



INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO

ESCOM

Trabajo Terminal

“AVIN”

Analizador de video para identificar naturaleza

2014-B021

Presentan

**Hernández García Rosaura
Morales Taxis Eliseo
Ortega Rojas Hugo**

Directores

M. en C. Jesús Alfredo Martínez Nuño, Dr. Jesús Yaljá Montiel Pérez



Diciembre 2015.



**INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO
SUBDIRECCIÓN ACADÉMICA**



No de TT:2014-B021

11 de Diciembre 2015

Documento Técnico

“AVIN”

Analizador de video para identificar naturaleza

Presentan:

Hernández García Rosaura¹

Morales Taxis Eliseo²

Ortega Rojas Hugo³

Directores

M. en C. Jesús Alfredo Martínez Nuño, Dr. Jesús Yaljá Montiel Pérez

RESUMEN

Se propone un sistema analizador de video para identificar elementos de naturaleza llamado AVIN. El sistema determinará los elementos de naturaleza que existen en distintas escenas del video, como resultado se generará un archivo de texto con los elementos de naturaleza identificados.

Para lograr el objetivo, a partir del video se obtienen muestras de imágenes (frames), se extraen vectores de características y se clasifican con herramientas de reconocimiento de imágenes. Los elementos de la naturaleza a identificar serán aquellos pertenecientes a paisajes mexicanos.

Palabras clave: Análisis de Imágenes, Análisis de video, Descriptores de naturaleza, Reconocimiento de Imágenes.

¹ auraros14@hotmail.com

² xeo.mortex@gmail.com

³ hugortega_ipn@hotmail.com



**ESCUELA SUPERIOR DE CÓMPUTO
SUBDIRECCIÓN ACADÉMICA**

**DEPARTAMENTO DE FORMACIÓN INTEGRAL
E INTITUCIONAL**

COMISIÓN ACADÉMICA DE TRABAJO DE TERMINAL



México, D.F. a 17 de Diciembre de 2015.

**DR. FLAVIO ARTURO SÁNCHEZ GARFIAS
PRESIDENTE DE LA COMISIÓN ACADÉMICA
DE TRABAJO TERMINAL
P R E S E N T E**

Por medio del presente, se informa que los alumnos que integran el **TRABAJOTERMINAL: 2014-B021**, titulado “AVIN Analizador de video para identificar naturaleza” concluyeron satisfactoriamente su trabajo. Los discos (DVDs) fueron revisados ampliamente por sus servidores y corregidos, cubriendo el alcance y el objetivo planteados en el protocolo original y de acuerdo a los requisitos establecidos por la Comisión que Usted preside.

ATENTAMENTE

M. en C. Jesús Alfredo Martínez Nuño

Dr. Jesús Yaljá Montiel Pérez

Advertencia

“Este documento contiene información desarrollada por la Escuela Superior de Cómputo del Instituto Politécnico Nacional, a partir de datos y documentos con derecho de propiedad y por lo tanto, su uso quedará restringido a las aplicaciones que explícitamente se convengan.”

La aplicación no convenida exime a la escuela su responsabilidad técnica y da lugar a las consecuencias legales que para tal efecto se determinen.

Información adicional sobre este reporte técnico podrá obtenerse en:

La Subdirección Académica de la Escuela Superior de Cómputo del Instituto Politécnico Nacional, situada en Av. Juan de Dios Bátiz s/n Teléfono: 57296000, extensión 52000.

Agradecimientos

Primero agradecer a mis padres **Ignacio Morales Salazar** y **Florencia Taxis Gutiérrez** por haber depositado su confianza y dedicación a lo largo de mi formación como Ingeniero en Sistemas Computacionales, que a pesar de las situaciones difíciles siempre me dieron todo su apoyo.

A mis hermanos **Oswaldo, Ruben, Luis Angel** por sus continuos consejos, confianza y paciencia a lo largo del desarrollo de este trabajo terminal.

Al Profesor **Jesús Alfredo Martínez Nuño** y al **Dr. Jesús Yaljá Montiel Pérez** directores de este trabajo terminal, por su orientación, seguimiento y supervisión continúa de la misma.

Al Dr. Alejandro por depositar su confianza y permitir poder trabajar en el módulo de naturaleza del proyecto MEX-CULTURE por su atención y seguimiento en los avances del proyecto.

A los sinodales que evaluaron el proyecto, por su seguimiento, atención, consejos y conocimiento para mejorar el proyecto, así como permitir que este proyecto sea aprobado.

A mis compañero del Trabajo Terminal Hugo por su confianza y dedicación para poder concluir este trabajo terminal, por su compromiso y apoyo.

A mi amiga y compañera de Trabajo Terminal Rosaura por todos sus sabios consejos, apoyo, confianza y dedicación con el proyecto.

A mis amigos Mauro, Juan Carlos, Jonathan, Luz, Jessica, Karen quienes me acompañaron en mis presentaciones para mostrarme su apoyo, a todos con los que alguna vez trabajamos juntos en alguna materia, gracias por sus consejos y amistad.

A una persona especial Brenda Aide Montaña Lara a quien en el último año me ha acompañado en todo lo que hago, por su apoyo incondicional, su paciencia y sus consejos.

A la Escuela Superior de Cómputo (ESCOM) por permitir realizar mis estudios en esta institución en la cual viví experiencias inolvidables

A todos los docentes de la ESCOM que me aportaron sus conocimientos y experiencias a lo largo de la carrera.

A todas las personas que estuvieron y trabajaron conmigo a lo largo de este trabajo terminal.

¡MUCHAS GRACIAS POR SU APOYO!

MORALES TEXIS ELISEO

Agradecimientos

A mi padre, quien me ha enseñado con su ejemplo, que hay que trabajar para lograr cualquier objetivo, que la ayuda no se cobra ni se presume, que lo material va y viene, el conocimiento y la familia es lo que hay que aprovechar a diario y que los días tienen 24 horas, cada uno sabemos cómo las usamos.

A mi madre que ha sido mi compañera de vida, en quien he podido confiar incondicionalmente y quien ha sabido brindarme sus palabras o sus silencios cuando la situación lo amerita.

A mi hermanos, gran motivo de mi esfuerzo, quienes todos los días con risas y peleas me han enseñado a ser paciente y que nuestras diferencias nos convierten en un gran equipo.

A mis directores de trabajo terminal y al grupo de trabajo de MEX-CULTURE, que me han brindado su apoyo desde el primer día que entre en este proyecto, les agradezco la oportunidad de trabajar con ellos.

De manera especial el profesor nuño a quien considero más que un profesor, un amigo, y de quien he aprendido todo lo que sé de análisis de imágenes.

Al Dr. Alejandro, que aunque está lejos nos ayudaba en todo cuanto podía.

A mi equipo de TT que trabajó mucho para que esto saliera bien, vivimos momentos de gran tensión, pero la confianza hizo que esto saliera adelante. De forma particular a Eliseo quien no solo es mi equipo, también es un gran amigo con quien he disfrutado mucho este tiempo -Te dije que hacíamos buen equipo.

Al profesor Salinas Lugo que ha hecho que cambie mi perspectiva de cualquier tipo de documentación y por quien ahora busco referencias en todo, porque tiene razón, es justo dar el crédito a quien lo merece.

Al profesor Edgardo A. Franco que me ha acompañado como tutor durante estos años e impulso mi interés por el análisis de imágenes desde la primera clase que tome con él.

A mi amigo Víctor que cada vez que hablo con él descubro algo nuevo, y quien sin darse cuenta me ha dado muchos motivos para seguir adelante aun en los momentos en que las cosas no van bien.

A Dae que con sus clases me mantuvo alerta estos últimos meses, no habría salido todo bien sin él.

A Mauro que nos ayudo en la presentación y que siempre me hace sonreír.

A mi amigo y compañero José Antonio, de quien aprendí muchas cosas de la carrera y otras más de la vida, me aconsejó y apoyó a su muy especial manera, durante bastante tiempo.

A mis amigos, consejeros, delfines, compañeros de clases, maestros, a toda mi familia, a quien aun estando lejos alegra los días más nublados.

Al IPN y la ESCOM que me han dado momentos inolvidables, desde que fui aceptada como parte de esta maravillosa comunidad, de la cual estoy sumamente orgullosa.

A todas las personas que he conocido durante la carrera, y que han formado parte de mi día a día, en mis clases, talleres, y actividades que he realizado durante mi estancia en ESCOM, porque cada una me ha enseñado algo.

A Dios por darnos la vida y por poner las oportunidades y personas indicadas en los momentos indicados.

Con todo mi cariño, ¡Gracias!

HERNÁNDEZ GARCÍA ROSAURA

Agradecimientos

Expreso mis agradecimientos a mis directores de este trabajo terminal al M. en C. Jesús Alfredo Martínez Nuño al Dr. Jesús Yaljá Montiel Pérez y al Dr. Alejandro Ramírez por la dedicación y apoyo que brindaron a este trabajo terminal.

Asimismo, agradezco a mis compañeros de trabajo Eliseo Morales Taxis y Rosaura Hernández García por su apoyo, confianza y amistad que me brindaron.

Agradezco especialmente a mi familia, a mis padres, mi hermana y hermanos fuente de apoyo constante e incondicional en toda mi vida y más aún en mis años duros de carrera profesional.

También agradezco a mis amigos que siempre me han brindado un gran apoyo necesario en los momentos difíciles de este trabajo y esta profesión.

A todos, Muchas Gracias.

ORTEGA ROJAS HUGO

Índice

1	Introducción.....	13
1.1	Objetivo general.....	13
1.2	Objetivo particular.....	13
1.3	Justificación.....	13
2	Antecedentes.....	14
2.1	Estado del arte.....	14
2.2	Marco teórico.....	17
2.2.1	Imágenes digitales.....	17
2.2.2	Modelos de color.....	17
2.2.3	Descriptores de imágenes.....	19
2.2.4	Descriptores de textura.....	21
2.2.5	Uso de GLCM y GLDM.....	23
2.2.6	Redes neuronales artificiales (RNA).....	24
2.2.7	Algoritmo K-Means.....	26
2.2.8	Ecosistemas de México.....	27
3	Análisis.....	27
3.1	Metodología.....	27
3.2	Estudio de factibilidad.....	28
3.2.1	Factibilidad operativa.....	28
3.2.2	Factibilidad técnica.....	29
3.3	Métricas y estimación.....	30
3.3.1	Estimación COCOMO Básico Orgánico.....	34
3.4	Descripción de herramientas.....	37
3.4.1	Software.....	37
3.5	Análisis de riesgos.....	38
4	Diseño.....	42
4.1	Diagramas de casos de uso AVIN.....	42
4.1.1	Diagrama de caso de uso. Sistema AVIN.....	42
4.1.2	Caso de uso. Selecciona video.....	43
4.1.3	Caso de uso. Inicia Analizador.....	44
4.2	Diagrama de procesos.....	45
4.3	Diagramas de secuencia.....	47

4.3.1	Diagrama de secuencia Selecciona video (DSSV).....	47
4.3.2	Diagrama de Secuenciade Archivo (DSA).....	48
4.4	Diagrama de Clases AVIN (DCA).....	49
4.5	Interfaz gráfica del sistema AVIN	50
5	Desarrollo	52
5.1	Segmentación de la imagen.	52
5.2	Procedimiento de las etapas del sistema AVIN.....	53
5.3	Proceso del sistema AVIN.....	53
5.4	Red Feedforward (FF).....	54
5.4.1	Clasificación con la red FeedForward.....	56
5.5	Red de Base Radial	59
5.5.1	Entrenamiento	59
5.5.2	Proceso de simulación.....	61
5.5.3	Entrenamiento de la red de base radial con algoritmo K-Means	62
6	Pruebas	63
6.1	Módulo de extracción de características	63
6.2	Pruebas de descriptores.....	65
6.2.1	Pruebas de los descriptores en el plano RGB.....	65
6.2.2	Pruebas con el modelo HSV	68
6.3	Pruebas de los descriptores de textura en niveles de gris.....	70
6.3.1	Pruebas GLCM	70
6.3.2	Pruebas GLDM.....	73
6.4	Pruebas con red de base radial y Feedforward.....	77
6.5	Matriz de confusión.....	79
6.6	Estructura del archivo del resultado final	80
7	Conclusiones.....	80
8	Trabajo a futuro	81
	Referencias.....	82

Índice de Diagramas

Diagrama 1. Arquitectura de la red de base radial [30].....	26
Diagrama 2. Caso de Uso del Sistema (CUS).....	42
Diagrama 3 Procesos.....	46

Diagrama 4. Diagrama de Secuencia Selecciona Video (DSSV).....	47
Diagrama 5. Diagrama de Secuencia de Archivo (DSA).....	48
Diagrama 6. Diagrama de Clases AVIN (DCA).....	49
Diagrama 7. Entrenamiento de la red neuronal.....	57
Diagrama 8. Uso de red neuronal.....	58
Diagrama 9. Proceso de entrenamiento de la red de base radial.....	60
Diagrama 10. Representación Gaussiana.....	62
Diagrama 11. Comparación de distintos algoritmos.....	62

Índice de Ilustraciones

Ilustración 1. Esquema del cubo de color RGB, puntos a lo largo de la diagonal principal tienen valores de gris, de negro en el origen a blanco en el punto (1, 1, 1). En idioma inglés.[17].....	17
Ilustración 2 Método GLCM, Posición de las ventanas y Matriz resultante.....	22
Ilustración 3. Modelo de una red neuronal artificial Feedforward.....	25
Ilustración 4. Ciclo de vida en espiral en idioma inglés [36].....	28
Ilustración 5. Interfaz principal.....	50
Ilustración 6. Búsqueda de archivo.....	50
Ilustración 7. Descripción del archivo.....	51
Ilustración 8. Mensaje de advertencia.....	51
Ilustración 9. Segmentación de la imagen.....	52
Ilustración 10. Proceso sistema AVIN.....	53
Ilustración 11. Arquitectura de la red FeedForwarded.....	54
Ilustración 12. Clasificación de las clases en FeedForward y Base Radial.....	56
Ilustración 13. Representación del cálculo del centro y radio de cada clase.....	59
Ilustración 14. Asimetría.....	65
Ilustración 15. Media.....	66
Ilustración 16 Desviación estándar.....	66
Ilustración 17 Entropía.....	67
Ilustración 18 Media en el modelo HSV.....	68
Ilustración 19 Desviación estándar en el modelo HSV.....	69
Ilustración 20 GLCM, Homogeneidad, Contraste y Correlación.....	70
Ilustración 21 GLCM, Homogeneidad, Contraste y energía.....	71
Ilustración 22 GLCM, Energía, Correlación y Contraste.....	72
Ilustración 23 GLDM, Asimetría, Media, Desviación.....	73
Ilustración 24 GLDM, curtosis, Desviación, Media.....	74
Ilustración 25 GLDM, Asimetría, Entropía y Desviación.....	74
Ilustración 26 GLDM Kurtosis, Desviación y Entropía.....	75
Ilustración 27 GLDM, Media y Desviación.....	76
Ilustración 28.....	77
Ilustración 29.....	77
Ilustración 30.....	77
Ilustración 31.....	77

Ilustración 32	77
Ilustración 33	77
Ilustración 34 Ilustración 35	78
Ilustración 36 Ilustración 37	78
Ilustración 38 Ilustración 39	78
Ilustración 40	78
Ilustración 41	78
Ilustración 42 Ilustración 43	79

Índice de Tablas

Tabla 1. Recursos Hardware	29
Tabla 2. Recursos de software.....	30
Tabla 3. Recursos adicionales	30
Tabla 14 Coeficientes.....	34
Tabla 1-5 Costos de Desarrollo.....	36
Tabla 16 Estimaciones	36
Tabla 7. Tabla comparativa Java y C	38
Tabla 8. Riesgos de proyecto para AVIN	39
Tabla 9. Probabilidad de riesgos de producto	39
Tabla 10. Estrategias para los riesgos de producto	40
Tabla 11. Riesgos de proyecto para AVIN	41
Tabla 12. Probabilidad de riesgos de proyecto y negocio	41
Tabla 13. Estrategias para los riesgos de proyecto y negocio	42
Tabla 14. Caso de uso selecciona video	43
Tabla 15. Caso de uso inicia analizador.....	44
Tabla 16. Matriz de confusión para la red feedforward	79
Tabla 17. Matriz de confusión para la red de base radial	79

1 Introducción

Las actividades del ser humano generan una gran cantidad de información que se llega a almacenar en diferentes formatos, ya sean en medio electrónico o no. Para el caso de imágenes, sonido y video existen actualmente formatos digitales. Cada formato tiene sus características y en algunos casos se almacenan para formar bases de datos con referencias generales y se hacen de manera manual. En lo referente a las búsquedas, se pueden realizar de manera automatizada y rápida.

El proyecto de investigación MEX-CULTURE con número CONACYT: 158781 y registro SIP:SIP-2012-RE/64 es un proyecto que tiene como objetivo el indexado de bibliotecas multimedia para la preservación y disseminación de la cultura mexicana.

El proyecto MEX-CULTURE se enfoca en elaborar nuevos métodos automatizados para el procesamiento a gran escala de contenido visual codificado (video, audio, imágenes y discurso).

1.1 Objetivo general

Implementar un descriptor para el módulo de detección de la naturaleza en videos que forma parte del proyecto MEX-CULTURE, el cual tiene como objetivo la indexación de contenido multimedia para la disseminación del patrimonio de la cultura mexicana.

1.2 Objetivo particular

Identificar elementos de naturaleza en un video, creando un algoritmo analizador de fotogramas (frames) dando como resultado un archivo de texto donde se especificará la existencia de elementos de la naturaleza los cuales serán agua, cielo y vegetación.

1.3 Justificación

El prototipo propuesto AVIN será un módulo en el proyecto “MEX-CULTURE” y servirá como apoyo a la preservación y disseminación de la cultura mexicana; México es uno de los países más ricos en cuanto a biodiversidad por lo cual detectar naturaleza en los videos ayuda a cumplir el objetivo del proyecto general. En el proyecto “MEX-CULTURE” se tiene en cuenta un módulo de obtención de descriptores, video a color, el cual abordará los siguientes algoritmos: algoritmo de obtención de descriptor de la naturaleza, de personas y de ciudad. Este trabajo terminal pretende colaborar con el algoritmo de obtención de descriptor de la naturaleza.

2 Antecedentes

2.1 Estado del arte

Los trabajos de Y. Liu y X. Zhou , “*Automatic texture segmentation for texture based image-retrieval*” [1] y Kalaiselvi con “*HMM Based Automatic Video Classification Using Static and Dynamic Features*” [2] están enfocados a diferentes ambientes de , sin embargo el trabajo de P. Santana et al “*Water detection with segmentation guided dynamic texture recognition*” [3] tiene relación con la detección de la naturaleza, pero específicamente del agua usando sensores ópticos para su detección. A diferencia de estos artículos nuestro trabajo terminal (AVIN) solo se enfocará en la naturaleza, por ello implementaremos un algoritmo automatizado para realizar esta tarea.

Los siguientes proyectos tienen cierta similitud con este trabajo terminal, usan diferentes métodos para atacar el problema de reconocimiento de imágenes:

- C. Schmid en “Weakly supervised learning of visual models and its application to content-based image retrieval” [4]. Este trabajo está enfocado a la recuperación de imágenes. Hace cálculos de descriptores genéricos invariantes a rotaciones y aplicados a cada elemento de la imagen que haremos referencia como pixel. Utiliza imágenes en escala de grises. Ordena sus descriptores haciendo agrupaciones mediante el algoritmo de K-means. Usa la distancia euclidiana para hacer la comparación entre los descriptores. Hace uso de una función de distribución gaussiana para calcular la probabilidad de un descriptor.
- Julia Vogel en su tesis “SemanticSceneModeling and Retrieval PhD Thesis”[5] y en su artículo “Semantic modeling of natural scenes for content-based image retrieval”[6] establece la forma de recuperación de imágenes con base en anotaciones o etiquetas. Utiliza una máquina de soporte vectorial para entrenar un clasificador de características y un clasificador tipo K-NN. A cada región de la imagen se le extraen histogramas de HSI.
- Para el artículo de Solmaz, B et al en “*Classifying Web Videos using a Global Video Descriptor*”[7] se usó un descriptor global para clasificar videos web, este descriptor sirve para identificar las acciones de personas a lo largo del video como puede ser caminar, correr, andar en bicicleta, etc., se realizaron varias pruebas para hacer el descriptor más preciso y para ello se tomaron los conjuntos de datos KTH [8], UCF50 [9] y HMDB51 [10].

Para poder encontrar el descriptor web se analizaron los espectros de frecuencia en un conjunto de fotogramas integrados en videos cortos conocido como clip (cortometraje) de video, el cual puede ser estimado calculando la transformada discreta de Fourier. En este artículo se consideró las diferentes direcciones y frecuencias del movimiento y para esto se usaron los filtros de Gabor 3-D. A partir de las ecuaciones de la transformada discreta de Fourier se puede calcular el espectro de frecuencia, para así generar un banco de filtros, al final se genera un vector que contiene características de longitud que representa un solo clip de video. El último paso es aplicar PCA (Principal Components Analysis), un

método popular para la reducción de dimensionalidad, con el fin de generar el descriptor global. Los resultados de este descriptor comparado con los otros descriptores son sobresalientes.

