación

| imagen | | Número de Pixeles | Extensión (kilometros cuadrados) |
|--------|------------------------------|-------------------|----------------------------------|
| 0426 | Pixeles negros (Petróleo) | 112,330 | 4,212.375 |
| 0502 | Pixeles negros (Petróleo) | 165,949 | 6,275.38 |
| 0509 | Pixeles negros (Petróleo) | 165,626 | 8,190.29 |

Comparando las tres imágenes capturadas por el satélite Envisat (éstas corresponden al mismo vertido de petróleo y fueron tomadas en diferentes fechas), se concluye que la extensión de petróleo alcanza un nivel muy grande que es de 8,190.29 km² para la fecha de mayo. Recalcando que el objetivo del presente tema de tesis es brindar herramientas para el cálculo de la extensión de petróleo derramado y **NO** en dar una solución específica a la remoción de dicho derrame ya que esta fuera de nuestros alcances.

9. Conclusiones

- Se logró optimizar las imágenes obtenidas por el satélite Envisat, de esta manera se determinó la extensión aproximada de petróleo para cada una de ellas.
- Los resultados se obtuvieron a partir de un proceso de segmentación en clases realizando investigación sobre la teoría de Bayes en técnicas de segmentación y esquemas estocásticos por estimación de máximo a posteriori para después aplicar el método de recocido simulado, al final la optimización provino de aplicarlo a la creación de una nueva imagen construida a partir de la imagen original con la desviación estándar de cada ventana colocada en el pixel central de cada una de ellas, concluyendo que la mejor ventana fue la de 5 x 5 pixeles.
- Otro método que se puede realizar es segmentar la imagen en 3 clases aplicando el mismo método optimizado, con esto se lograría una mejor precisión en el cálculo y además se obtendría la detección de 2 tipos de combustibles vertidos en el mar.
- Así dependiendo del número de clases en que se segmente la imagen se pueden detectar diversos tipos de combustibles y su respectiva extensión.
- Este método puede ser utilizado para los futuros accidentes marinos de derrame de petróleo que se presenten.

Bibliografía

- [1] Rincón Luis, Introducción a los procesos estocásticos, Departamento de Matemáticas, Facultad de Ciencias UNAM.
- [2] Canavos C. George, Probabilidad y estadística (aplicaciones y métodos).
- [3] Devore L. Jay, Probabilidad y estadística para ingeniería y ciencias, sexta edición, California Polytechnic State University.
- [4] Campos aleatorios de Markov, 5.3 Nociones topológicas, recuperado 09/10/13 de: <http://www.ptolomeo.unam.mx:8080/xmlui/bitstream/handle/132.248.52.100/437/A7.pdf?sequence=7>
- [5] Caiafa César Federico, Marzo 1996, Procesos Aleatorios Bidimensionales.
- [6] Zoltan Kato, Tesis aplicación a la segmentación de imágenes 1994.
- [7] European Space Agency, Earth Online, recuperado el 22 de Octubre de 2014, de <https://earth.esa.int/web/guest/missions/esa-operational-eo-missions/envisat>
- [8] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. Equation of state calculations by fast computing machines. J. of chem, Physic, Vol.21, pp 1087-1092, 1953.
- [9] Z. Kato, J. Zerubia, and M. Berthod. Bayesian image classification using markov random fields. In A. M. D. G. Demoments, editor, Maximum Entropy and Bayesian Methods, Kluwer Academic Publisher, 1993.
- [10] S. German and D. German. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. IEEE Trans, on Pattern Analysis and Machine Intelligence, 1984.