- En el artículo de Sasidharan, R et al en *Dynamic Texture Segmentation of Video Using Texture Descriptors and Optical Flow of pixels for Automating Monitoring*[11] la segmentación de textura dinámica suele ser más difícil ya que el desplazamiento de partículas se cambia en determinados tiempos.

La idea general de cualquier segmentación dinámica utiliza la textura como clave, consiste en identificación de áreas de textura similar utilizando el histograma y manteniéndolos como un conjunto único. La misma técnica se siguió en este trabajo con un enfoque ligeramente diferente.

El método desarrollado por Brian G. Schunck[12] es un método que maneja la textura dinámica porque aún mejor los histogramas los usan como uno solo y de ahí obtiene el valor acumulativo.

El algoritmo LBP y WLD [13] a pesar de que ambos son muy buenos, es recomendable utilizarlos en conjunto porque los dos manejan diferentes formas los histogramas y los reconocimientos de imágenes. El modo más fácil de detectar los movimientos en una imagen es aplicando el algoritmo de flujo óptico, obteniendo un vector de cualidades.

Las características principales de segmentación son:

Descripción de textura: Estas texturas descriptoras se utilizan como descriptor espacial-textura cuando se utiliza en plano XY de un vídeo. Estos se llaman temporal-textura descriptores cuando se utilizan en XT y el plano YT.

Flujo óptico: Flujo óptico como en [14] es la distribución de aparente velocidades de movimiento de los patrones de brillo en una imagen. Flujo óptico puede surgir de movimiento relativo de los objetos y también la del espectador. Esto implica que, flujo óptico puede dar información importante con respecto a la disposición espacial de los objetos vistos y también la tasa de cambio de este arreglo.

- La tesis de Serrano J., en *“Recuperación de imágenes mediante rasgos descriptores globales y locales”*[15] propone una metodología para recuperar imágenes con escenas naturales, para esto utiliza un conjunto de características que describen las escenas y que son obtenidos a partir de algunos puntos aleatorios de cada imagen, para después ser utilizados en algún clasificador, para así obtener una base de datos indexada. Las características que usan para describir las escenas son la media, la desviación estándar y la homegeneidad, para los clasificadores se realizaron pruebas con el clasificador 1-NN, el clasificador Bayesiano y una red neuronal artificial cuyo tipo de aprendizaje es del tipo de retropropagación.

Los artículos descritos anteriormente sirven de referencia al sistema AVIN debido a que están relacionados con el análisis de imágenes.

2.2 Marco teórico

2.2.1 Imágenes digitales

De acuerdo al Instituto de Tecnologías Educativas [16] define a las imágenes digitales como:

Las imágenes digitales se obtienen a través de dispositivos de conversión analógico-digital como escáner, cámara fotográfica digital o directamente desde la computadora utilizando cualquier programa de tratamiento de imágenes. La información digital que genera cualquiera de los medios citados es almacenada en la computadora mediante bits. Las computadoras trabajan con información digital, con información numérica. En una computadora la información analógica de textos, imágenes y sonidos se codifica por medio de bits.

A partir de esto podemos entender que una imagen digital parte a través de una entrada física ya sea foto o video capturado, por lo que el procesamiento para la imagen es por medio de bits, para así poder manipularla o en nuestro caso detectar elementos.

2.2.2 Modelos de color

El propósito de los modelos de color también llamados espacios de color o sistemas de color, es el de facilitar la especificación de los colores utilizando algún estándar.

En esencia, un modelo de color es una especificación de un sistema coordinado y de un subespacio, dentro de ese sistema, donde cada color puede ser representado por un sólo punto.

2.2.2.1 Modelo RGB (Red, Green and Blue)

En el modelo RGB, cada color aparece en sus componentes espectrales primarios de rojo, verde y azul. Este modelo está basado en un sistema coordinado cartesiano. En la ilustración 1 se muestra el subespacio de interés para el modelo RGB en un cubo:

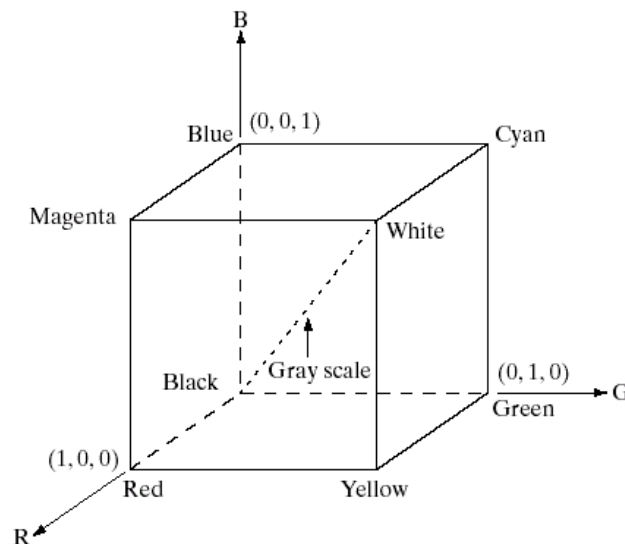


Ilustración 1. Esquema del cubo de color RGB, puntos a lo largo de la diagonal principal tienen valores de gris, de negro en el origen a blanco en el punto (1, 1, 1). En idioma inglés.[17]

2.2.2.2 Modelo HSI (Hue, Saturation and Intensity)

De acuerdo a Gonzalez y Woods en su libro “Digital image processing” [17] El tono es un atributo cromático que representa un color puro como amarillo, naranja, y rojo. Mientras que la saturación aporta una medida del grado en que el color está diluido en luz blanca. El modelo HSI se representa por las siguientes características:

1. La componente de intensidad está desacoplada de la información cromática.
2. Las componentes de tono y saturación están íntimamente relacionadas con la forma en que los humanos percibimos el color.

El modelo HSI es una herramienta ideal para el procesamiento de imágenes en color.[17]

2.2.2.3 El modelo HSV

Como menciona, Pajares y de la Cruz, en su libro de Visión por computador, Imágenes digitales y aplicaciones [18] el modelo HSV es similar al modelo HSI que hemos visto en la sección anterior, con la diferencia de que el valor de intensidad varia de negro a blanco en un único prisma, cabe mencionar que el modelo HSI representa estos valores de intensidad en dos prismas. A continuación en la Ilustración x se muestra el diagrama del modelo HSV.

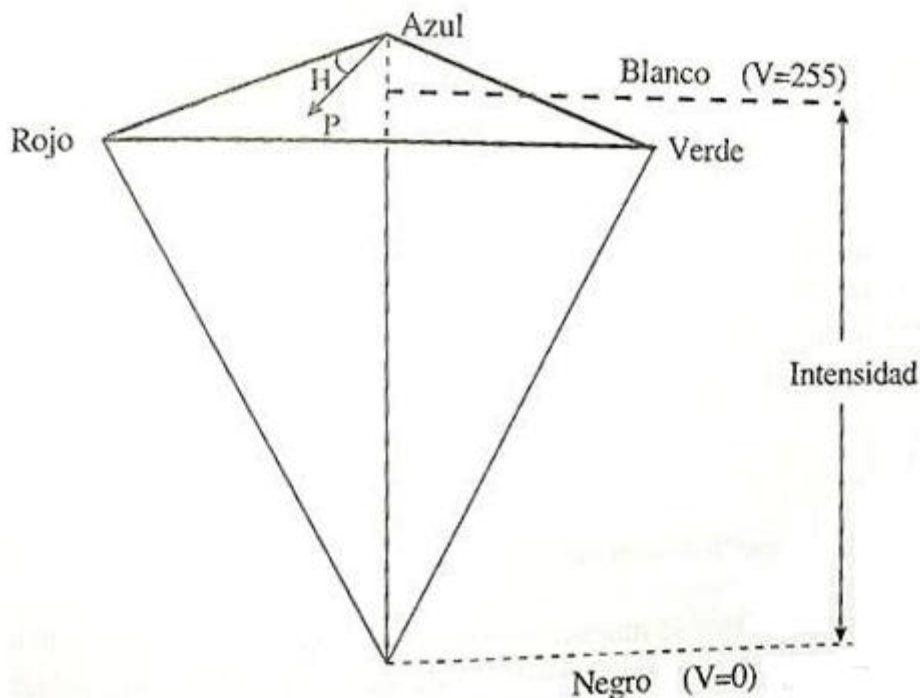


Imagen 1 Diagrama del modelo HSV [18]

2.2.3 Descriptores de imágenes

Nora La Serna Palomino y Luis Alvarado Ramírez [19] mencionan que:

Un descriptor de una imagen es una forma de representarla por sus características, con fines de almacenamiento y recuperación, en donde el descriptor es una representación matemática. La representación del descriptor se realiza tanto para características locales como globales.

A partir de la definición que hace La Serna y Alvarado nos damos cuenta de lo que consta un descriptor de imagen, cabe mencionar que para hacer un descriptor más robusto debemos elegir las características que mejor clasifiquen las clases.

Para crear el descriptor se necesita entrenar a la red neuronal la cual será nuestro clasificador, para ello se obtienen parches de las imágenes patrón y se analizan para obtener características que nos ayudaran al entrenamiento de la red.

Las características que se obtendrán de los segmentos previamente obtenidos se enlistan a continuación:

- Media
- Desviación estándar
- Entropía
- Asimetría

Para una mejor comprensión de su aplicación se describirán en lo subsecuente.

2.2.3.1 Media

En matemáticas y estadística una media o promedio es una medida de tendencia central que resulta al sumar todos los datos y dividir el resultado entre el número total de estos mismos.

La media se expresa mediante la siguiente ecuación:

$$x = \frac{1}{n} \sum_{i=1}^n x_i \quad [20]$$

Donde x es la media aritmética, x_i es el valor de canal del elemento i y n es el total de elementos.

Para el proyecto AVIN la media se aplica en el módulo de extracción de características. Después de obtener los parches de los frames del video se calcula en los vectores la media para obtener un valor que represente al grupo de pixeles de cada parche que se extrajo, con el fin de modular una media en el conjunto de parches.

2.2.3.2 Desviación estándar

La desviación típica o desviación estándar (se denotada con el símbolo s) es una medida de dispersión para variables de razón (variables cuantitativas o cantidades racionales) y de intervalo. Se define como la raíz cuadrada de la varianza de la variable.

Para conocer con detalle un conjunto de datos, no basta con conocer las medidas de tendencia central, sino que necesitamos conocer también la desviación que presentan los datos en su distribución respecto de la media aritmética de dicha distribución, con objeto de tener una visión de los mismos más acorde con la realidad al momento de describirlos e interpretarlos para la toma de decisiones.

La desviación estándar se expresa mediante la siguiente ecuación:

$$s = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2 \quad [20]$$

Donde s es la desviación estándar, n es el total de datos, x_i es valor del canal del elemento i , y \bar{x} es la media aritmética.

La desviación estándar es parte del vector de características (del módulo de extracción de características) debido a que representa el alejamiento de los píxeles en su distribución respecto a la media aritmética y es útil porque muestra la tendencia de desvío, dándonos una mejor visión de la realidad de los datos.

2.2.3.3 Entropía

La entropía se puede considerar como la cantidad de información promedio que contienen los símbolos usados. Los símbolos con menor probabilidad son los que aportan mayor información. La entropía está dada por la ecuación

$$H = - \sum_i p(x_i) \log_2 p(x_i) \quad [21]$$

Donde H es la entropía, x_i es p es la probabilidad de que se incluya el valor de nivel del canal x_i .

Esta es incluida en el vector de características para diferenciar una clase de otra de acuerdo a como se comporta la distribución, si es plana la entropía será alta y si no es plana la entropía será baja.

2.2.3.4 Asimetría

Las medidas de asimetría son indicadores que permiten establecer el grado de simetría (o asimetría) que presenta una distribución de probabilidad de una variable aleatoria sin tener que hacer su representación gráfica.

La asimetría de Fisher se expresa mediante la siguiente ecuación

$$A = \frac{E(x - \bar{x})^3}{s^3} \quad [22]$$

Donde A es la asimetría de Fisher, x es el valor de nivel de canal, \bar{x} es la media aritmética, y s es la desviación estándar.

La asimetría nos permite observar a través de un valor numérico si la distribución de píxeles en cada parche es simétrica o asimétrica sin tener que graficarlo, de esta manera podemos diferenciar las características entre las clases que tendremos, por esta razón la asimetría forma parte también del vector de características.

Una vez obtenido las características anteriormente definidas se procede a guardarlas temporalmente en un vector de características y así para poder hacer uso de ellas en el proceso de entrenamiento de la red neuronal.

2.2.4 Descriptores de textura

Utilizar descriptores enfocados en el comportamiento de color, como son la media y desviación en los planos de RGB limita el análisis en cualquier caso en el que se presente una imagen en escala de grises, por lo cual se decidió incluir descriptores de textura en este plano, para ello se explicara una breve introducción sobre que es textura en una imagen digital.

Se utiliza el término textura, para caracterizar la rugosidad de la superficie de un objeto o fenómeno. La textura es uno de los descriptores más usados en el procesamiento de imágenes, ya que está relacionada con la composición de una sustancia a través de las variaciones en la superficie de los materiales.

La textura de una imagen digital es una cuantificación del cambio espacial entre los valores de tono de una vecindad [23] otra forma de describir el comportamiento de la textura es diciendo que: *“El nivel de gris de un pixel en una imagen monocromática representa un tono, mientras que un nivel de textura es la variabilidad del tono en una vecindad”* [24]

A continuación estudiaremos los métodos GLCM, GLDM y LBP como descriptores posibles para el sistema AVIN.

2.2.4.1 GLCM

GLCM, Matriz de co-ocurrencia en niveles de gris o por sus siglas en inglés, Gray level Co-occurrence Matrix

La matriz de co-ocurrencia es una técnica basada en las estadísticas de dependencia espacial en niveles de gris, de Segundo orden. Esto implica el estudio de la interdependencia espacial de los píxeles y su distribución en un área local. Las estadísticas de segundo orden describen la forma en que los niveles de gris tienden a ocurrir cuando se encuentran juntos, en pares, además proporciona una descripción del tipo de textura que se encuentra presente. [25]

“La matriz de co-ocurrencia describe la frecuencia de un nivel de gris que aparece en una relación espacial específica con otro valor de gris, dentro del área de una ventana determinada”[24]

Para obtener una matriz de co-ocurrencia, se deben tomar en cuenta, el tamaño de la ventana, la distancia entre cada pixel y el ángulo en que se analizará.

El tamaño de la ventana debe ser cuadrado y de un número impar de píxeles, el resultado del cálculo de textura es un número que representa la ventana completa y que es colocado en la nueva matriz en el lugar del pixel central de esa ventana, después la ventana se mueve y se repite el cálculo, para todos los píxeles, de esta forma obtenemos una nueva matriz, tal como se muestra en la ilustración 2.

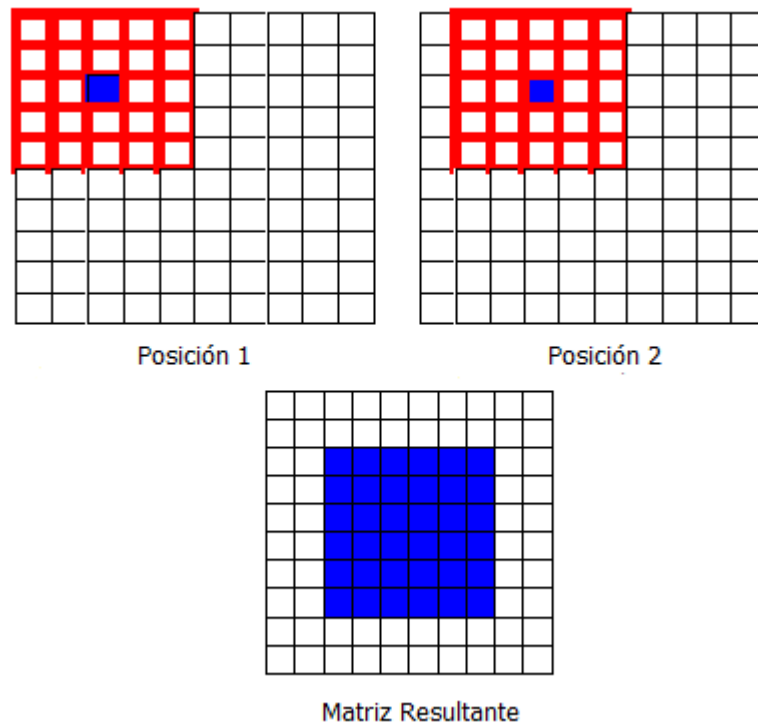


Ilustración 2 Método GLCM, Posición de las ventanas y Matriz resultante

2.2.4.2 GLDM

GLDM, método de diferencias en niveles de gris o por sus siglas en inglés Gray Level Difference Method, es un método similar a GLCM, este método también necesita de una ventana, una distancia y un ángulo para ser calculado y al igual que en GLCM se obtendrá una matriz con los valores calculados; Los elementos de esta matriz representan la diferencia absoluta entre las intensidades de un par de píxeles que se encuentran separados por una distancia d y un ángulo de inclinación.[26]

2.2.5 Uso de GLCM y GLDM

En las secciones anteriores se habló de los métodos de textura con GLCM y GLDM, de los cuales se obtiene como resultado matrices de textura, pero no se ha mencionado como son usadas estas para obtener valores específicos que sirvan como elemento en un vector característico, el cual será usado para el reconocimiento.

Una vez construidas las matrices, de ellas se podrán derivar diferentes medidas de textura, tales como la homogeneidad, media, desviación estándar, correlación, entropía y asimetría; Se dio una introducción a ellas en la sección de Descriptores de imágenes, pero en esta sección se dará una breve introducción a las que no se han mencionado, que son: homogeneidad y correlación.

2.2.5.1 Homogeneidad

La homogeneidad se calcula mediante la siguiente ecuación:

$$\sum_{i,j=0}^{N-1} P_{i,j} / 1 + (i - j)^2 \quad [24]$$

El valor de homogeneidad resultara alto cuando la matriz este concentrada a lo largo de la diagonal, esto ocurre cuando la imagen es localmente homogénea de acuerdo al tamaño de la ventana utilizada.

2.2.5.2 Correlación

La correlación se calcula de una manera bastante diferente a las otras por lo cual el resultado es muy distinto e independiente de las medidas anteriores. Algunas propiedades de la correlación son:

- Un objeto tiene más alta correlación dentro de él que entre objetos adyacentes.
- Los píxeles cercanos están más correlacionados entre sí que los píxeles más distantes [24].

La fórmula para calcular la correlación es la siguiente:

$$P_{i,j} = \frac{i - \mu_i}{\sigma_i^2} \frac{j - \mu_j}{\sigma_j^2} \quad [24]$$

2.2.6 Redes neuronales artificiales (RNA)

De acuerdo a la investigación que hizo R. Salas sobre las Redes Neuronales Artificiales[27] menciona que este tipo de redes son:

Un método de resolver problemas, para aquellas tareas de clasificación, identificación, diagnóstico, optimización o predicción en las que el balance datos/conocimiento se inclina hacia los datos y donde, adicionalmente, puede haber la necesidad de aprendizaje en tiempo de ejecución y de cierta tolerancia a fallos. En estos casos las RNAs adaptan dinámicamente reajustando constantemente los “pesos” de sus interconexiones.

Las RNA se basan en la analogía que existe en el comportamiento y función del cerebro humano, en particular del sistema nervioso, el cual está compuesto por redes de neuronas biológicas que poseen bajas capacidades de procesamiento, sin embargo toda su capacidad cognitiva se sustenta en la conectividad de éstas.

Se define la unidad de una red neuronal artificial en la misma investigación como:

La unidad de una red neuronal artificial es un procesador elemental llamado neurona que posee la capacidad limitada de calcular, en general, una suma ponderada de sus entradas y luego le aplica una función de activación para obtener una señal que será transmitida a la próxima neurona. Estas neuronas artificiales se agrupan en capas o niveles y poseen un alto grado de conectividad entre ellas, conectividad que es ponderada por los pesos. A través de un algoritmo de aprendizaje supervisado o no supervisado, las RNA ajustan su arquitectura y parámetros de manera de poder minimizar alguna función de error que indique el grado de ajuste a los datos y la capacidad de generalización de las RNA.

2.2.6.1 Red neuronal Feedforward

De acuerdo al artículo de Salas R.[27] las redes neuronales Feedforward (RNF) corresponden a la clase de RNA más estudiada por el ámbito científico y la más utilizada en los diversos campos de aplicación.