Índice de figuras

| | |
|---|----|
| Figura 1: Variables aleatorias..... | 6 |
| Figura 2: Gráfica de una distribución gaussiana..... | 10 |
| Figura 3: Componentes de un sistema de percepción remota..... | 13 |
| Figura 4: Malla para textura..... | 14 |
| Figura 5: Malla para contorno..... | 15 |
| Figura 6: Vecindario de orden 1 para el modelo de contorno..... | 16 |
| Figura 7: Sistema 4- vecinos (modelo de textura)..... | 17 |
| Figura 8: Sistema 8-vecinos (modelo de textura)..... | 17 |
| Figura 9: Tipos de cliques para sistemas de orden 1 (modelo de textura)..... | 18 |
| Figura 10: Tipos de cliques para el vecindario de orden 1 (modelo de contorno)..... | 18 |
| Figura 11: Tipos de cliques para sistemas de orden 2 en el modelo de textura..... | 19 |
| Figura 12: Modelo de Ising en una dimensión..... | 23 |
| Figura 13: Modelo de Potts..... | 23 |
| Figura 14: Representación gráfica para el cálculo de la excentricidad..... | 24 |
| Figura 15: Representación gráfica del periastro..... | 25 |
| Figura 16: Instrumentos del satélite ENVISAT..... | 31 |
| Figura 17: Logotipo de la herramienta Nest..... | 32 |
| Figura 18: Interfaz gráfica de la herramienta Nest..... | 33 |
| Figura 19: Búsqueda de datos del producto a elegir..... | 34 |
| Figura 20: Búsqueda de datos del Instrumento “ENVISAT ASAR” | 34 |
| Figura 21: Imagen desplegada con el programa EOLI-SA..... | 35 |
| Figura 22: Vista de navegación..... | 36 |
| Figura 23: Movimiento de partículas de acuerdo al algoritmo de metrópolis..... | 40 |
| Figura 24: Imagen original X “parís.bmp” | 47 |
| Figura 25: Imagen en grises (izquierda) e imagen dividida en clases (derecha)..... | 47 |
| Figura 26: Funciones de energía..... | 48 |

| | |
|--|----|
| Figura 27: Cálculo de la energía $U(X W_i)$ a través de la distribución gaussiana..... | 49 |
| Figura 28: Esquema iterativo..... | 50 |
| Figura: 29 Logo de Matlab..... | 53 |
| Figura 30: Interfaz gráfica de la herramienta de trabajo Matlab..... | 53 |
| Figura 31,32,33,34 y 35: Selección de clase..... | 54 |
| Figura 36: Imagen 0426..... | 65 |
| Figura 37: Visualización del esquema propuesto..... | 69 |
| Figura 38: imagen 0502..... | 76 |
| Figura 39: imagen 0509..... | 80 |

Anexos

Anexo 1:

%%%%%%%% Algoritmo de recocido simulado esquema general %%%%%%%%%

```
img='paris.bmp';
beta=0.1; %% Beta cambia a 0.35 y 0.9 %%%%
iter=50;
nclas=5;
[imagen,map]=imread(img);
imagen=im2double(imagen);
area1=imcrop(imagen);
area2=imcrop(imagen);
area3=imcrop(imagen);
area4=imcrop(imagen);
area5=imcrop(imagen);
m(1:nclas)=0;
v(1:nclas)=0;
[m(1),v(1)]=fmv(area1);
[m(2),v(2)]=fmv(area2);
[m(3),v(3)]=fmv(area3);
[m(4),v(4)]=fmv(area4);
[m(5),v(5)]=fmv(area5);
close;
v=sqrt(v);
copia=imagen;
[ren,col]=size(imagen);
pxw1=ones(ren,col);
pxw2=ones(ren,col);
pxw3=ones(ren,col);
pxw4=ones(ren,col);
pxw5=ones(ren,col);
pxw(:, :, 1)=pxw1;
pxw(:, :, 2)=pxw2;
pxw(:, :, 3)=pxw3;
pxw(:, :, 4)=pxw4;
pxw(:, :, 5)=pxw5;
for i=1:ren
for j=1:col
        X=imagen(i,j);
        [a indice] = max(normpdf(X,m,v));
        pxw(i,j,indice)=0;
end
end
y=unidrnd(nclas,ren,col);
t=2.5;
ren
col
for n=1:iter
        n
for i=1:ren
for j=1:col
        actual=y(i,j);
        a=i-1;b=j-1;c=i+1;d=j+1;
        eleccion=ceil(random('unif',0,nclas));
        act=0;elec=0;
```

```

for w=a:c
for z=b:d
if (w>0&&w<ren&&z>0&&z<col)
if actual==y(w,z)
act=act+1;
end
if eleccion==y(w,z)
elec=elec+1;
end
end
end
E_actual=-beta*act;
E_eleccion=-beta*elec;
if (actual>=1&&actual<=nclas)
E_actual=E_actual+pxw(i,j,actual);
end
if (eleccion>=1&&eleccion<=nclas)
E_eleccion=E_eleccion+pxw(i,j,eleccion);
end
if E_eleccion<=E_actual
y(i,j)=eleccion;
else
p=exp(-((E_eleccion-E_actual)/t));
theta=rand(1);
if theta<p
y(i,j)=eleccion;
else
y(i,j)=actual;
end
end
end
end
t=t*0.95;
if n==1
figure(2);
mapa=[0 0 0 ; 1 0 0 ; 0 0 1 ; 1 1 0; 1 0 1 ; 0 1 1 ; 1 1 1];
colormap(mapa);
image(y);
elseif n==10
figure(3);
mapa=[0 0 0 ; 1 0 0 ; 0 0 1 ; 1 1 0; 1 0 1 ; 0 1 1 ; 1 1 1];
colormap(mapa);
image(y);
elseif n==20
figure(4);
mapa=[0 0 0 ; 1 0 0 ; 0 0 1 ; 1 1 0; 1 0 1 ; 0 1 1 ; 1 1 1];
colormap(mapa);
image(y);
elseif n==40
figure(5);
mapa=[0 0 0 ; 1 0 0 ; 0 0 1 ; 1 1 0; 1 0 1 ; 0 1 1 ; 1 1 1];
colormap(mapa);
image(y);
elseif n==50
figure(6);

```

```

mapa=[0 0 0 ; 1 0 0 ; 0 0 1 ; 1 1 0; 1 0 1 ; 0 1 1 ; 1 1 1];
    colormap(mapa);
image(y);
end
end
for nclas=1:5
for i=1:ren
for j=1:col
if y(i,j)==nclas
copia(i,j)=m(nclas);
else
                e=0;
end
end
end
end
% Resultados
% figure(2);
% colormap(map);
% image(copia);
% mapa=[0 0 0 ; 1 0 0 ; 0 0 1 ; 1 1 0; 1 0 1 ; 0 1 1 ; 1 1 1];
% figure(3);
% colormap(mapa);
% image(y);

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%función a crear%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fmv.m

```

```

function [media,varianza] = fmv(x)
media = mean2(x);
varianza = mean2((x-media).^2);

```

Anexo 2:

%%%%%%%% código para el esquema general aplicado a la imagen 0426 %%%%%%%%%%

```
img='mw0426.bmp';
nclas=2;
beta=0.35;
iter=51;
[imagen,map]=imread(img);
imagen=im2double(imagen);
[ren,col]=size(imagen);
copia=imagen;

area1=imcrop(copia,[516 512 26 90]);
area2=imcrop(copia,[700 400 30 300]);
m(1:nclas)=0;
v(1:nclas)=0;
[m(1),v(1)]=fmv(area1);
[m(2),v(2)]=fmv(area2);
close;
v=sqrt(v);
pxw1=ones(ren,col);pxw2=ones(ren,col);
pxw(:,1)=pxw1;
pxw(:,2)=pxw2;
for i=1:ren
for j=1:col
k=copia(i,j);
[a_3 indice] = max(normpdf(k,m,v));
pxw(i,j,indice)=0;
end
end
y=unidrnd(nclas,ren,col);
t=2.5;

for n=1:iter
n
for i=1:ren
for j=1:col
actual=y(i,j);
a=i-1;b=j-1;c=i+1;d=j+1;
eleccion=ceil(random('unif',0,nclas));
act=0;elec=0;
for w=a:c
for z=b:d
if (w>0&&w<ren&&z>0&&z<col)
if actual==y(w,z)
act=act+1;
end
if eleccion==y(w,z)
```

```

        elec=elec+1;
    end
end
end
end
E_actual=-beta*act;
E_eleccion=-beta*elec;
if (actual>=1&&actual<=nclas)
    E_actual=E_actual+pxw(i,j,actual);
end
if (eleccion>=1&&eleccion<=nclas)
    E_eleccion=E_eleccion+pxw(i,j,eleccion);
end
if E_eleccion<=E_actual
    y(i,j)=eleccion;
else
    p=exp(-((E_eleccion-E_actual)/t));
theta=rand(1);
    if theta<p
        y(i,j)=eleccion;
    else
        y(i,j)=actual;
    end
end
end
end
end
t=t*0.95;

if n==1
    figure(3);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
elseif n==10
    figure(4);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
elseif n==20
    figure(5);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
elseif n==30
    figure(6);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
elseif n==40