Las diferentes clases de RNA se distinguen entre sí por los siguientes elementos:

- Las neuronas o nodos que constituye los elementos básicos de procesamiento.
- La arquitectura de la red descrita por las conexiones ponderadas entre los nodos.
- El algoritmo de entrenamiento, usado para encontrar los parámetros de la red.

Morfológicamente, las RNF consisten en un conjunto de neuronas organizadas en capas como se muestra en la Ilustración 2[28]

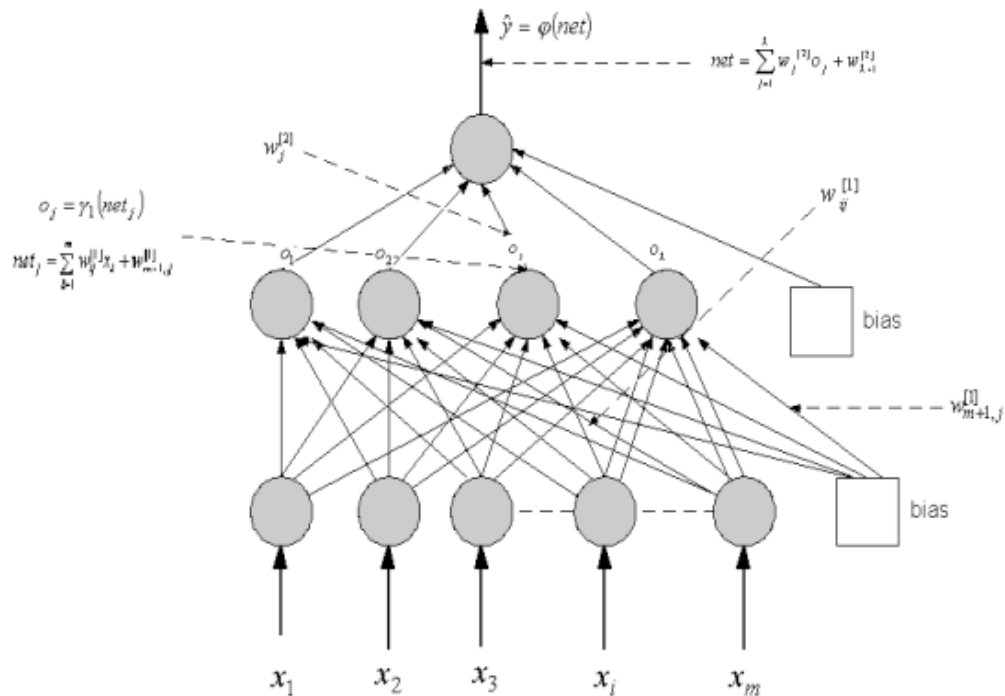


Ilustración 3. Modelo de una red neuronal artificial Feedforward.

2.2.6.2 Propagación hacia atrás (Backpropagation)

La propagación hacia atrás de errores (Backpropagation del inglés) es un algoritmo de aprendizaje supervisado, este método formalizado en 1986 por, Rumelhart, Hinton y Williams se usa para entrenar redes neuronales artificiales la asociación que existe entre los patrones de entrada y las clases correspondientes, utilizando varios niveles de neuronas.

El funcionamiento de este método consiste en el aprendizaje de un conjunto predefinido de pares de entradas-salidas dados como ejemplo: primero se aplica un patrón de entrada como estímulo para la primera capa de las neuronas de la red, se va propagando a través de todas las capas superiores hasta generar una salida, se compara el resultado en las neuronas de salida con la salida que se desea obtener y se calcula un valor de error para cada neurona de salida. A continuación, estos errores se transmiten hacia atrás, partiendo de la capa de salida hacia todas las neuronas de la capa intermedia que contribuyan directamente a la salida. Este proceso se repite, capa por capa, hasta que todas las neuronas de la red hayan recibido un error que describa su aportación relativa al error total. Basándose en el valor del error recibido, se reajustan los pesos de conexión de cada neurona, de manera que en la siguiente vez que se presente el mismo patrón, la salida esté más cercana a la deseada. [29]

2.2.6.3 Red neuronal de Base Radial

La red neuronal de base radial consta de tres capas, una de entrada, una oculta y una de salida, en la capa oculta se calcula la función de pertenencia de cada clase y se implementa un algoritmo de clusterización para su entrenamiento, la arquitectura general de la red de base radial se visualiza en la siguiente imagen.

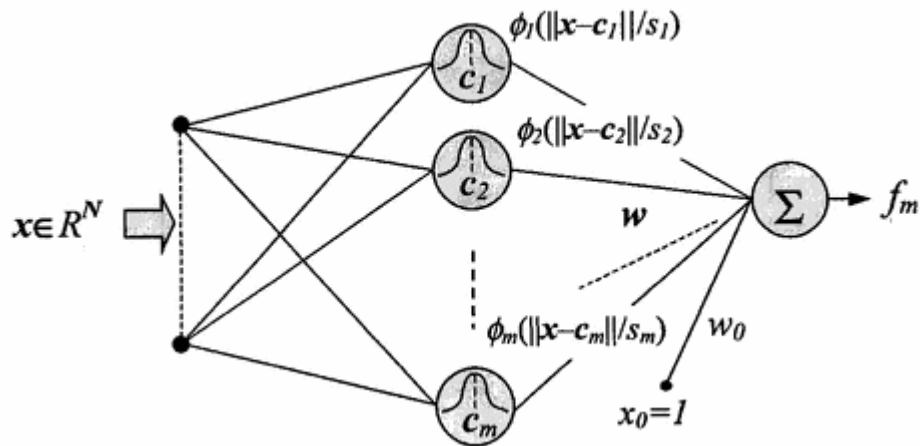


Diagrama 1. Arquitectura de la red de base radial [30].

2.2.7 Algoritmo K-Means

Este algoritmo fue presentado por James MacQueen en 1967 [31] en el cual describe el algoritmo de K-Means, este algoritmo es un método de agrupamiento en el que se definen el número de grupos en los que se desea separar, es uno de los algoritmos más simples para resolver el problema de agrupamiento.

De acuerdo a lo que presento MacQueen el algoritmo de K-Means se puede describir en los siguientes pasos:

- Definir n centros aleatorios dentro de los elementos que se desean agrupar.
- Calcular la distancia Euclídeana de cada uno de los elementos a los centros definidos en el paso anterior.
- Agrupar los elementos de acuerdo a que elemento está más cerca de cada centro.
- Calcular los nuevos centros, recorriendo todos los elementos de cada grupo y sacando la media ya explicada anteriormente.
- Repetir los pasos 2, 3 y 4 hasta que los centros ya no se muevan ó sea que sean los mismos, podemos definir el número de iteraciones para que la agrupación sea mejor en cuanto separación.

2.2.8 Ecosistemas de México

Considerando que el proyecto general de “MEX-CULTURE” se enfocara a la preservación y diseminación de la cultura mexicana es importante mencionar los ecosistemas existentes en México.

La Comisión Nacional para el Conocimiento y Uso de la Biodiversidad (CONABIO) habla sobre los ecosistemas de México “Debido a su ubicación geográfica y a su diverso relieve, México tiene una gran diversidad de ecosistemas, que van desde lo más alto de las montañas hasta los mares profundos, pasando por desiertos y arrecifes de coral, bosques nublados y lagunas costeras” [32]. Los ecosistemas considerados se enlistan a continuación:

- Templados
- Tropicales
- Subterráneos
- Terrestres modificados
- Acuáticos modificados
- Dulceacuícolas
- Costeros
- Marinos

3 Análisis

3.1 Metodología

El modelo en espiral que fue propuesto originalmente por Boehm [33] el cual es un modelo de proceso de software evolutivo que conjuga la naturaleza iterativa de la construcción de prototipos con los aspectos controlados del modelo en cascada [34]. Seleccionamos esta metodología por ser eficiente y flexible ya que el proyecto requiere de modificación constante y de análisis del algoritmo que se va usar, además de las pruebas que se deben de estar realizando a tiempo ya sea a lo largo del proyecto o bien si se desea hacer posteriormente una vez finalizado el proyecto. Esta metodología nos sirve para reducir riesgos y permite aplicar el enfoque de construcción de prototipos en cualquier etapa de evolución del producto. El diagrama de espiral se basa en cuatro etapas la cual se iterará las veces que sea necesaria y se enlistan a continuación [35].

I- Definición de objetivos

II- Evaluación y reducción de riesgos

III- Desarrollo y validación

IV- Planificación

En la Ilustración 4 se muestra el diagrama del ciclo de vida del modelo en espiral.

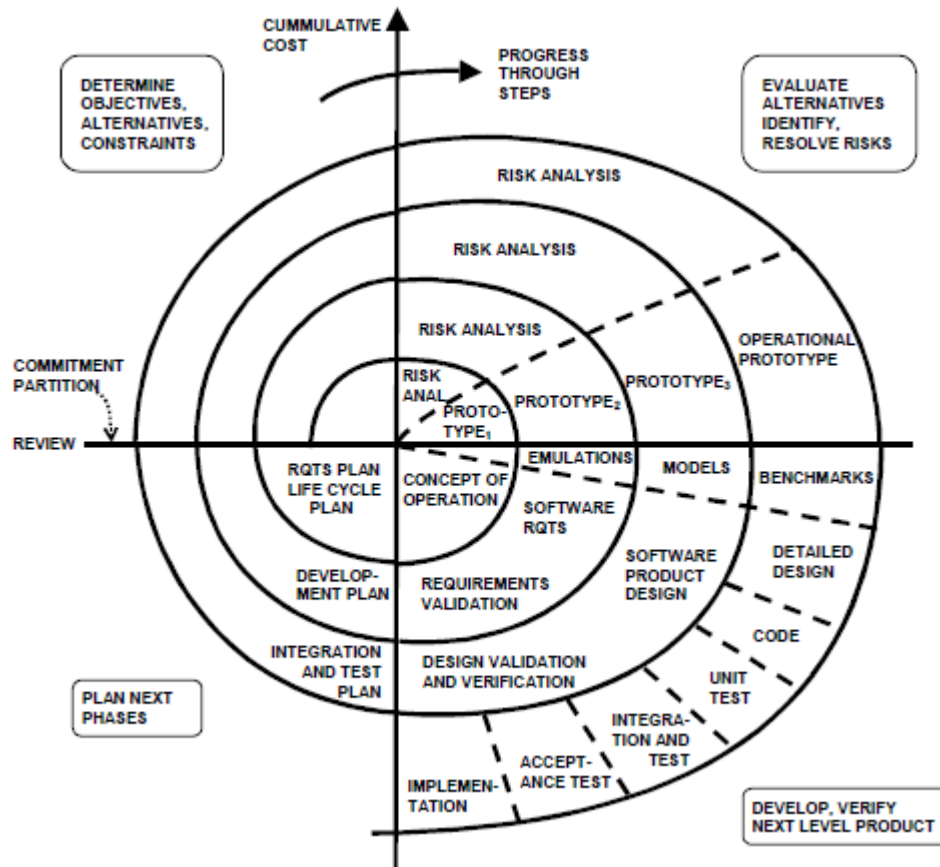


Ilustración 4. Ciclo de vida en espiral en idioma inglés [36].

3.2 Estudio de factibilidad

3.2.1 Factibilidad operativa

El estudio de Factibilidad operativa permite determinar si el desarrollo del sistema propuesto se llevara a cabo, tomando en cuenta los beneficios que aportará a los usuarios.

El proyecto “MEX-CULTURE” tiene como objetivo preservar y diseminar la cultura mexicana, ya que dada la importancia de los contenidos del patrimonio cultural surge la necesidad de crear dicho proyecto, para indexar a las bibliotecas multimedia todo el contenido se necesita analizar varios videos.

El proyecto “MEX-CULTURE” busca elaborar con esfuerzos de investigación bilaterales, nuevos métodos automatizados para procesamiento a gran escala e indexación de contenidos multimedia, además para lograr el objetivo se aplicaran todos los métodos ideados en el proyecto a las grandes bases de datos de la Fonoteca Nacional (Fonoteca Nacional de México), parte de CONACULTA (Consejo Nacional para la Cultura y las Artes de México) y la Videoteca (TVUNAM, más de 100.000 horas de vídeo) de la UNAM (Universidad Nacional Autónoma de

México). El prototipo propuesto AVIN será un módulo para la detección de naturaleza ya que la diversidad de ecosistemas en México es muy amplia. AVIN será parte fundamental para cumplir el objetivo del proyecto “MEX-CULTURE”.

3.2.2 Factibilidad técnica

El análisis de factibilidad técnica es fundamental ya que se debe hacer un análisis de la tecnología requerida para conseguir la funcionalidad y el rendimiento del sistema propuesto. La información acerca de los recursos para poder desarrollar el proyecto “AVIN” se muestra en las siguientes tablas.

En la Tabla 1 se muestran los recursos de hardware para el desarrollo del sistema.

Recurso	Característica
Equipo de computo	Procesador Intel Core I3, 8 GB Memoria RAM, 500 GB capacidad en disco duro
Punto de acceso	Dispositivo para establecer la conexión a Internet. Con red inalámbrica y ancho de banda de 2MB

Tabla 1. Recursos Hardware

El equipo de cómputo a usar nos servirá mayormente para realizar pruebas con los diferentes descriptores, en ocasiones el procesamiento de las pruebas suele ser costoso por lo que en una maquina con las características anteriores es suficiente para desarrollar el proyecto.

El punto de acceso para la conexión a internet nos servirá de ayuda con fines de investigación, algoritmos, artículos, etc. por lo tanto es fundamental tener este recurso a la mano, además de ser una herramienta de comunicación para los integrantes del equipo y así facilitar la organización, el punto de acceso que ocuparemos serán los que se encuentran localizados en las instalaciones de la ESCOM.

En la Tabla 2 se muestran los recursos de software necesarios para el desarrollo del proyecto.

Recurso	Disponible
Sistema Operativo. Windows 7 o Linux(Ubuntu)	Si
MATLAB	Si
Máquina Virtual Java	Si
IDE de desarrollo Java	SI

Tabla 2. Recursos de software

Para el equipo de cómputo se necesitará instalar diferentes programas, se planea trabajar sobre el sistema operativo Windows 7, la otra alternativa es sobre Ubuntu, el software en el que se trabajará principalmente será en MATLAB, ya que se necesita hacer suficientes pruebas para así ir mejorando la precisión del descriptor.

Para el desarrollo total del proyecto se usara algún IDE de desarrollo de Java el cual puede ser ECLIPSE o NETBEANS, para asegurar el correcto funcionamiento del IDE se necesitará la Máquina virtual de Java.

En la Tabla 3 se muestran los recursos adicionales que se necesitarán en el proyecto.

Recurso	Disponible
Base de Imágenes patrón	Si
Videos de prueba.	Si

Tabla 3. Recursos adicionales

Se hará uso de una base de imágenes patrón, la cual nos servirá para el entrenamiento de la red neuronal, esta base de imágenes será propia; se hará búsqueda en la red para elegir las. Cuando se tenga la red entrenada necesitaremos una serie de videos de prueba, para poder determinar si el descriptor tiene un correcto funcionamiento.

3.3 Métricas y estimación

Para el estudio de factibilidad económica utilizamos el métricas orientadas a función que nos ayudarán a estimar el tiempo y esfuerzo que se requiere para realizar este proyecto.

En el proyecto AVIN se creará una aplicación de escritorio que permita el manejo de videos e imágenes, para su clasificación respecto a La Naturaleza para lo cual se tienen en cuenta los siguientes procesos.

Para entrenamiento:

- Búsqueda de imágenes patrón
- Extracción de parches
- Extracción de características
- Proceso de entrenamiento

Para uso de red neuronal entrenada:

- Uso de la red neuronal entrenada
- Extracción de parches
- Extracción de característica
- Uso de red neuronal
- Mensaje de identificación

Para usuario final:

- Selección de video
- Obtención de frames
- Creación de archivo de salida

Estos procesos los clasificaremos de la siguiente manera para poder hacer los cálculos de las métricas de puntos de función:

- **Número de entradas externas (EE)** -Se considerarán formularios de entrada de datos de los usuarios.
 - ✓ Selección de video
- **Número de salidas externas (SE)** -Se considerarán páginas web estáticas o dinámicas, así como reportes.
 - ✓ Reporte final que contiene la clasificación.
 - ✓ Extracción de características
- **Número de consultas externas (CE)** - Se refiere a las interfaces orientadas a mensaje.
 - ✓ Extracción de frames exitosa
 - ✓ Mensaje de procesando información
 - ✓ Mensaje de proceso de clasificación
- **Número de archivos lógicos internos (ALI)** - Se consideran como las tablas lógicas de una base de datos.
 - ✓ Entrenamiento de la red neuronal
 - ✓ Datos de la red entrenada
 - ✓ Resultados del proceso de extracción de características
- **Número de archivos de interfaz externos (AIE)** - Archivos que contienen información que utilizará la aplicación, pero de manera externa.
 - ✓ Videos seleccionados
 - ✓ Frames a utilizar

Valor del dominio de información	Conteo	Factor de Ponderación			
		Simple	Promedio	Complejo	
Entradas Externas (EE)	1	3	4	6	=6
Salidas externas(SE)	2	4	5	7	=14
Consultas externas (CE)	3	3	4	6	=18
Archivos de lógica interna (ALI)	3	7	10	15	=45
Archivos de interfaces externas(AIE)	2	5	7	10	=20
Total de Conteos					103

Para determinar los puntos de función utilizamos la siguiente fórmula:

$$PF = \text{conteo total} \times 0.65 + 0.01 \times F_i$$

Donde conteo total es el Total de la tabla anterior y F_i son factores de ajuste de valor basados en las respuestas a las siguientes preguntas.

Cada una de estas preguntas se responde usando una escala que va de 0 a 5, donde 0 significa “no importante o aplicable” y 5 “absolutamente esencial”.

1. ¿El sistema requiere respaldo y recuperación confiables?

5

Es importante que el sistema pueda recurrar la información de las actividades, para su buen funcionamiento.

2. ¿Se requieren comunicaciones de datos especializadas para transferir información a la aplicación u obtenerla de ella?

2

Es importante trasferir información, para poder realizar las salidas del sistema.

3. ¿Hay funciones distribuidas de procesamiento?

0

Los procesos no son complejos, y no requieren de procesamiento distribuido

4. ¿El desempeño es crítico?

5

Es importante el desempeño de la aplicación, pero al ser una herramienta de apoyo, no interfiere en ningún “critico”

5. ¿El sistema se ejecutara en un entorno existente que tiene un uso pesado de operaciones?

5

Está planeado para poder ejecutarse en una maquina sin un uso pesado de operaciones.

6. ¿El sistema requiere entrada de datos en línea?

0

No son requeridos datos en línea.

7. ¿La entrada de datos en línea requiere que la transacción de entrada se construya en varias pantallas u operaciones?

0

Se utilizara solo una pantalla para la entrada de datos.

8. ¿Los ALI (Archivos lógicos internos) se actualizan en línea?

0

Es necesaria la actualización de los archivos de salida.

9. ¿Las entradas, las salidas, los archivos o las consultas son complejos?

4

Las entradas, salidas, archivos son complejas, debido a que se generan a través de la red entrenada..

10. ¿Es complejo el procesamiento interno?

5

El procesamiento si es complejo, pues se realizan actividades de procesamiento de imágenes.

11. ¿El código diseñado será reutilizado?

5

Es posible reutilizarlo.

12. ¿Se incluyen la conversión e instalación en el diseño?

3

Se toma en cuenta este punto debido a que la aplicación si requiere la instalación de algunas bibliotecas.

13. ¿Está diseñado el sistema para instalaciones múltiples en diferentes organizaciones?

5

Esta planeado para distintos tipos de organizaciones

14. ¿La aplicación está diseñada para facilitar el cambio y para que el usuario lo use fácilmente?

5

Ese es uno de los objetivos principales.

Ya que hemos contestado las preguntas anteriores hacemos el cálculo de F_i

$$F_i = 5 + 2 + 0 + 5 + 5 + 0 + 0 + 0 + 4 + 5 + 5 + 3 + 5 + 5$$

$$F_i = 44$$

Ahora podemos calcular el punto de función.

$$PF = 103 \times 0.65 + (0.01 \times 44)$$

$$PF = 112.27$$

Debido a que el proyecto será realizado utilizando el lenguaje Java, podemos calcular de la siguiente manera:

$$LDC \text{ en Java} = 112.27 \times 63 = 7073.016$$

$$KLDC = LDC/1000 = 7.07301$$

3.3.1 Estimación COCOMO Básico Orgánico

COCOMO básico es un forma rápida y sencilla de estimar la magnitud de los costes de un proyecto software. Se eligió el tipo orgánico debido a que el proyecto no rebasa las 50 KLDC y el grupo de trabajo es pequeño.

Las ecuaciones de COCOMO [37] en este modo básico son:

$$E = a \times KLDC^b D = c \times E^d$$
$$P = E/D$$

Donde E es el esfuerzo aplicado en persona-mes, D es el tiempo de desarrollo en meses, KLOC es el número de líneas estimadas para el proyecto (en miles) y P es el número de personas necesarias. Los coeficientes a, b, c y d se obtienen de la siguiente tabla:

Tabla 14 Coeficientes

Tipo de Proyecto	a	b	c	d
Orgánico	2.4	1.05	2.5	0.38
Medio	3.0	1.12	2.5	0.35
Embebido	3.6	1.20	2.5	0.32

Ahora haremos los cálculos pertinentes, para observar esa estimación:

Esfuerzo en persona-mes

$$E=2.4*7.07^{1.05}= 18.71$$

Tiempo en meses

$$D = 2.5*18.71^{0.38}= 7.6$$

Personas necesarias

$$P = 18.71/7.6=2.4$$

Productividad

Se estima mediante la relación de los PF sobre el Esfuerzo:

$$\text{Productividad} = \frac{103}{18.71} = 5.5$$

Documentación

La documentación está dada por el número de páginas documentadas sobre el punto de función, para esto hacemos una estimación de 100 páginas de documentación aproximadamente

$$\text{Documentación} = \frac{\text{No páginas}}{\text{PF}} = \frac{100}{103} = 0.97$$

Costo

El costo se determina dividiendo el Precio final con los PF.

El precio final se determino en la siguiente tabla, tomando en cuenta el salario de 80uds por hora programador y trabajo de 4 horas por día.

Tabla 1-5 Costos de Desarrollo

Costos de Desarrollo		Costo/Hora 80USD	
Concepto	Tiempo (Días)	Costo	
Arranque del Proyecto			
Levantamiento de Requerimientos	20	6400	
Diseño	41	13120	
Subtotal	61	19520	
Desarrollo del sistema			
Programación del clasificador	30	9600	
Programación de la aplicación	31	9920	
Subtotal	61	19520	
Control de calidad/ Ajustes finales			
Control de calidad	60	19200	
Subtotal	60	19200	
Documentación			
Documento final	27	8640	
Desarrollo del manual de usuario	20	6400	
Subtotal	47	15040	
Total	229	73280	

$$\text{Costo} = \frac{\text{Precio}}{\text{PF}} = \frac{73280}{103} = \$711.4/\text{PF}$$

Calidad

La calidad considera el número de errores que tiene el proyecto. Se considera, aproximadamente 10 errores, por lo tanto:

Tabla 16 Estimaciones

Proyecto	Costo	Calidad	Productividad	Documentación
AVIN	\$73280	10.3	5.5	0.97

3.4 Descripción de herramientas

3.4.1 Software

En este proyecto se necesitarán muchas herramientas, pero las más importantes son las de desarrollo, por lo que hay que elegir el lenguaje de programación en el cual será implementado, de ello depende la rapidez del sistema y consumo de recursos adecuado.

3.4.1.1 Herramientas para el desarrollo de actividades

La principal tarea en el proceso de desarrollo del sistema serán las pruebas ya que a partir de ahí se generarán resultados que nos ayudarán a determinar si nuestro descriptor es preciso.

3.4.1.1.1 Herramienta MATLAB

Es un software matemático que ofrece un entorno de desarrollo integrado con un lenguaje de programación propio el lenguaje M. Entre las funcionalidades básicas se encuentran: la manipulación de matrices, representación de datos y funciones, implementación de algoritmos, creación de interfaces de usuario y la comunicación con programas en otros lenguajes.

El uso de MATLAB será fundamental en el desarrollo del proyecto AVIN. Una vez que tengamos las imágenes patrón, se hará la extracción de parches para después extraer las características, estos procesos son muy costosos y nos llevaría mucho tiempo implementarlo en un lenguaje de bajo nivel, por lo tanto se usará MATLAB ya que por las funcionalidades que tiene nos resulta más eficiente en la ejecución de este tipo de procesos, la extracción de características se hará muchas veces y hacerlo en otro lenguaje nos resultaría muy costoso en cuanto al tiempo.

Una vez que se hicieron las pruebas y ya tenemos análisis de dichas pruebas, se implementará en un lenguaje de programación, de acuerdo al procesamiento de imágenes se realizó el siguiente análisis entre el lenguaje de programación Java y C.

3.4.1.1.2 Lenguaje de programación Java

El lenguaje de programación Java es considerado el mejor lenguaje de desarrollo de aplicaciones. Es un lenguaje de programación orientado a objetos que se utiliza para crear eficientes aplicaciones de computadoras. Java ha sido desarrollado después de incluir los conceptos de diversos lenguajes, como C y C++ y además es un lenguaje de alto nivel.

3.4.1.1.3 Lenguaje de programación C

Es un lenguaje de programación estructurada. C es apreciado por la eficiencia del código que produce y es el lenguaje más popular para crear software de sistemas y servidores, aunque también se utiliza para crear aplicaciones.

Se trata de un lenguaje de tipos de datos estáticos, de medio nivel. Dispone de las estructuras típicas de los lenguajes de alto nivel pero, a su vez, dispone de construcciones del lenguaje que permiten un control a muy bajo nivel. Los compiladores suelen ofrecer extensiones al lenguaje

que permite incluir código en ensamblador con código C o acceder directamente a memoria o dispositivos periféricos.

En la Tabla 5 se presentan las principales características que nos interesan en la implementación de este proyecto.

Característica	JAVA	C
Sencillez	SI	NO
Robustez	SI	SI
Seguridad	SI	SI
Interpretado	NO	NO
Dinamicidad	SI	NO
Portabilidad	SI	NO
Neutralidad	SI	NO
Threads	SI	SI
Excepciones	SI	NO
Representación	ALTA	ALTA

Tabla 7. Tabla comparativa Java y C

El lenguaje Java nos permite trabajar con librerías para el procesamiento de imágenes, además de que es más eficiente implementar algoritmos complejos sobre este. El lenguaje C por ser de medio nivel resulta complejo implementar algún descriptor por lo que nos tomaría mucho tiempo hacer pruebas en este lenguaje. Analizando estos lenguajes de programación, considerando que trabajaremos con procesamiento de imágenes y entrenamiento de redes neuronales concluimos que hacer lo siguiente: se hará pruebas con MATLAB, y hasta tener un descriptor eficiente y preciso se implementara en lenguaje Java ya que por la variedad de las librerías podemos crear interfaces completas y una implementación más eficiente además de que la portabilidad nos permite ejecutarlo en diferentes plataformas.

3.5 Análisis de riesgos

El proceso del análisis del riesgo empieza con un estudio e identificación de los diferentes peligros potenciales y deberá facilitar e identificar los mecanismos más eficientes para proteger el sistema.

En la Tabla 6 se muestran los principales riesgos del producto identificados en el desarrollo del este trabajo terminal AVIN.

Riesgo	Tipo de Riesgo
1. El formato del video de entrada no es compatible, implicando mal procesamiento en la extracción de frames.	Producto
2. El tamaño del video de entrada es demasiado grande por lo que el tiempo de extracción de frames puedes ser de días.	Producto

3. Diseño inadecuado del analizador, generando trabajo de desarrollo costoso.	Producto
4. Las imágenes patrón son inadecuadas para el entrenamiento de la red neuronal, ocasionando un clasificador no funcional.	Producto
5. El algoritmo para la extracción de parches es poco eficiente por lo que el cálculo de las características puede llegar a ser incorrecto	Producto
6. Las imágenes extraídas del video son de baja calidad por lo que el clasificador puede llegar a identificarlo incorrectamente.	Producto
7. Los frames extraídos del video de entrada contienen mucha variación en su contenido por lo que el margen de error será mayor.	Producto

Tabla 8. Riesgos de proyecto para AVIN

Una vez que se han identificado los riesgos, tenemos que decir cuales son más importantes, en la tabla 7 se muestra los riesgos con la probabilidad y sus efectos

Riesgo	Probabilidad	Efectos
1. El formato del video de entrada no es compatible, implicando mal procesamiento en la extracción de frames.	Media	Tolerable
2. El tamaño del video de entrada es demasiado grande por lo que el tiempo de extracción de frames puedes ser de días.	Alta	Tolerable
3. Diseño inadecuado del analizador, generando trabajo de desarrollo costoso.	Media	Serio
4. Las imágenes patrón son inadecuadas para el entrenamiento de la red neuronal, ocasionando un clasificador no funcional.	Baja	Tolerable
5. El algoritmo para la extracción de parches es poco eficiente por lo que el cálculo de las características puede llegar a ser incorrecto	Media	Serio
6. Las imágenes extraídas del video son de baja calidad por lo que el clasificador puede llegar a identificarlo incorrectamente.	Baja	Insignificante
7. Los frames extraídos del video de entrada contienen mucha variación en su contenido por lo que el margen de error será mayor.	Baja	Tolerable

Tabla 9. Probabilidad de riesgos de producto

Una vez que se hizo la probabilidad de todos los riesgos, planteamos unas estrategias en caso de que pase dicho riesgo las cuales se describen en la tabla 8.

Riesgo	Estrategias
1. El formato del video de entrada no es compatible, implicando mal procesamiento en la extracción de frames.	Seleccionar los videos con el formato específico asegurando la compatibilidad para el buen procesamiento.
2. El tamaño del video de entrada es demasiado grande por lo que el tiempo de extracción de frames puedes ser de días.	Anticipar el tiempo de procesamiento, ya que este riesgo es inevitable, introducir videos de menor tamaño.
3. Diseño inadecuado del analizador, generando trabajo de desarrollo costoso.	Rediseñar el analizador cuando el desarrollo dé señales de ineficiencia.
4. Las imágenes patrón son inadecuadas para el entrenamiento de la red neuronal, ocasionando un clasificador no funcional.	Volver a entrenar la red seleccionando las imágenes patrón ideales para lo que se requiera identificar
5. El algoritmo para la extracción de parches es poco eficiente por lo que el cálculo de las características puede llegar a ser incorrecto	Hacer un análisis previo para los diferentes algoritmos para la extracción de parches y así seleccionar el más óptimo.
6. Las imágenes extraídas del video son de baja calidad por lo que el clasificador puede llegar a identificarlo incorrectamente.	Se volverá a seleccionar un video que tenga una calidad mayor y así poder extraer imágenes
7. Los frames extraídos del video de entrada contienen mucha variación en su contenido por lo que el margen de error será mayor.	Entrenar a la red con la restricción de que si no se clasifica en alguna de las clases, las clasifique en una clase desconocida.

Tabla 10. Estrategias para los riesgos de producto

Se identificaron también los riesgos del proyecto y negocio para hacer el proceso anterior, dichos riesgos se describen en la Tabla 9.

Riesgo	Tipo de Riesgo
1. Se subestimó el tamaño del sistema, originando mayores costos.	Proyecto
2. Algún integrante se enferma, generando atrasos en los avances del proyecto	Negocio
3. El equipo de cómputo de los integrantes no está disponible al 100% por lo que resulta imposible avanzar de manera rápida.	Negocio
4. El software a usarse (MATLAB) contiene defectos que limita su funcionalidad	Proyecto

5. La versión de MATLAB usada por los integrantes es diferente por lo que ocasiona incompatibilidad de funcionalidades.	Proyecto
6. El tiempo requerido para desarrollar el software esta subestimado	Proyecto y Negocio

Tabla 11. Riesgos de proyecto para AVIN

La probabilidad de riesgos de proyecto y negocio se describen en la Tabla 10.

Riesgo	Probabilidad	Efectos
1. Se subestimó el tamaño del sistema, originando mayores costos.	Proyecto	Serio
2. Algún integrante se enferma, generando atrasos en los avances del proyecto	Negocio	Tolerable
3. El equipo de cómputo de los integrantes no está disponible al 100% por lo que resulta imposible avanzar de manera rápida.	Negocio	Catastrófico
4. El software a usarse (MATLAB) contiene defectos que limita su funcionalidad	Proyecto	Serio
5. La versión de MATLAB usada por los integrantes es diferente por lo que ocasiona incompatibilidad de funcionalidades.	Proyecto	Tolerable
6. El tiempo requerido para desarrollar el software esta subestimado	Proyecto y Negocio	Serio

Tabla 12. Probabilidad de riesgos de proyecto y negocio

Se plantearon las diferentes estrategias a aplicar para los riesgos de proyecto y negocio en la Tabla 11

Riesgo	Estrategias
1. Se subestimó el tamaño del sistema, originando mayores costos.	Hacer un nuevo análisis de los alcances y delimitar el tamaño del proyecto
2. Algún integrante se enferma, generando atrasos en los avances del proyecto	Reorganizar las tareas con los integrantes restantes hasta que el integrante se incorpore.

3. El equipo de cómputo de los integrantes no está disponible al 100% por lo que resulta imposible avanzar de manera rápida.	Tener equipo de cómputo de respaldo que tenga instalado el software que se ocupara.
4. El software a usarse (MATLAB) contiene defectos que limita su funcionalidad	Si es necesario reinstalar el software en el equipo de computo
5. La versión de MATLAB usada por los integrantes es diferente por lo que ocasiona incompatibilidad de funcionalidades.	Identificar las funcionalidades a usar y establecer una versión del software para que todos lo usen.
6. El tiempo requerido para desarrollar el software esta subestimado	Replantear las actividades con los integrantes, asignando mucho más tareas para poder cubrir con el tiempo establecido

Tabla 13. Estrategias para los riesgos de proyecto y negocio

4 Diseño

4.1 Diagramas de casos de uso AVIN

4.1.1 Diagrama de caso de uso. Sistema AVIN

En el diagrama 1 DCUA muestra el diagrama del caso de uso del sistema AVIN, este diagrama solo cuenta con un actor como usuario y dos casos de uso, selecciona video es para seleccionar e indicarle al sistema cual es el video que se va a analizar e inicia analizador es para iniciar el analizador de video.

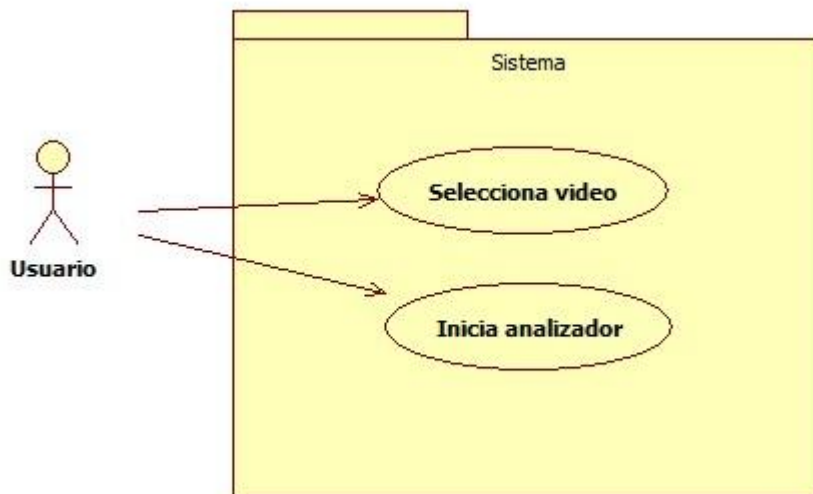


Diagrama 2. Caso de Uso del Sistema (CUS).

4.1.2 Caso de uso. Selecciona video

Resumen.

Este caso de uso brinda al usuario la posibilidad de buscar en el sistema el video que desea analizar. El video podrá ser en diferentes formatos los cuales son MP4, AVI, FLV y MPEG.

La tabla 12 muestra la descripción de este caso de uso

Descripción.

Caso de uso: Selecciona video	
Resumen de atributos	
Actor:	Usuario.
Propósito:	Permite al usuario seleccionar el video que desea analizar.
Entradas:	Para llevar a cabo el video que desee el usuario deberá realizar los siguientes pasos: <ul style="list-style-type: none">• Se presiona el botón Seleccionar video.• Se selecciona con el mouse el video.• Se presiona el botón Aceptar.
Salidas:	Se mostrara en la interfaz del sistema las características del video.
Precondiciones:	<ul style="list-style-type: none">• Se requiere que existan videos relacionados con naturaleza.• Es necesario que el video sea del formato válido.
Errores:	<ul style="list-style-type: none">• Se mostrara en la pantalla el mensaje Archivo inválido cuando el usuario no haya seleccionado un video con formato válido.

Tabla 14. Caso de uso selecciona video

Trayectorias del caso de uso.

Trayectoria principal.

1. Usuario: Desea seleccionar un video para analizar presionando el botón Seleccionar video en la interfaz del sistema.
2. Le muestra al usuario una pantalla donde podrá buscar el video que desee.
3. Solicita al usuario seleccionar un video presionando el botón Aceptar. [Trayectoria A] [Trayectoria B].

Trayectoria alternativa A

Condición: El usuario desea cancelar la operación.

A-1. Usuario: solicita cancelar la operación presionando el botón Cancelar.

A-2. Cancela la operación y regresa a la interfaz de usuario.

A-3 Fin de caso de uso.

---- fin de trayectoria.

Trayectoria alternativa B

Condición: El usuario selecciona un archivo con formato inválido.

B-1. Usuario: Selecciona archivo con formato inválido presionando el botón **Aceptar**.

B-2. Muestra en la pantalla el mensaje **Archivo inválido** indicando que no es posible analizar ese archivo/video.

B-3 Fin de caso de uso.

---- *fin de trayectoria.*

4.1.3 Caso de uso. Inicia Analizador

Resumen.

Este caso de uso inicia el procedimiento para analizar un video y proporcionarle al usuario un archivo de texto con información de escenas que se encuentran en dicho video.

La tabla 13. Caso de uso inicia analizador, muestra la descripción de este caso de uso.

Descripción.

Caso de uso: Inicia analizador	
Resumen de atributos	
Actor:	Usuario.
Propósito:	Proporcionarle al usuario un archivo de texto con información del contenido en el video.
Entradas:	Para llevar a cabo el archivo de texto con información del video que el usuario ha seleccionado deberá realizar los siguientes pasos: <ul style="list-style-type: none">• Se presiona el botón Analizar video.
Salidas:	Se mostrara el archivo de texto con información de escenas que contiene el video.
Precondiciones:	<ul style="list-style-type: none">• Se requiere que el usuario haya seleccionado un video.
Errores:	<ul style="list-style-type: none">• Se mostrara en la pantalla el mensaje Error al crear archivo cuando el sistema le sea imposible crear el archivo de salida.

Tabla 15. Caso de uso inicia analizador

Trayectorias del caso de uso.

Trayectoria principal.

1. Usuario: Desea analizar un video para obtener un archivo de texto con sus características de contenido presionando el botón **Analizar video** en la interfaz del sistema.
2. Le muestra al usuario un archivo de texto con información obtenida del contenido del video. [Trayectoria A] [Trayectoria B].

Trayectoria alternativa A

Condición: El usuario desea cancelar la operación.

A-1. Usuario: solicita cancelar la operación presionando el botón **Cancelar**.

A-2. Cancela la operación y regresa a la interfaz de usuario.

A-3 Fin de caso de uso.

---- *fin de trayectoria.*

Trayectoria alternativa B

Condición: El usuario no ha seleccionado un video.

B-1. Usuario: Presiona el botón **Aceptar** sin haber seleccionado un video.

B-2. Muestra en la pantalla el mensaje **Selecciona un video** indicando que es necesario que seleccione el video que desea analizar.

B-3 Fin de caso de uso.

---- *fin de trayectoria.*

4.2 Diagrama de procesos

El diagrama 2 representa los principales procesos que se desarrollaran en este Trabajo Terminal, en el cual se describe el entrenamiento de la red neuronal a usar y el proceso para usar la red entrenada.

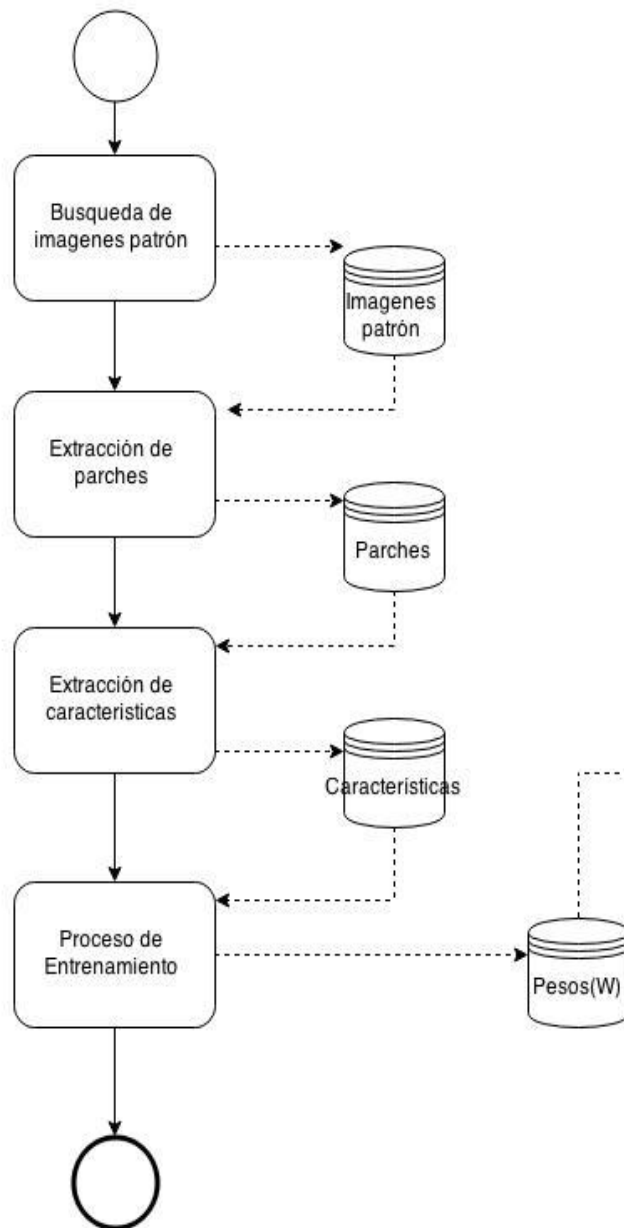
El primer paso en el proceso de entrenamiento consiste en realizar una búsqueda de imágenes patrón, la cual se guarda en una base de datos temporal, estas imágenes nos servirán para poder establecer una base de identificación de elementos. El siguiente paso es la extracción de parches, se tiene que hacer divisiones del mismo tamaño a las imágenes patrón, estos parches al igual que las imágenes patrón se guardan en una base temporal para que posteriormente se haga la extracción de características las cuales son media, desviación estándar, entropía, asimetría. Estas características se guardarán en una base de datos y serán las entradas en el proceso de entrenamiento de la red neuronal.