```



```
    figure(7);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
    elseif n==50
figure(8);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
end
end
for nclas=1:2
    for i=1:ren
        for j=1:col
            if y(i,j)==nclas
copia(i,j)=m(nclas);
            else
                e=0;
            end
        end
    end
end
end
end
```

Anexo 3:

%%%%%%%% Código ventanas 3x3 , 5x5 y 7x7 %%%%%%%%%%

```
img='mw0426.bmp';
```

```
nclas=2;
```

```
beta=0.35;
```

```
iter=51;
```

```
[imagen,map]=imread(img);
```

```
imagen=im2double(imagen);
```

```
[ren,col]=size(imagen);
```

```
copia=imagen;
```

```
%%% 3 por 3 %%%%
```

```
for i=2:3:ren
```

```
    for j=2:3:col
```

```
        if (i>0&&i<=798&&j>0&&j<=749)
```

```
l=imcrop(imagen,[j i 2 2]);
```

```
Media=mean2(l);
```

```
varianza = mean2((l-Media).^2);
```

```
d=sqrt(varianza);
```

```
copia(i,j)=d;
```

```
    end
```

```
end
```

```
end
```

```
%%%%%%%%
```

```
%%%5 por 5 %%%%
```

```
copia=imagen;
```

```
for i=3:5:ren
```

```
    for j=3:5:col
```

```
        if(i>0&&i<=800&&j>0&&j<=750)
```

```
l=imcrop(imagen,[j i 4 4]);
```

```
Media=mean2(l);
```

```
varianza = mean2((l-Media).^2);
```

```
d=sqrt(varianza);
```

```
copia(i,j)=d;
```

```
    end
```

```
end
```

```
end
```

```
%%%%%%%%%
```

```
%%% 7 por 7 %%%%
```

```
copia=imagen;
```

```
for i=4:7:ren
```

```
    for j=4:7:col
```

```
        if (i>0&&i<=798&&j>0&&j<=749)
```

```

I=imcrop(imagen,[j i 6 6]);
Media=mean2(I);
varianza = mean2((I-Media).^2);
d=sqrt(varianza);
copia(i,j)=d;
    end
    end
end

%%%%%%%%%%

area1=imcrop(copia,[516 512 26 90]);
area2=imcrop(copia,[700 400 30 300]);
m(1:nclas)=0;
v(1:nclas)=0;
[m(1),v(1)]=fmv(area1);
[m(2),v(2)]=fmv(area2);
close;
v=sqrt(v);
pxw1=ones(ren,colpxw2=ones(ren,col));
pxw(:,,1)=pxw1;
pxw(:,,2)=pxw2;
for i=1:ren
for j=1:col
    k=copia(i,j);
    [a_3 indice] = max(normpdf(k,m,v));
pxw(i,j,indice)=0;
    end
end
y=unidrnd(nclas,ren,col);
t=2.5;

for n=1:iter
    n
    for i=1:ren
for j=1:col
    actual=y(i,j);
    a=i-1;b=j-1;c=i+1;d=j+1;
    eleccion=ceil(random('unif',0,nclas));
act=0;elec=0;
    for w=a:c
        for z=b:d
if (w>0&&w<ren&&z>0&&z<col)
if actual==y(w,z)
act=act+1;
                end
            end
        end
    end
end