Al entrenar la red neuronal guardaremos los pesos (w) en una base de datos porque son los que usaremos en el proceso de identificar los elementos en los frames.

Para el uso de la red neuronal previamente entrenada, primero tendremos los frames a analizar los cuales para nuestro caso serán las imágenes extraídas del video a analizar, se extraen los parches y se guardan, se extraen las características de esos parches y se guardan para que sean la entrada de la red neuronal, ya que para el uso de la red neuronal debemos tener los pesos del

entrenamiento y las características de las imágenes a analizar, el proceso termina cuando la red neuronal genera un mensaje (o archivo de texto) de salida identificando los tipos de elementos que se encuentran en dichos frames a analizar.

ENTRENAMIENTO DE RED NEURONAL



USO DE LA RED NEURONAL ENTRENADA

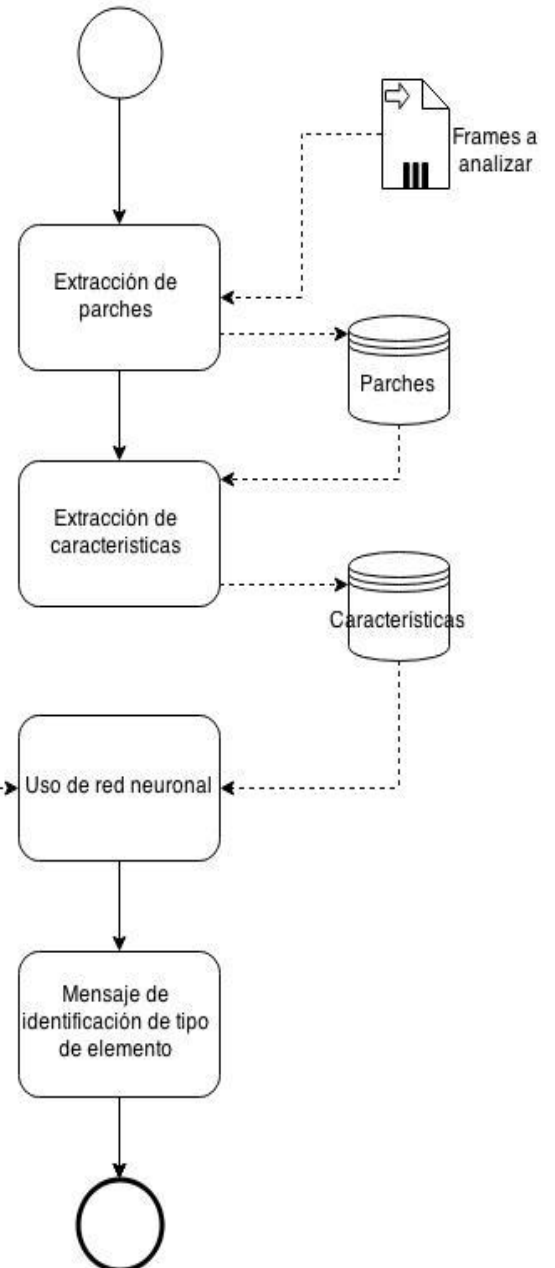


Diagrama 3 Procesos.

4.3 Diagramas de secuencia.

4.3.1 Diagrama de secuencia Selecciona video (DSSV)

En el diagrama 3. DSSV se muestra la secuencia del caso de uso Selecciona video, este cuenta con tres objetos Inicio, Interfaz y Sistema. Inicio representa al usuario, Interfaz es la pantalla principal de AVIN y Sistema es la plataforma donde se está ejecutando.

Resumen.

El diagrama 3. DSSV representa la secuencia de interacción entre el usuario y la interfaz para seleccionar el video que se desea analizar.

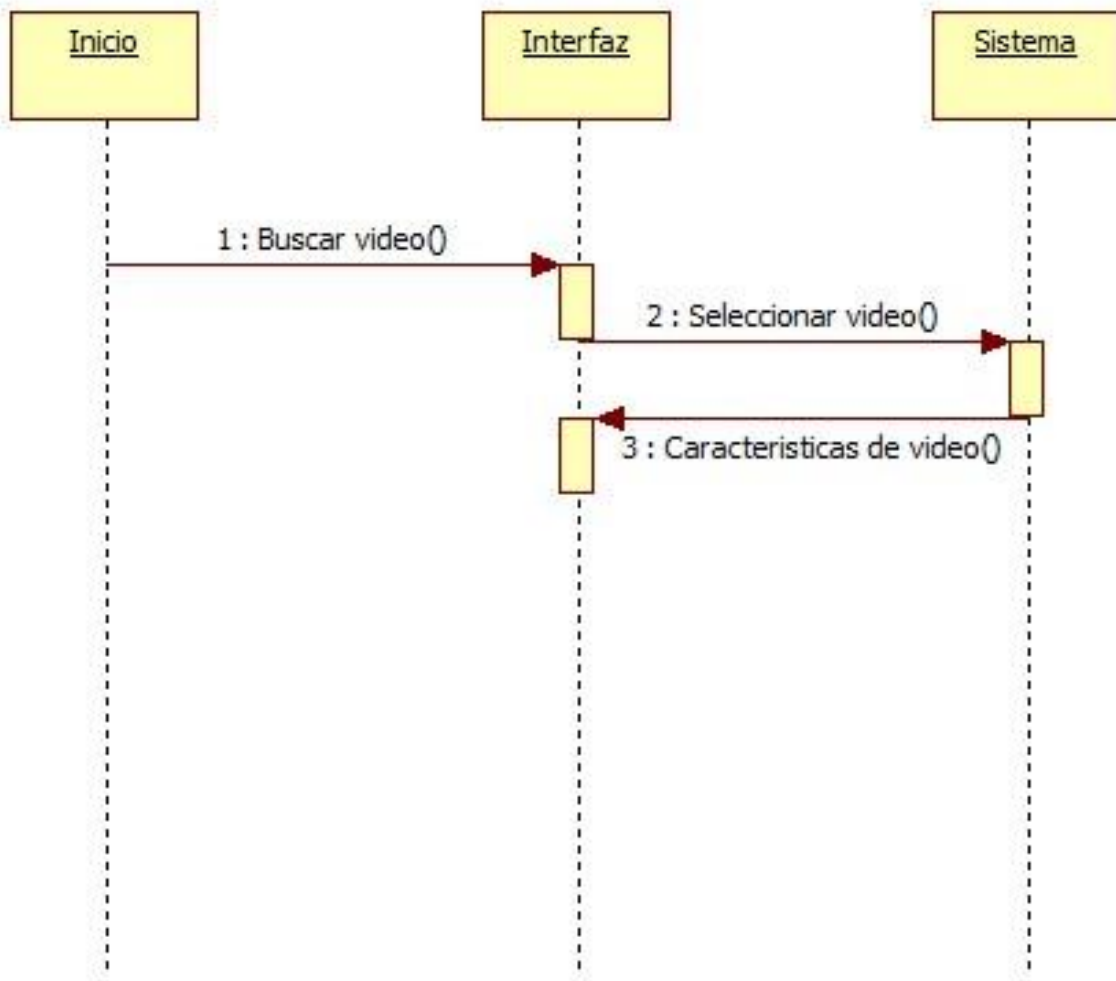


Diagrama 4. Diagrama de Secuencia Selecciona Video (DSSV).

4.3.2 Diagrama de Secuenciade Archivo (DSA)

En el diagrama 4. DSA se muestra la secuencia del caso de uso Inicia analizador, este diagrama cuenta con 7 objetos Inicio, Interfaz, Extracción parches, Extracción características, Entrenamiento red, Base de datos y Archivo salida.

Resumen.

El diagrama 4 DSA representa la secuencia de interacción entre el usuario y la interfaz para obtener un archivo de salida con características del contenido del video basándose en naturaleza.

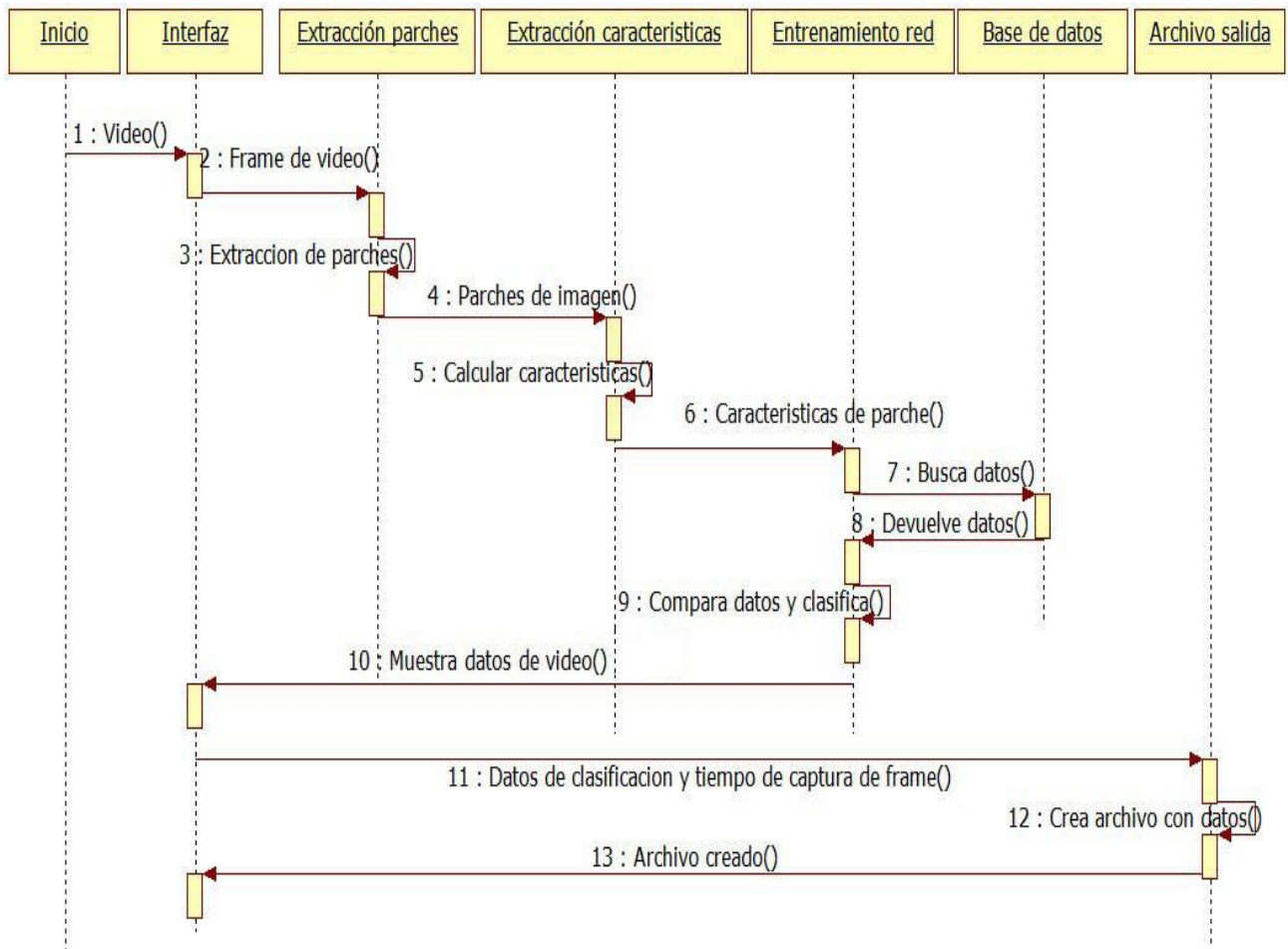


Diagrama 5. Diagrama de Secuencia de Archivo (DSA).

4.4 Diagrama de Clases AVIN (DCA)

En el diagrama 5. DCA se muestran las clases del sistema AVIN y se describe el entrenamiento de la red neuronal y el funcionamiento del sistema.

Resumen

La clase de inicio es la principal del sistema porque es donde inicia el proceso de entrenamiento de la red neuronal y el funcionamiento del sistema. Para el entrenamiento se inicia cargando una imagen patrón, posteriormente esta se envía a la clase extracción de parches donde se obtendrán aleatoriamente parches de la imagen. La clase de extracción de características recibirá los parches de la clase anterior y creará vectores característicos por cada parche y estos cálculos se envían a la clase red neuronal para que se entrene y guarde estos datos en la base de datos del sistema AVIN.

Para el funcionamiento de AVIN es un proceso similar al del entrenamiento de la red neuronal mencionado anteriormente a excepción de que en la clase de inicio se selecciona un video. La clase archivo de salida será la que genere toda la información que obtuvo la red neuronal al clasificar la imagen, colocando en el archivo de texto la escena que representa el fotograma y el tiempo en el que fue tomado del video.

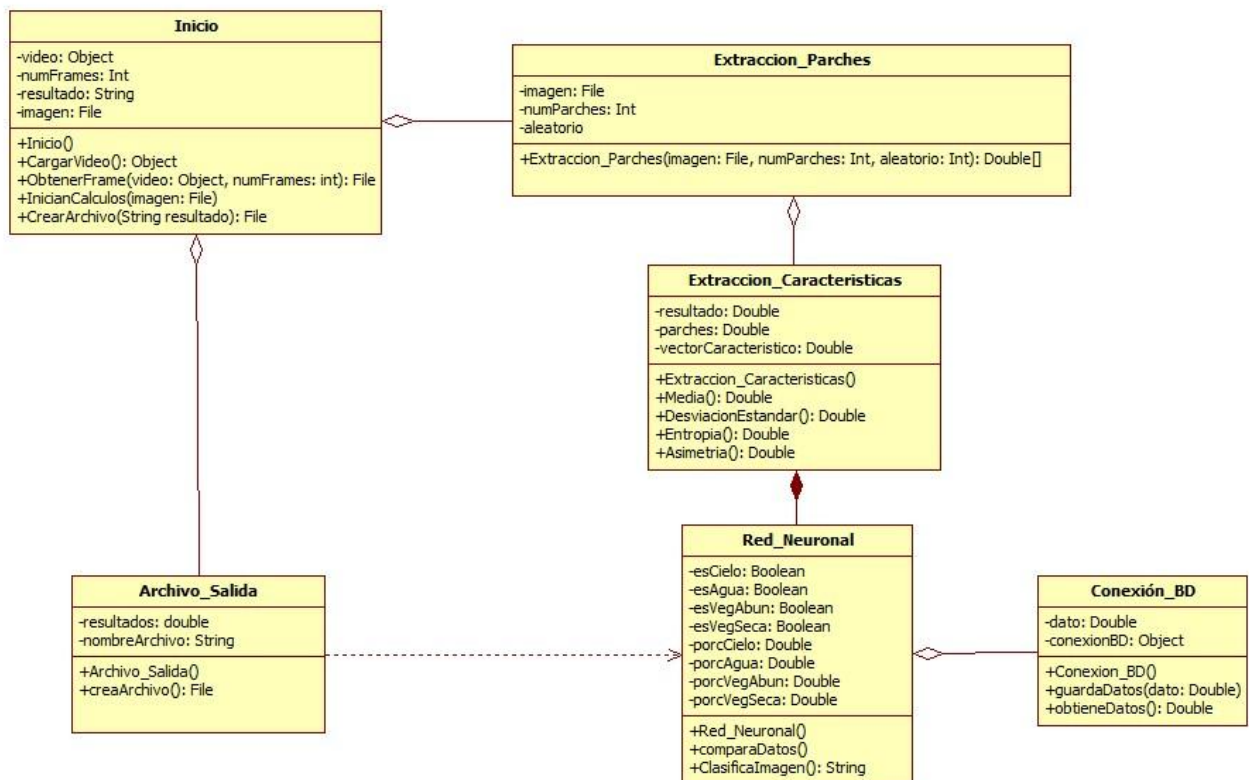


Diagrama 6. Diagrama de Clases AVIN (DCA).

4.5 Interfaz gráfica del sistema AVIN



Ilustración 5. Interfaz principal



Ilustración 6. Búsqueda de archivo

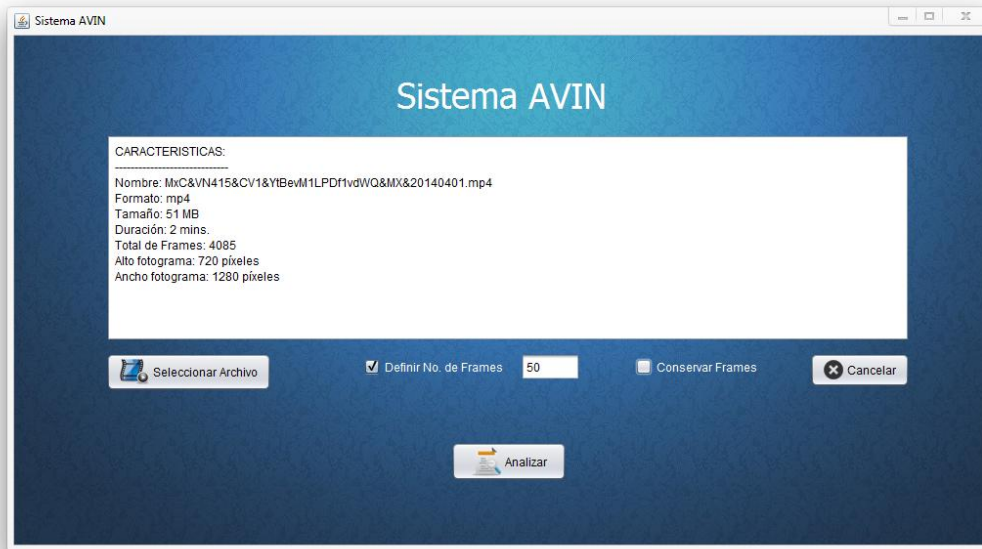


Ilustración 7. Descripción del archivo

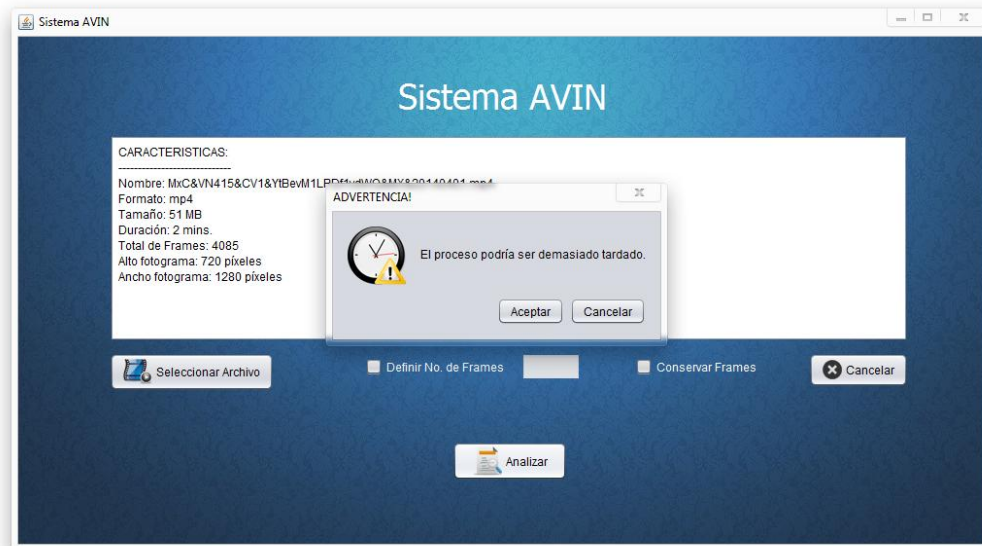


Ilustración 8. Mensaje de advertencia

5 Desarrollo

5.1 Segmentación de la imagen.

La imagen para el entrenamiento de las redes neuronales y para el análisis del sistema AVIN se hace mediante un método de malla (GRIP) se divide la imagen de manera que se obtengan 100 parches de esta; es decir 10 parches de modo vertical y 10 de modo horizontal, como se muestra en la siguiente imagen. Con un método de malla obtenemos un mayor número de datos de la imagen.

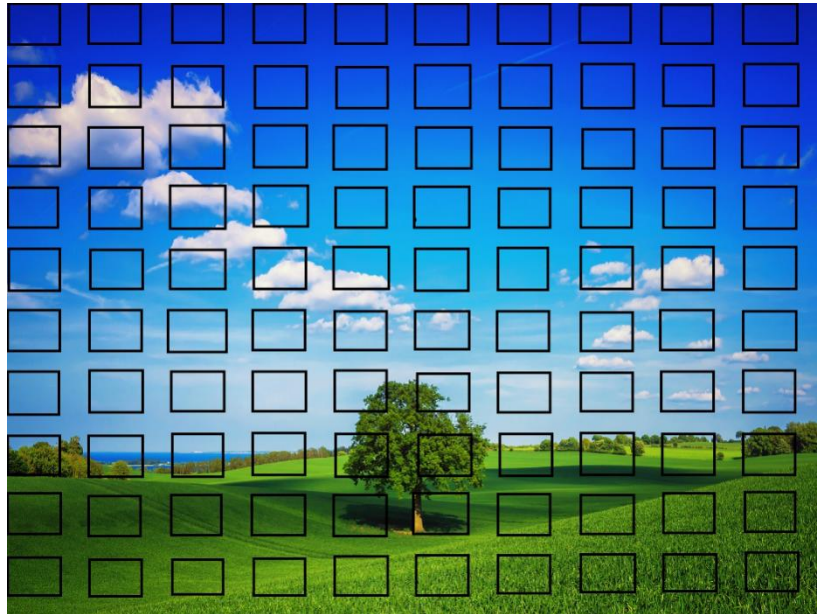


Ilustración 9. Segmentación de la imagen

5.2 Procedimiento de las etapas del sistema AVIN

El proceso del sistema AVIN se observa en el siguiente diagrama de flujo; se tiene un video de formato mp4 posteriormente se extraen N frames y cada frame se le aplica un proceso como el del diagrama, primero se divide la imagen en 100 parches del mismo tamaño en modo de malla, después a cada parche se obtienen sus características de estos los cuales son la media y la desviación estándar sobre el modelo de color RGB, una vez realizado este proceso obtenemos 6 vectores de características.

La red neuronal fue entrenada en el software Matlab y los datos obtenidos del entrenamiento son guardados en un archivo de texto (txt) para que sean evaluados con los datos obtenidos a partir de las características. Este proceso se realiza en dos etapas con dos funciones de transferencia.

5.3 Proceso del sistema AVIN.

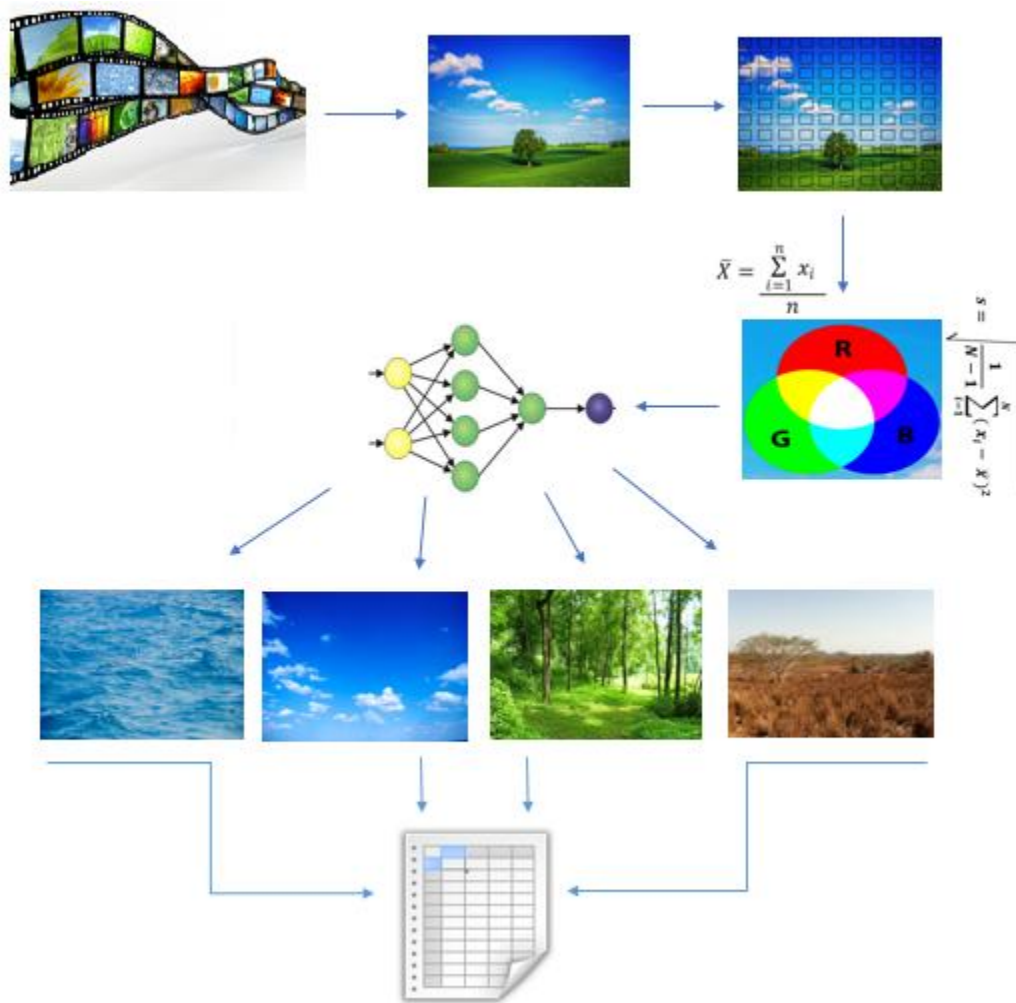


Ilustración 10. Proceso sistema AVIN.

5.4 Red Feedforward (FF)

La red neuronal Feedforward está compuesta por dos capas como se observa en la siguiente imagen:

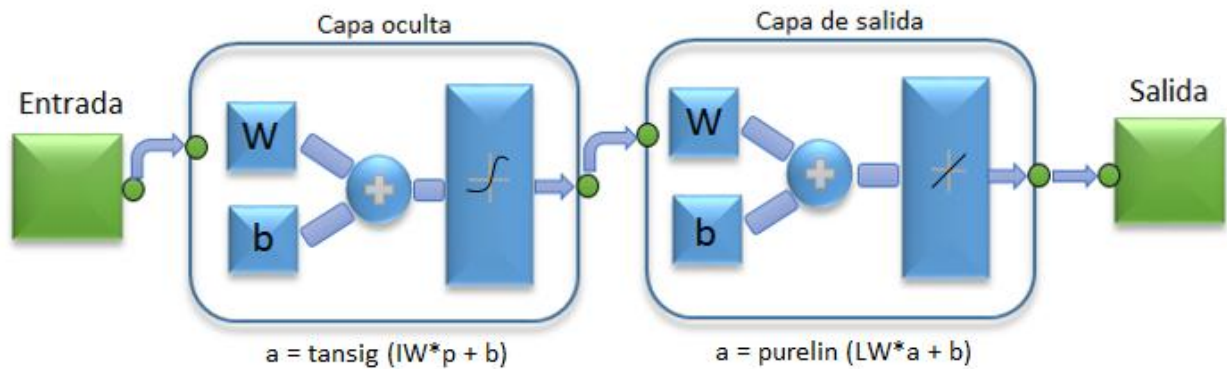
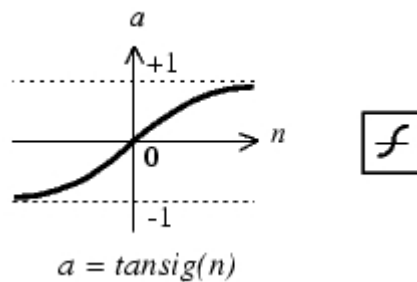


Ilustración 11. Arquitectura de la red FeedForwarded.

En la primera etapa se aplica la función TANSIG (esta es la notación utilizada por MATLAB), esta función básicamente es aplicar al vector de salida una tangente hiperbólica (tanh) la cual nos define si el resultado que devuelve es 1 o -1 su función la podemos ver en la siguiente gráfica.

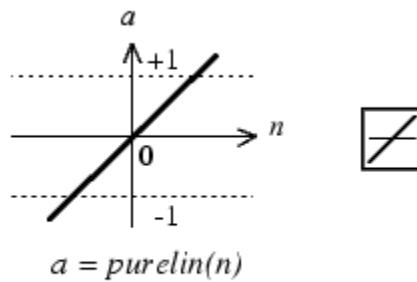


Y está representada por la siguiente fórmula:

$$\text{tansig}(x) = \frac{2}{1 + \exp(-2x)} - 1.$$

Donde x es igual al valor de la característica calculada de cada frame.

En la segunda etapa de la misma manera al vector de salida se aplica la función de transferencia PURELIN (notación utilizada por MATLAB) esta consta de devolver la entrada del mismo modo, su gráfica la observamos en la siguiente imagen.

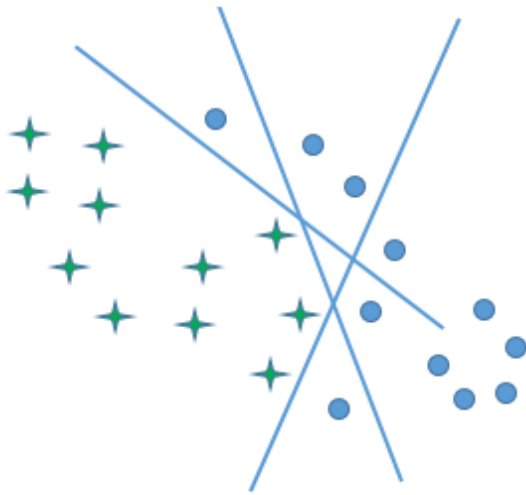


Linear Transfer Function

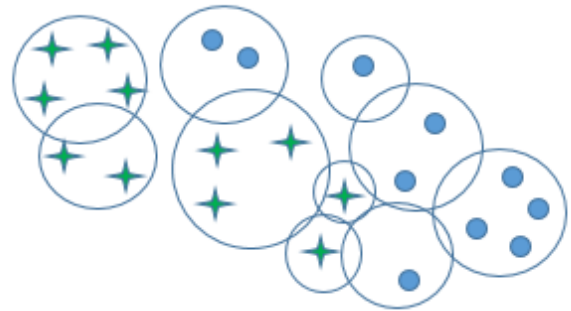
Y su fórmula está dada por:

5.4.1 Clasificación con la red FeedForward

La clasificación de las clases en la red neuronal Feedforward es a través de rectas en un espacio de n dimensiones, se crean rectas para separar los datos, agrupando estos en nubes o clusters, de esto proviene que se tengan errores en la clasificación debido a que si llegase un dato que no pertenece a una clase pero se encuentre en cualquier plano esto provocara que se clasifique en una clase, esto es un problema que causa la red Feedforward que siempre clasificará cualquier dato aunque no pertenezca a esa clase.



FeedForward



Base Radial

Ilustración 12. Clasificación de las clases en FeedForward y Base Radial.

El proceso de entrenamiento de la red neuronal se representa a continuación con un diagrama de flujo (diagrama 6). El cual se observan los pasos que conlleva para obtener un buen descriptor a partir del entrenamiento de la red neuronal. La precisión del descriptor dependerá principalmente de la selección de las imágenes patrón.



Diagrama 7. Entrenamiento de la red neuronal

El proceso general que consiste desde seleccionar el video a analizar hasta que se generara un archivo de texto con el elemento identificado, se muestra en el diagrama 7.

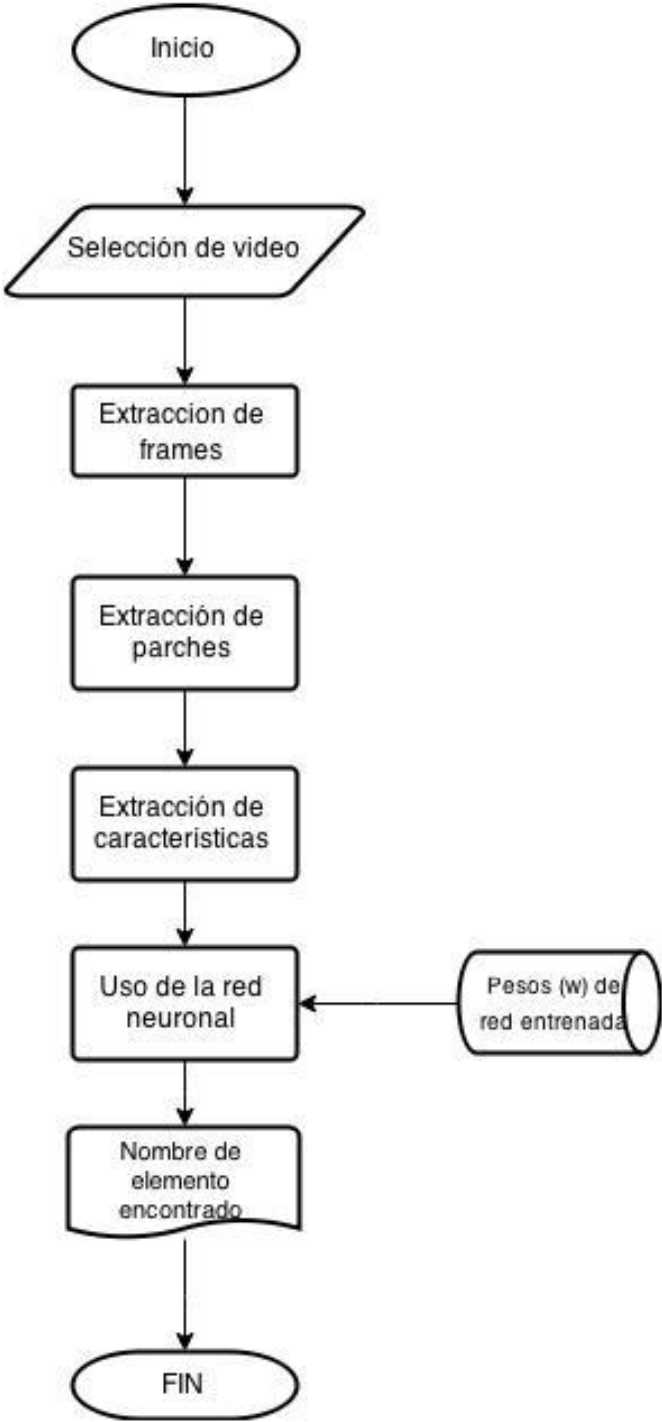


Diagrama 8. Uso de red neuronal

5.5 Red de Base Radial

A continuación se describe una red de base radial para poder mejorar la clasificación de las clases, esta red constituye de un centro y un radio el cual se usa para identificar las clases que pertenecen en dichas clases.

Es importante definir como se hicieron los cálculos que se han usado a lo largo de este trabajo para mejorar la comprensión del lector; el término que usaremos será la de la **distancia euclidiana** propuesta por el matemático Euclides en la cual se calculó con lo siguiente, la distancia entre los puntos $\mathbf{P} = (p_1, p_2, \dots, p_n)$ y $\mathbf{Q} = (q_1, q_2, \dots, q_n)$ del espacio euclídeo n-dimensional, se define como:

$$d_E(P, Q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \dots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^n (p_i - q_i)^2}.$$

5.5.1 Entrenamiento

El proceso de entrenamiento para la primera prueba es calcular un centro y un radio, este centro se calculara a partir de la extracción de características de las imágenes patrón, se tendrán ciertos vectores de características de las cuales se obtendrá el promedio para así definir el centro, el radio se calculará una vez obtenido el centro, calculando la distancia euclidiana a cada uno de los vectores que conforman la clase de todos los datos de cada clase.

Para expresar de una manera gráfica de cómo queremos implementar la red de base radial observamos un ejemplo de grafica de los puntos para vegetación abundante en dos dimensiones:

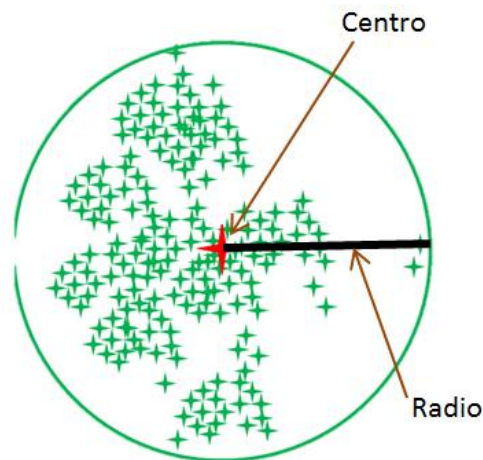


Ilustración 13. Representación del cálculo del centro y radio de cada clase

Expresaremos el entrenamiento de base radial de acuerdo al siguiente diagrama:

ENTRENAMIENTO DE RED NEURONAL
DE BASE RADIAL

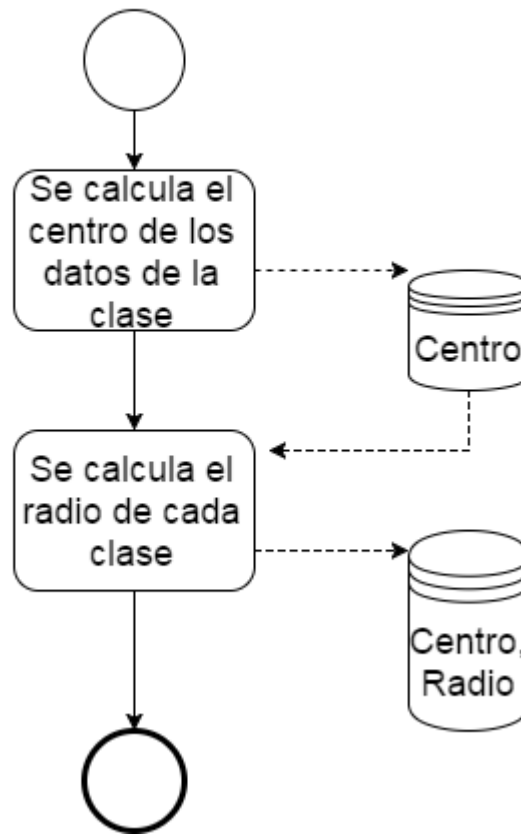
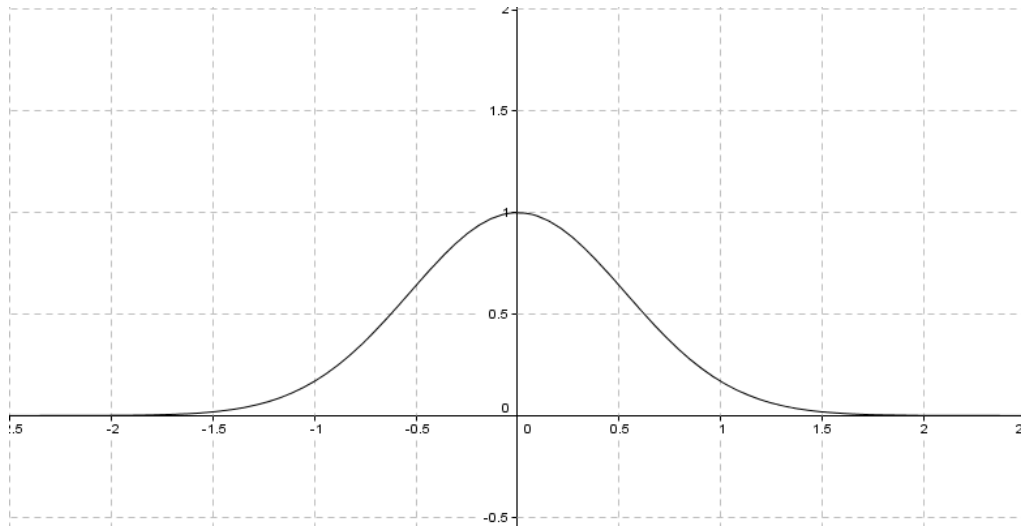


Diagrama 9. Proceso de entrenamiento de la red de base radial

Una vez que nuestra red neuronal de base radial ya quedo entrenada guardaremos los centros y radios en un archivo para poder usarlo en el proceso de simulación.

5.5.2 Proceso de simulación

Para el proceso de simulación se aplica una función Gaussiana la cual es representada en la siguiente imagen:



La función gaussiana que usamos es la siguiente:

$$e^{-\left(\frac{\|Rc-R\|}{\frac{\sigma}{2}}\right)^2}$$

Dónde:

Rc = Vector centro de la clase.