```

```

        if eleccion==y(w,z)
elec=elec+1;
        end
        end
        end
        end
E_actual=-beta*act;
E_eleccion=-beta*elec;
if (actual>=1&&actual<=nclas)
    E_actual=E_actual+pxw(i,j,actual);
end
if (eleccion>=1&&eleccion<=nclas)
    E_eleccion=E_eleccion+pxw(i,j,eleccion);
end
if E_eleccion<=E_actual
    y(i,j)=eleccion;
else
    p=exp(-((E_eleccion-E_actual)/t));
theta=rand(1);
    if theta<p
        y(i,j)=eleccion;
    else
        y(i,j)=actual;
end
    end
    end
    end
t=t*0.95;

if n==1
    figure(3);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
elseif n==10
    figure(4);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
elseif n==20
    figure(5);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
elseif n==30
    figure(6);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);

```

```
elseif n==40
    figure(7);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
elseif n==50
figure(8);
mapa=[0 0 0 ; 1 1 1];
colormap(mapa);
image(y);

end
end

for nclas=1:2
    for i=1:ren
        for j=1:col
            if y(i,j)==nclas
copia(i,j)=m(nclas);
else
    e=0;
end
end
end
```

Anexo 5:

%%%% código utilizado para detección de bordes utilizado para las tres imágenes %%%%%%

```
IM=imread('50.bmp');
imshow(IM);
%Elemento Estructurante
B=[0 1 0;1 1 1;0 1 0];
%Dilatación
IM2=imdilate(IM,B);
%Borde
BORDE=double(IM2)-double(IM);
imshow(BORDE);
```

Anexo 6:

%%%% código de sobreposición del borde la imagen sobre la imagen original 0426 %%%%

```
A = imread('BORDE.bmp');
C = imread('mw0502.bmp');
B=imresize(A,[800 750]);
d= C+0.5*B;
figure(1);
imshow(d)
```

Anexo 7:

%%%% Contador de pixeles imagen 0426 %%%%%%

```
IM=imread('50.bmp');
B=imresize(IM,[800 750]);
negro=0;
blanco=0;
for i=1:700
    for j=1:650
        k=B(i,j);
        if (k==0)
            negro=negro+1;
        else (k==255)
            blanco=blanco+1;
        end
    end
end
blanco
negro
```

Anexo 8:

%%%% recocido simulado esquema general aplicado a la imagen 0502%%%

```
img='mw0502.bmp';
nclas=2;
beta=0.35;
iter=51;
[imagen,map]=imread(img);
imagen=im2double(imagen);
[ren,col]=size(imagen);
copia=imagen;
area1=imcrop(copia,[131 237 130 118]);
area2=imcrop(copia,[102 90 100 89]);
m(1:nclas)=0;
v(1:nclas)=0;
[m(1),v(1)]=fmv(area1);
[m(2),v(2)]=fmv(area2);
close;
v=sqrt(v);
pxw1=ones(ren,col);
pxw2=ones(ren,col);
pxw(:,1)=pxw1;
pxw(:,2)=pxw2;
for i=1:ren
    for j=1:col
        k=copia(i,j);
        [a_3 indice] = max(normpdf(k,m,v));
        pxw(i,j,indice)=0;
    end
end
y=unidrnd(nclas,ren,col);
t=2.5;
for n=1:iter
    n
    for i=1:ren
        for j=1:col
            actual=y(i,j);
            a=i-1;b=j-1;c=i+1;d=j+1;
            eleccion=ceil(random('unif',0,nclas));
            act=0;elec=0;
            for w=a:c
                for z=b:d
                    if (w>0&&w<ren&&z>0&&z<col
                        if actual==y(w,z)
                            act=act+1;
                        end
                    if eleccion==y(w,z)
                        elec=elec+1;
                    end
                end
            end
        end
    end
end
```

```

        end
end
    end
    E_actual=-beta*act;
    E_eleccion=-beta*elec;
    if (actual>=1&&actual<=nclas)
        E_actual=E_actual+pxw(i,j,actual);
    end
    if (eleccion>=1&&eleccion<=nclas)
        E_eleccion=E_eleccion+pxw(i,j,eleccion);
    end
    if E_eleccion<=E_actual
        y(i,j)=eleccion;
    else
        p=exp(-((E_eleccion-E_actual)/t));
        theta=rand(1);
        if theta<p
            y(i,j)=eleccion;
        else
            y(i,j)=actual;
        end
    end
end
end
end
t=t*0.95;
if n==1
    figure(3);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
elseif n==10
    figure(4);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
elseif n==20
    figure(5);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
elseif n==30
    figure(6);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
elseif n==40
    figure(7);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);