R = Vector a evaluar

σ = radio de la clase

Observando la gráfica de la función gaussiana tomamos como criterio que si evaluamos un vector con el centro (Rc) y radio (σ) obtenemos un valor el cual se debe de acercar al centro de la clase para que pueda ser clasificado, si el punto está lejos entonces el valor de la gaussiana se acercara más a cero. Las gráficas de cada clase se muestran a continuación, cabe aclarar que las gráficas mostradas son en el caso ideal en el que las clases están separadas.

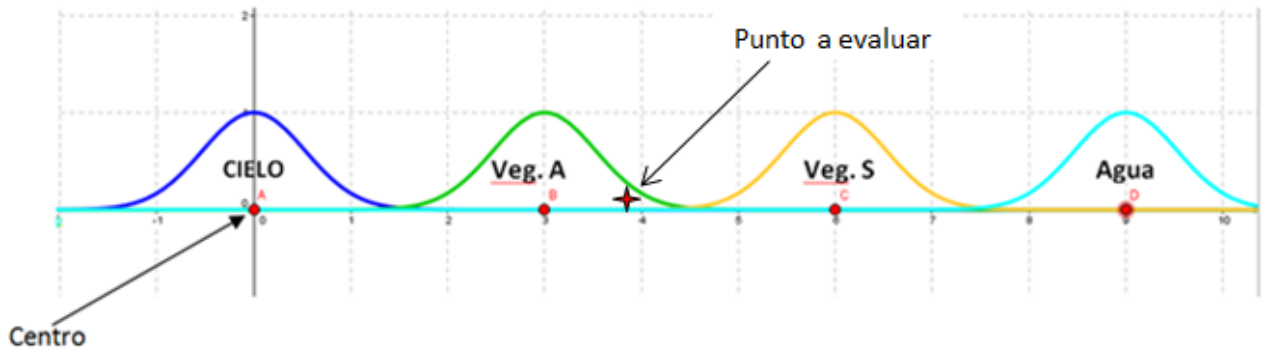
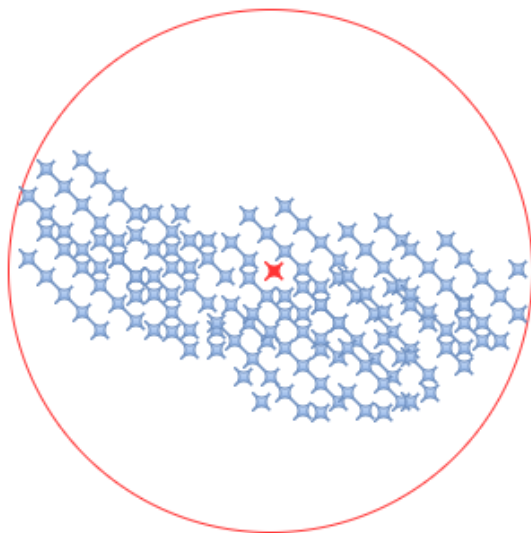


Diagrama 10. Representación Gaussiana

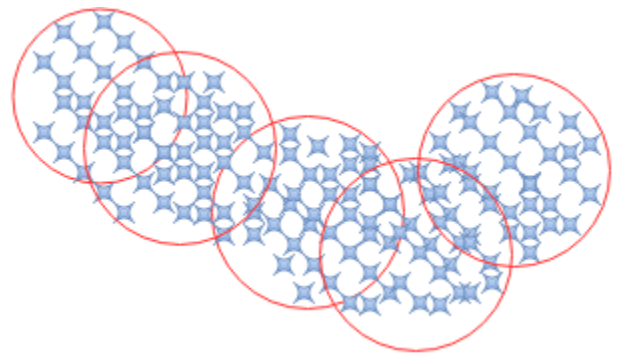
El punto a evaluar se hará con respecto a cada centro y se hará comparación, el resultado de la gaussiana que más se acerque a 1 es en la que pertenece.

5.5.3 Entrenamiento de la red de base radial con algoritmo K-Means

El problema que se presentó en el entrenamiento de la red de base radial, es que los círculos se traslapaban mucho y creaba confusión a la hora de clasificar, con el algoritmo de clusterización creamos círculos más pequeños dentro de la clase para así poder tener una precisión mucho mayor, a continuación se muestra la solución de cada clase con los círculos.



Red de base radial.



Red de base radial con K-Means

Diagrama 11. Comparación de distintos algoritmos

6 Pruebas

6.1 Módulo de extracción de características

Para el módulo de extracción de características, en la fase de prueba utilizamos MATLAB como herramienta de desarrollo.

Este módulo se encarga de extraer pequeños parches de la imagen y calcular de cada uno de ellos la media, desviación, asimetría y entropía. Los parches son secciones de la imagen que se toman como muestra y tienen un tamaño 20x20 pixeles.

La función principal es la de extracción de características, se encarga de:

- Asignar el tamaño del parche y de los saltos a realizar entre cada parche
- Definir los vectores en los cuales se guardara la información obtenida
- Leer las imágenes a partir de una carpeta y de manera secuencial (esto para el caso de entrenamiento)
- Utilizar la función estadística para realizar los cálculos
- Guardar los resultados de las características en un vector

A continuación se muestra el pseudocódigo de este módulo.

```
/* pseudocódigo extracción de características */
```

```
funcionextraccionCaracteristicas
    tamParche<- 20 %tamaño del parche
    salto<- 50 %salto al siguiente parche

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% VectorCaracteristico%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
mediaR<- [] mediaG<- [] mediaB<- []   %% MEDIA %%
stdR<- [] stdG<- [] stdB<- []   %% VARIANZA %%
skewR<- [] skewG<- [] skewB<- []   %% ASIMETRIA %%
entR<- [] entG<- [] entB<- []   %% ENTROPIA %%

    para k <- 1 hasta 5
        numero<- convertirEntero2String(k)
        nombreImg<- concatenar('nombreImagen')

        Irgb<- leerImagen(nombreImg)
        estRgb<- estadistica(Irgb , salto , tamParche)
        mediaR<- [mediaR , estRgb.m1] mediaG<- [mediaG , estRgb.m2] mediaB
= [mediaB , estRgb.m3]
        stdR<- [stdR , estRgb.d1] stdG<- [stdG , estRgb.d2] stdB<-
[stdB , estRgb.d3]
        skewR<- [skewR , estRgb.s1] skewG<- [skewG , estRgb.s2] skewB<-
[skewB , estRgb.s3]
        entR<- [entR ,estRgb.en1] entG<- [entG ,estRgb.en2] entB<- [entB
,estRgb.en3]

    fin

vectorCaracteristico = [mediaR;mediaG;mediaB; stdR;stdG;stdB;
skewR;skewG;skewB; entR;entG;entB]
```

```
fin
```

La función estadística se utiliza para:

- Recorrer y extraer los parches de cada imagen
- Calcular media, desviación estándar, asimetría y entropía de cada uno de los tres planos de color (RGB)
- Guarda los datos en un vector y lo devuelve

El pseudocódigo de la función estadística se muestra a continuación.

```
/* pseudocódigo estadística */
```

```
funcionestadistica(imagen,tamParche,salto)

[m , n , p] <- tamaño(imagen)
ind<- 0;
para i <- 1; hasta m; salto+salto
para j <- 1; hasta n; salto+salto
parche =imcrop(imagen,[i,j,tamParche-1,tamParche-1])

        [x,y,z] <- tamaño(parche)
if(x==tamParche& y==tamParche& z==3)
ind<- ind+1

        R    <- parche(: , : , 1)
        G    <- parche(: , : , 2)
B    <- parche(: , : , 3)

est.m1(ind) <- media(R(:))
est.m2(ind) <- media(G(:))
est.m3(ind) <- media(B(:))

est.d1(ind) <- desviacion(double(R(:)))
est.d2(ind) <- desviacion(double(G(:)))
est.d3(ind) <- desviacion(double(B(:)))

est.s1(ind) <- asimetria(double(R(:)))
est.s2(ind) <- asimetria(double(G(:)))
est.s3(ind) <- asimetria(double(B(:)))

est.en1(ind) <- entropia(double(R(:)))
est.en2(ind) <- entropia(double(G(:)))
est.en3(ind) <- entropia(double(B(:)))
fin
fin
        devuelveest
fin
```


6.2 Pruebas de descriptores

Para identificar las características que describen a los elementos de la naturaleza y que son adecuadas para separar las clases propuestas aquí, lo que se hizo fue obtener los momentos de probabilidad descritos en el marco teórico (entropía, asimetría, media, desviación), así como los descriptores de textura en niveles de gris a partir de la base de imágenes patrón, de los elementos a identificar (agua, cielo, vegetación abundante y escasa), que se construyó para este proyecto.

6.2.1 Pruebas de los descriptores en el plano RGB

En esta sección se muestran las pruebas realizadas, a la base de imágenes patrón antes mencionada, de los descriptores, media, desviación, entropía y asimetría, en los planos R, G y B.

En la ilustración 14 se muestra cómo se separan las clases a través de la asimetría. En esta imagen podemos ver representado el cielo con el color cyan, el agua con el color azul marino, la vegetación abundante con el color verde y la vegetación escasa con el color amarillo. Observamos que las clases son separadas en gran parte de la nube de muestras que están graficadas, en el centro se ve que se unen estas nubes de datos, esto podría representar una confusión a la hora de clasificar.

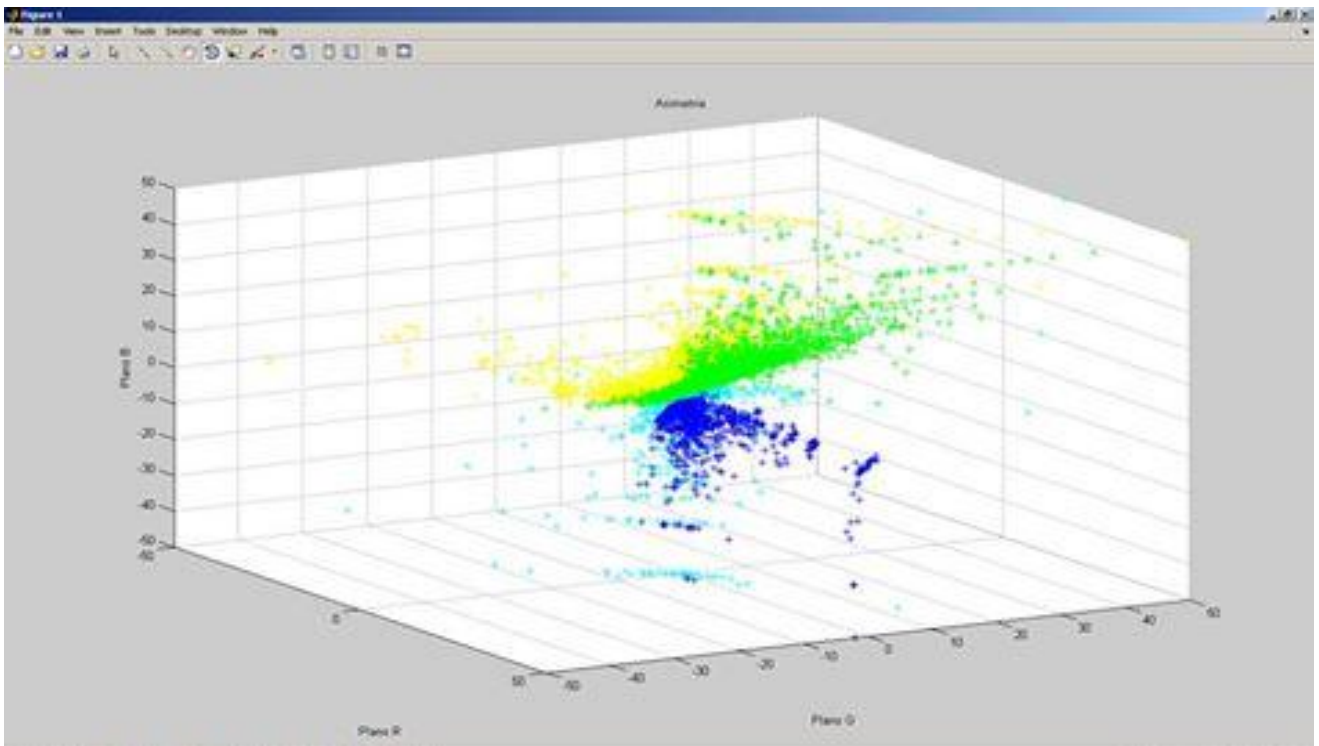


Ilustración 14. Asimetría

La Ilustración 15 muestra la separación de clases a través de la media aritmética, esta característica es muy buena pues las clases están separadas casi por completo y esto se observa muy bien en la imagen.

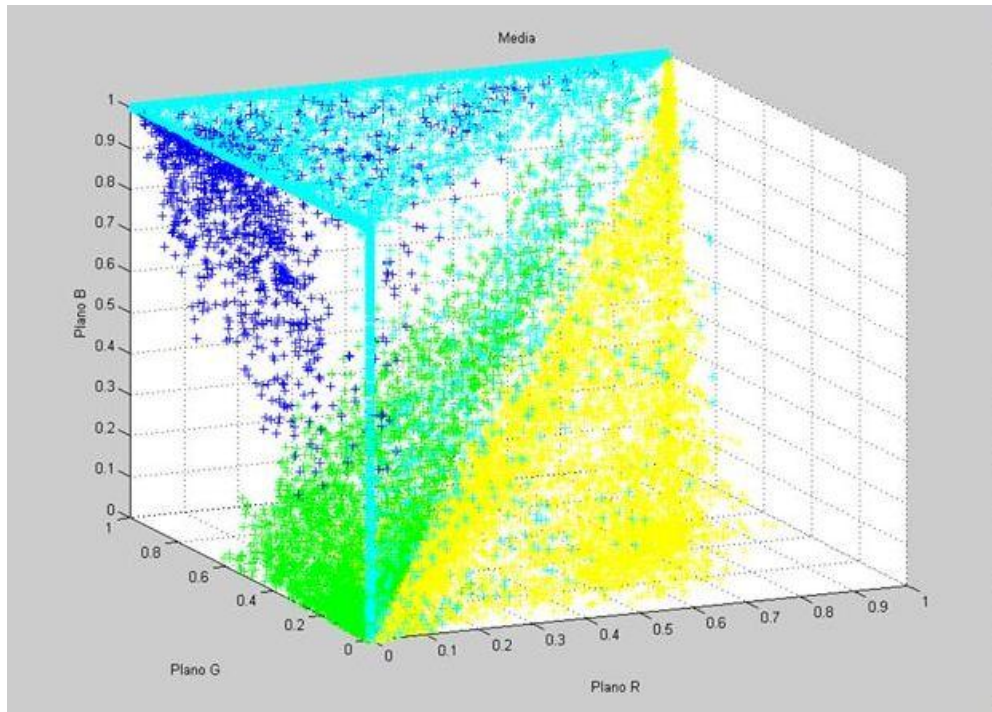


Ilustración 15. Media

La ilustración 16 pertenece a los datos graficados de la desviación estándar, donde podemos ver como los datos de cielo (cyan) se concentran en la parte de debajo de la gráfica y los demás se distribuyen en conjuntos de datos en los cuales es fácil identificar cada clase.

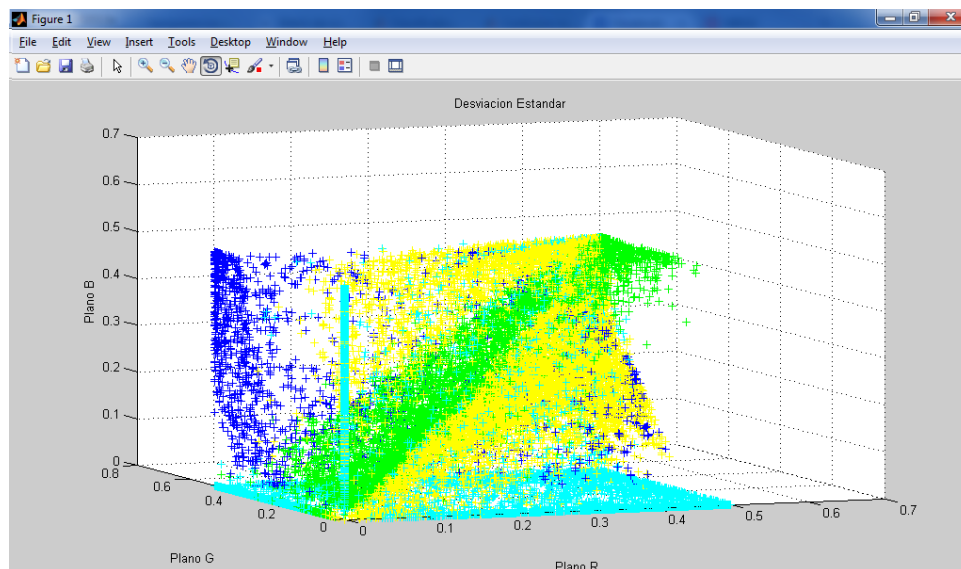


Ilustración 16 Desviación estándar

Por último se muestra la gráfica de la entropía (Ilustración 17) donde se puede ver que las clases cielo y agua se separan un poco mejor de las otras dos clases, pero se sigue viendo como unos datos se mezclan en la parte central del plano.

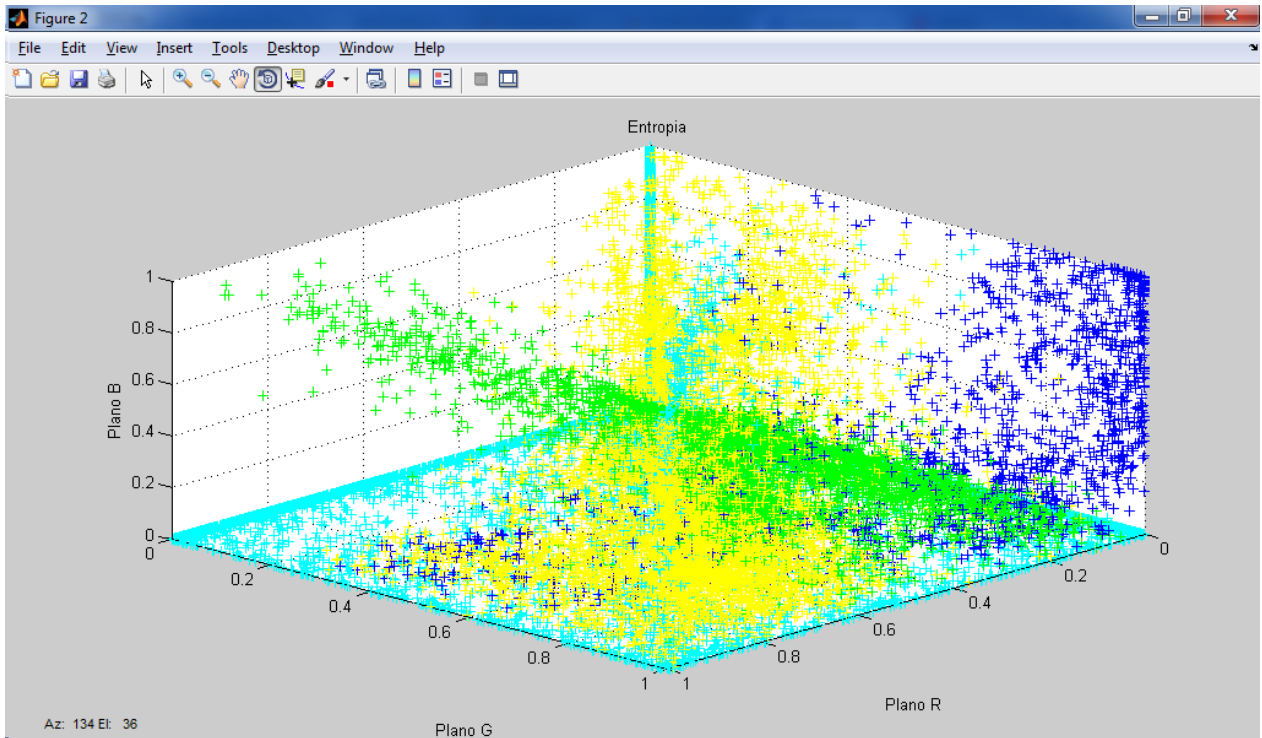


Ilustración 17 Entropía

Las características anteriores en conjunto son las que se utilizarán en el vector característico para identificar los 4 elementos de la naturaleza que se han propuesto en este trabajo.

6.2.2 Pruebas con el modelo HSV

Aquí se incluyen las pruebas realizadas con la media y la desviación estándar en el modelo HSV, los colores de las clases para estas pruebas se conservaron a excepción del color del cielo, que en este caso es rojo.

En la Ilustración 18 podemos observar el comportamiento de los datos en las cuatro clases en el modelo HSV utilizando la media.