```



```
    image(y);
    elseif n==50
    figure(8);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
end
end
for nclas=1:2
    for i=1:ren
        for j=1:col
            if y(i,j)==nclas
                copia(i,j)=m(nclas);
            else
                e=0;
            end
        end
    end
end
end
```

Anexo 9:

%%%% código del método propuesto utilizando ventanas de 5 x 5 a la imagen 0502 %%%%

```
img='mw0502.bmp';
nclas=2;
beta=0.35;
iter=51;
[imagen,map]=imread(img);
imagen=im2double(imagen);
[ren,col]=size(imagen);
copia=imagen;
for i=3:5:ren
    for j=3:5:col
        if (i>0&&i<=700&&j>0&&j<=850)
            I=imcrop(imagen,[j i 4 4]);
            Media=mean2(I);
            varianza = mean2((I-Media).^2);
            d=sqrt(varianza);
            copia(i,j)=d;
        end
    end
end
area1=imcrop(copia,[131 237 130 118]);
area2=imcrop(copia,[102 90 100 89]);
m(1:nclas)=0;
v(1:nclas)=0;
[m(1),v(1)]=fmv(area1);
[m(2),v(2)]=fmv(area2);
close;
v=sqrt(v);
pxw1=ones(ren,col);
pxw2=ones(ren,col);
pxw(:,1)=pxw1;
pxw(:,2)=pxw2;
for i=1:ren
    for j=1:col
        k=copia(i,j);
        [a_3 indice] = max(normpdf(k,m,v));
        pxw(i,j,indice)=0;
    end
end
y=unidrnd(nclas,ren,col);
t=2.5;
for n=1:iter
    n
    for i=1:ren
        for j=1:col
```

```

actual=y(i,j);
a=i-1;b=j-1;c=i+1;d=j+1;
eleccion=ceil(random('unif',0,nclas));
act=0;elec=0;
for w=a:c
    for z=b:d
        if (w>0&&w<ren&&z>0&&z<col
            if actual==y(w,z)
                act=act+1;
            end
            if eleccion==y(w,z)
                elec=elec+1;
            end
        end
    end
end
E_actual=-beta*act;
E_eleccion=-beta*elec;
if (actual>=1&&actual<=nclas)
    E_actual=E_actual+pxw(i,j,actual);
end
if (eleccion>=1&&eleccion<=nclas)
    E_eleccion=E_eleccion+pxw(i,j,eleccion);
end
if E_eleccion<=E_actual
    y(i,j)=eleccion;
else
    p=exp(-((E_eleccion-E_actual)/t));
    theta=rand(1);
    if theta<p
        y(i,j)=eleccion;
    else
        y(i,j)=actual;
    end
end
end
end
t=t*0.95;
if n==1
    figure(3);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
elseif n==10
    figure(4);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
end

```

```

        elseif n==20
            figure(5);
            mapa=[0 0 0 ; 1 1 1];
            colormap(mapa);
            image(y);
            elseif n==30
                figure(6);
                mapa=[0 0 0 ; 1 1 1];
                colormap(mapa);
                image(y);
            elseif n==40
                figure(7);
                mapa=[0 0 0 ; 1 1 1];
                colormap(mapa);
                image(y);
                elseif n==50
                    figure(8);
                    mapa=[0 0 0 ; 1 1 1];
                    colormap(mapa);
                    image(y);
                end
            end
        end
    for nclas=1:2
        for i=1:ren
            for j=1:col
                if y(i,j)==nclas
                    copia(i,j)=m(nclas);
                else
                    e=0;
                end
            end
        end
    end
end
end
end

```

Anexo 10:

%%% sobreposición del borde con la imagen original 0502%%%

```

A = imread('BORDE.bmp');
C = imread('mw0502.bmp');
B=imresize(A,[700 850]);
d= C+0.5*B;
figure(1);
imshow(d)

```

Anexo 11:

%%% contador de pixels imagen 0502 %%%

```
IM=imread('50.bmp');
B=imresize(IM,[700 850]);
negro=0;
blanco=0;
for i=1:700
    for j=1:850
        k=B(i,j);
        if (k==0)
            negro=negro+1;
        else (k==255)
            blanco=blanco+1;
        end
    end
end
blanco
negro
```

Anexo 12:

%%% recocido simulado esquema general aplicado a la imagen 0509 %%%

```
img='mw0509.bmp';
nclas=2;
beta=0.35;
iter=51;
[imagen,map]=imread(img);
imagen=im2double(imagen);
copia=imagen;
[ren,col]=size(imagen);
area1=imcrop(imagen,[430 220 60 60]);
area2=imcrop(imagen,[560 592 90 62]);
m(1:nclas)=0;
v(1:nclas)=0;
[m(1),v(1)]=fmv(area1);
[m(2),v(2)]=fmv(area2);
close;
v=sqrt(v);
pxw1=ones(ren,col);
pxw2=ones(ren,col);
pxw(:,1)=pxw1;
pxw(:,2)=pxw2;
for i=1:ren
    for j=1:col
        k=copia(i,j);
```

```

        [a_3 indice] = max(normpdf(k,m,v));
        pxw(i,j,indice)=0;
    end
end
y=unidrnd(nclas,ren,col);
t=2.5;
for n=1:iter
    n
    for i=1:ren
        for j=1:col
            actual=y(i,j);
            a=i-1;b=j-1;c=i+1;d=j+1;
            eleccion=ceil(random('unif',0,nclas));
            act=0;elec=0;
            for w=a:c
                for z=b:d
                    if (w>0&&w<ren&&z>0&&z<col)
                        if actual==y(w,z)
                            act=act+1;
                        end
                        if eleccion==y(w,z)
                            elec=elec+1;
                        end
                    end
                end
            end
            end
            E_actual=-beta*act;
            E_eleccion=-beta*elec;
            if (actual>=1&&actual<=nclas)
                E_actual=E_actual+pxw(i,j,actual);
            end
            if (eleccion>=1&&eleccion<=nclas)
                E_eleccion=E_eleccion+pxw(i,j,eleccion);
            end
            if E_eleccion<=E_actual
                y(i,j)=eleccion;
            else
                p=exp(-((E_eleccion-E_actual)/t));
                theta=rand(1);
                if theta<p
                    y(i,j)=eleccion;
                else
                    y(i,j)=actual;
                end
            end
        end
    end
end
t=t*0.95;
if n==1

```

```

    figure(3);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
    elseif n==10
    figure(4);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
    elseif n==20
    figure(5);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
    elseif n==30
    figure(6);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
elseif n==40
    figure(7);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);
    elseif n==50
    figure(8);
    mapa=[0 0 0 ; 1 1 1];
    colormap(mapa);
    image(y);

end
end

for nclas=1:2
    for i=1:ren
        for j=1:col
            if y(i,j)==nclas
                copia(i,j)=m(nclas);
            else
                e=0;
            end
        end
    end
end
end
end

```

Anexo 13:

%%%% código del método propuesto aplicado a la imagen 0509 %%%%

```
img='mw0509.bmp';
nclas=2;
beta=0.35;
iter=51;
[imagen,map]=imread(img);
imagen=im2double(imagen); %
[ren,col]=size(imagen);
copia=imagen;
for i=3:5:ren
    for j=3:5:col
        if (i>0&&i<=700&&j>0&&j<=650)
I=imcrop(imagen,[j i 4 4]);
Media=mean2(I);
varianza = mean2((I-Media).^2);
d=sqrt(varianza);
copia(i,j)=d;
        end
    end
end
area1=imcrop(copia,[430 220 60 60]);
area2=imcrop(copia,[560 592 90 62]);
m(1:nclas)=0;
[m(1),v(1)]=fmv(area1);
[m(2),v(2)]=fmv(area2);
close;
v=sqrt(v);
pxw1=ones(ren,col);
pxw2=ones(ren,col);
pxw(:,1)=pxw1;
pxw(:,2)=pxw2;
for i=1:ren
    for j=1:col
        k=copia(i,j);
        [a_3 indice] = max(normpdf(k,m,v));
        pxw(i,j,indice)=0;
    end
end
y=unidrnd(nclas,ren,col);
t=2.5;
for n=1:iter
    n
    for i=1:ren
        for j=1:col
            actual=y(i,j);
            a=i-1;b=j-1;c=i+1;d=j+1;
            eleccion=ceil(random('unif',0,nclas));
            act=0;elec=0;
```



```

for w=a:c
  for z=b:d
    if (w>0&&w<ren&&z>0&&z<col)
      if actual==y(w,z)
        act=act+1;
      end
      if eleccion==y(w,z)
        elec=elec+1;
      end
    end
  end
end
E_actual=-beta*act;
E_eleccion=-beta*elec;
if (actual>=1&&actual<=nclas)
  E_actual=E_actual+pxw(i,j,actual);
end
if (eleccion>=1&&eleccion<=nclas)
  E_eleccion=E_eleccion+pxw(i,j,eleccion);
end
if E_eleccion<=E_actual
  y(i,j)=eleccion;
else
  p=exp(-((E_eleccion-E_actual)/t));
  theta=rand(1);
  if theta<p
    y(i,j)=eleccion;
  else
    y(i,j)=actual;
  end
end
end
end
t=t*0.95;
if n==1
  figure(3);
  mapa=[0 0 0 ; 1 1 1];
  colormap(mapa);
  image(y);
elseif n==10
  figure(4);
  mapa=[0 0 0 ; 1 1 1];
  colormap(mapa);
  image(y);
elseif n==20
  figure(5);
  mapa=[0 0 0 ; 1 1 1];
  colormap(mapa);
  image(y);

```

```

        elseif n==30
            figure(6);
            mapa=[0 0 0 ; 1 1 1];
            colormap(mapa);
            image(y);
        elseif n==40
            figure(7);
            mapa=[0 0 0 ; 1 1 1];
            colormap(mapa);
            image(y);
        elseif n==50
            figure(8);
            mapa=[0 0 0 ; 1 1 1];
            colormap(mapa);
            image(y);
        end
    end
end

for nclas=1:2
    for i=1:ren
        for j=1:col
            if y(i,j)==nclas
                copia(i,j)=m(nclas);
            else
                e=0;
            end
        end
    end
end
end
end

```

Anexo 14:

%%% sobreposición con la imagen original 0509%%%

```

A = imread('BORDE1.bmp');
C = imread('mw0509.bmp');
B=imresize(A,[700 650]);
d= C+0.5*B;
figure(1);
imshow(d)

```

Anexo 15:

%%% contador de pixels 0509%%%

```

IM=imread('50.bmp');
B=imresize(IM,[700 650]);
negro=0;
blanco=0;
for i=1:700
    for j=1:650

```

```
k=B(i,j);
if (k==0)
    negro=negro+1;
else (k==255)
    blanco=blanco+1;
end
end
end
blanco
negro
```