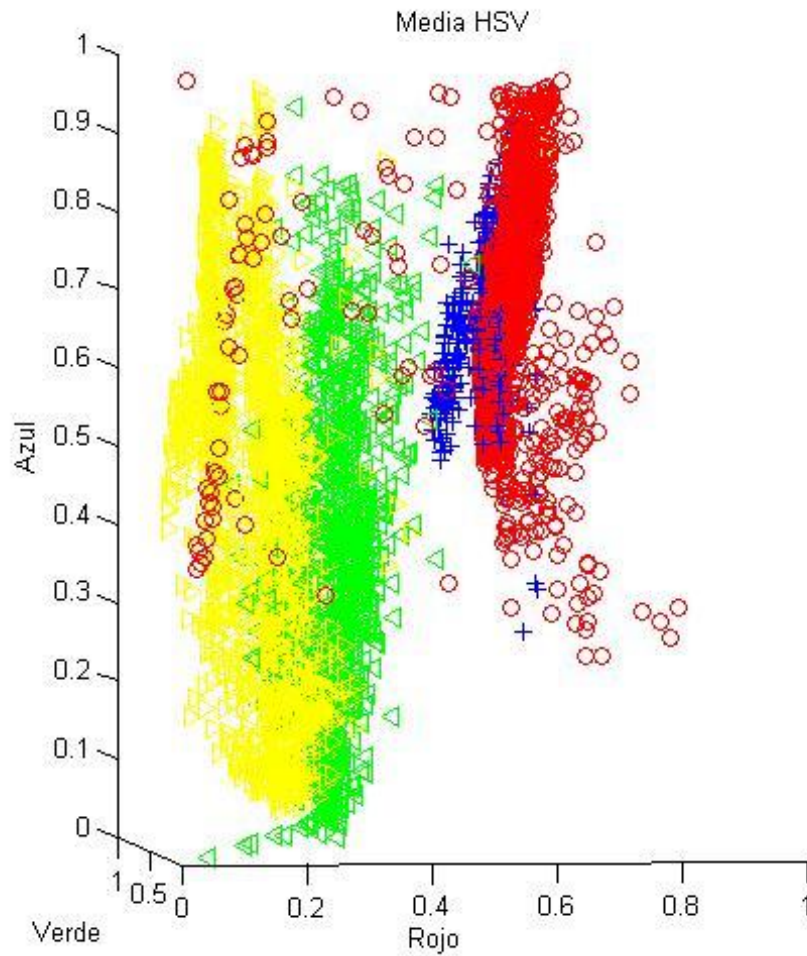


Ilustración 18 Media en el modelo HSV

La ilustración 19 muestra como se distribuyen las clases mediante el modelo HSV utilizando la desviación estándar. Se puede observar que los datos se concentran en una misma parte de la grafica, con lo cual podemos concluir que este modelo de color no es bueno para separar nuestras clases.

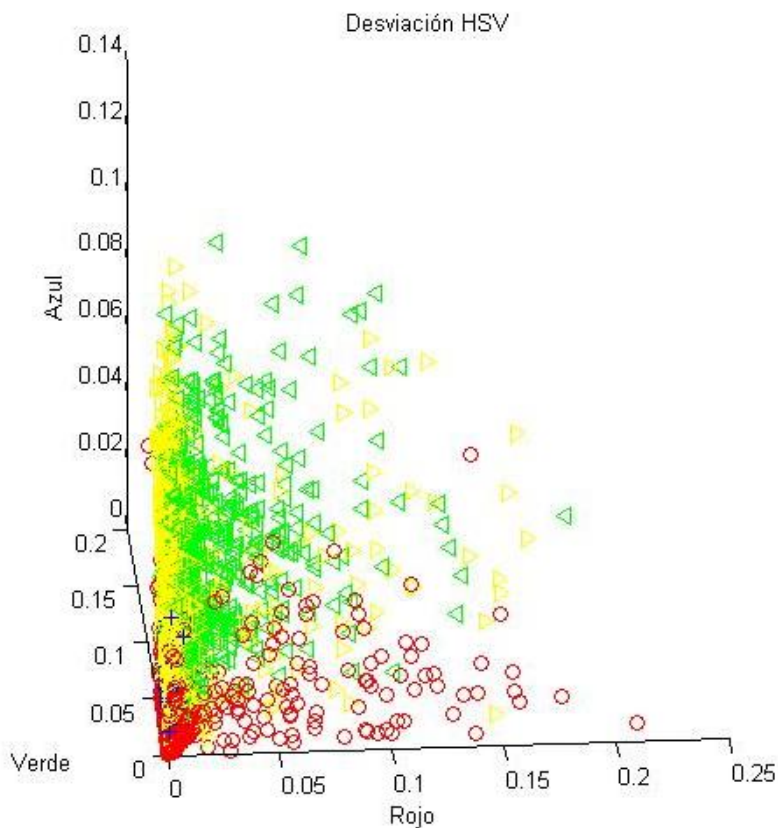


Ilustración 19 Desviación estándar en el modelo HSV

6.3 Pruebas de los descriptores de textura en niveles de gris

Esta sección está dedicada a observar el comportamiento de los descriptores de textura que han sido aplicados a las imágenes de la base de imágenes del proyecto.

6.3.1 Pruebas GLCM

Las pruebas de GLCM, se han graficado haciendo una combinación entre las cinco características obtenidas de este método, de las cuales mostramos solo las tres más representativas.

La Ilustración 20 muestra el comportamiento de la homogeneidad, contraste y correlación obtenidos de cada uno de los parches extraídos de las imágenes, los colores son idénticos que en las gráficas anteriores, teniendo al color cyan como representante del cielo, el color azul del agua, el verde de vegetación abundante y el amarillo de vegetación escasa (los colores serán los mismos para cada clase en las siguientes gráficas).

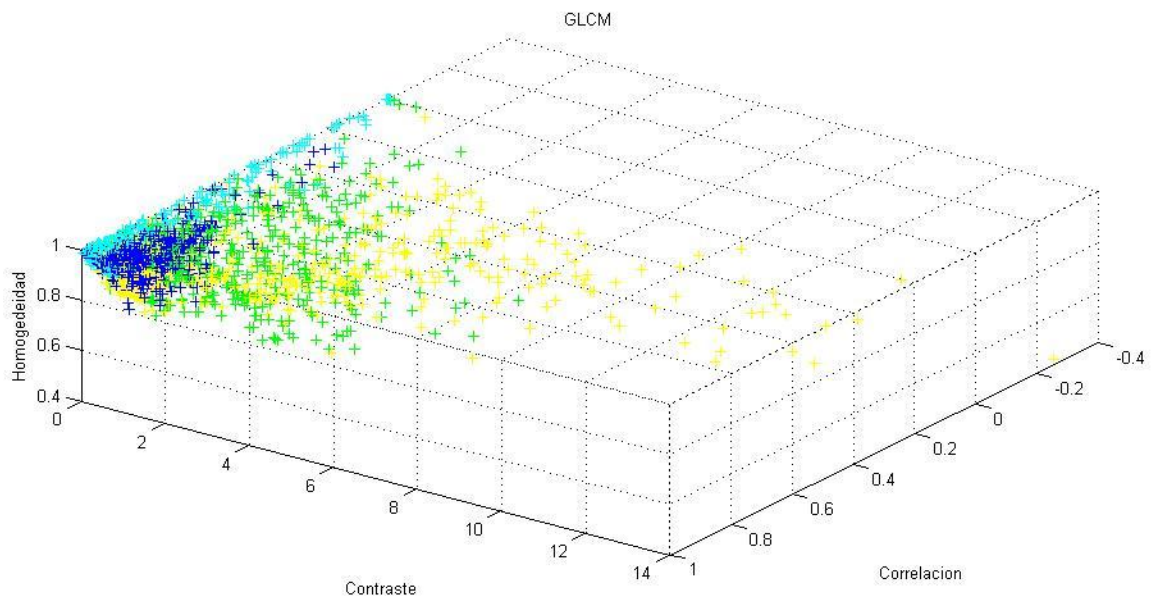


Ilustración 20 GLCM, Homogeneidad, Contraste y Correlación

En la ilustración 21 se grafican las características homogeneidad, contraste y energía, se puede ver que estas características, no separan las clases, por el contrario las concentra en un área de la gráfica.

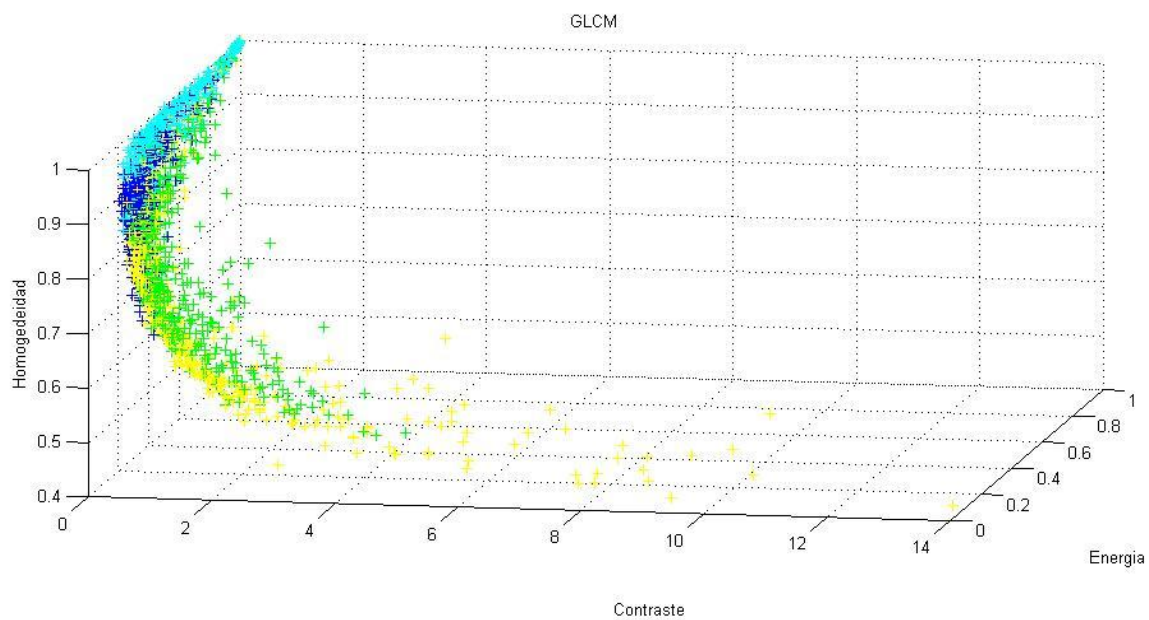


Ilustración 21 GLCM, Homogeneidad, Contraste y energía

En la Ilustración 22 (GLCM, Energía, Correlación y Contraste), se puede identificar un mejor separación entre las clases, pero aún están concentradas en una mismo zona, además se mezclan demasiado el cielo con el agua y la vegetación abundante con la vegetación escasa.

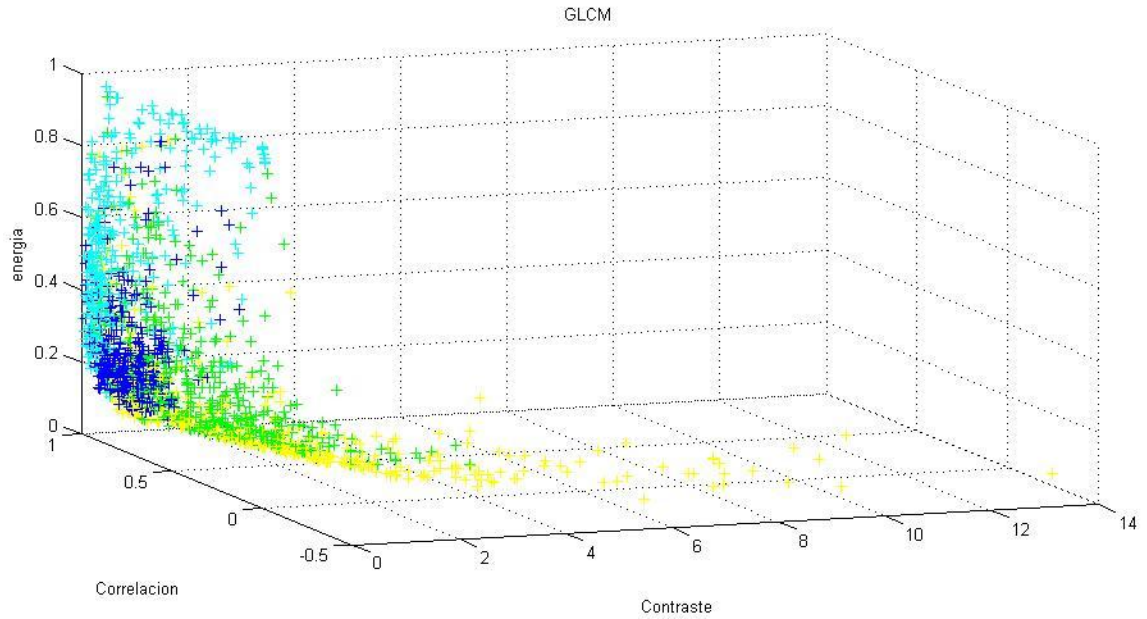


Ilustración 22 GLCM, Energía, Correlación y Contraste

6.3.2 Pruebas GLDM

Las pruebas del algoritmo GLDM se han realizado combinando en cada gráfica tres de las características que se obtuvieron de este, con lo cual hemos generado diez gráficas en las que analizamos el comportamiento de estas medidas de textura, para de esta manera decidir cuáles de ellas usaremos. A continuación se muestran las cuatro gráficas más representativas y la que se utilizará en el vector característico.

La ilustración 23 muestra una combinación de la asimetría, media y desviación obtenidas de la matriz GLDM, podemos observar que la media, separa las clases y en la asimetría se traslapan.

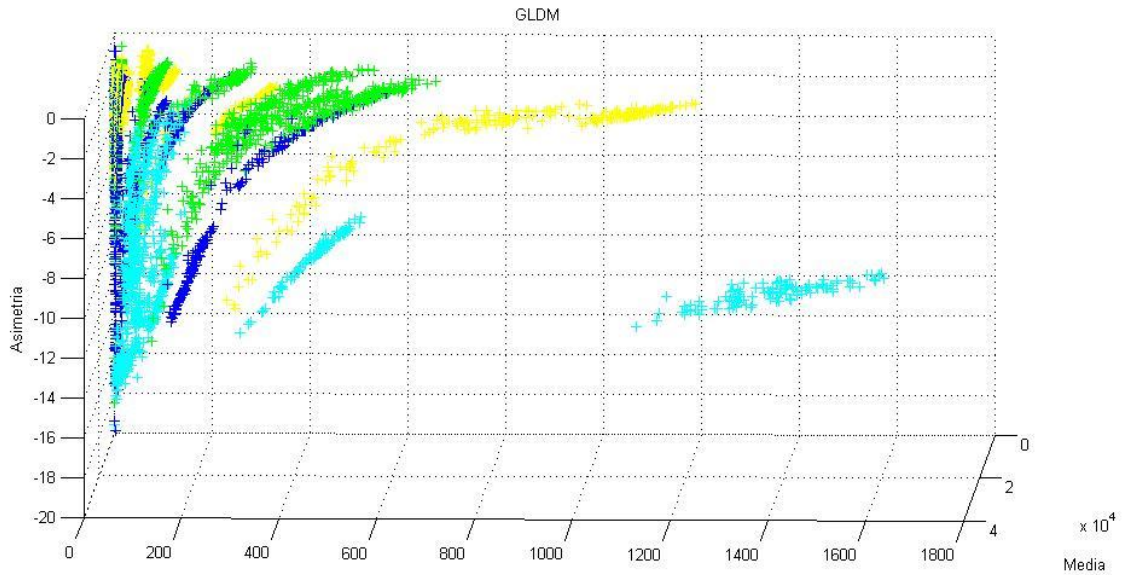


Ilustración 23 GLDM, Asimetría, Media, Desviación

La ilustración 24 nos muestra la curtosis, media y desviación formando una distribución de las clases bastante homogénea respecto al plano de la curtosis, por lo cual está no parece ser una buena característica para la clasificación.

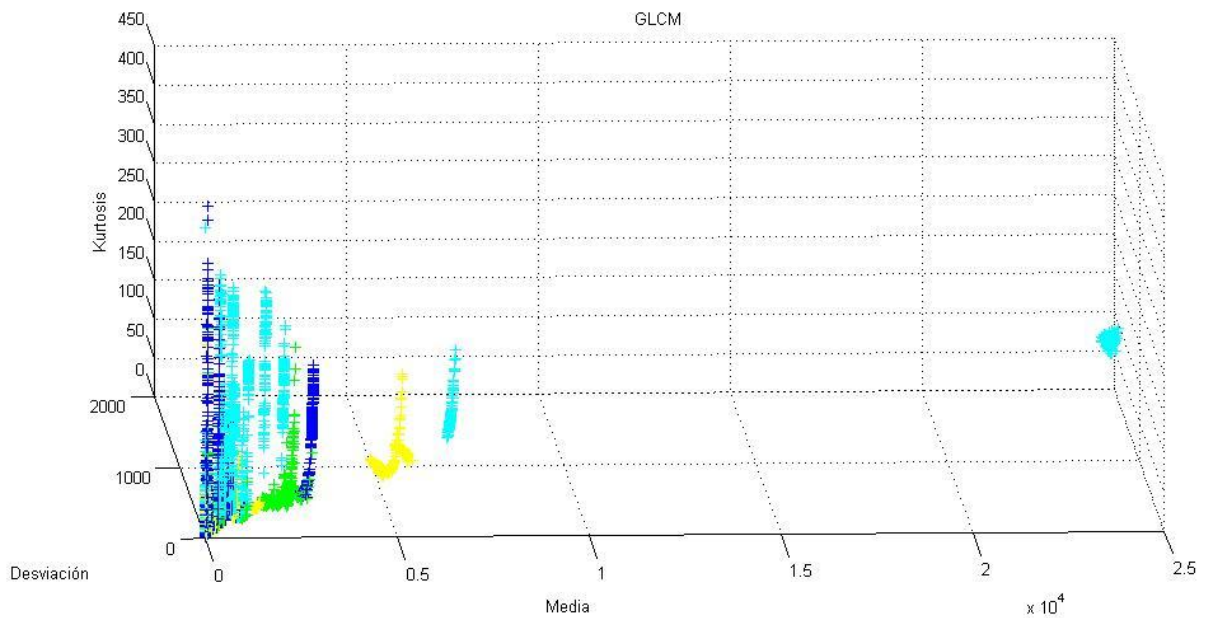


Ilustración 24 GLDM, curtosis, Desviación, Media

La prueba realizada con la asimetría, entropía y desviación (Ilustración 25) no muestra resultados favorables, pues mezcla las clases en un mismo punto y evita su fácil separación.

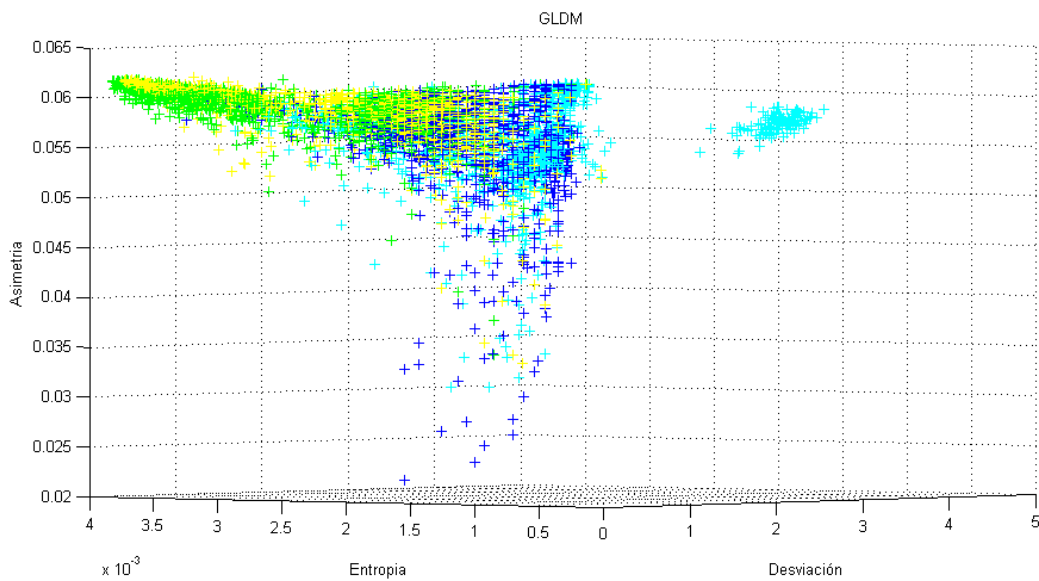


Ilustración 25 GLDM, Asimetría, Entropía y Desviación

La combinación de entropía, desviación y curtosis, mostrada en la Ilustración 26 muestra la separación de las clases en dos conjuntos principales, agua y cielo de un lado y vegetación abundante y vegetación escasa del otro lado, juntándose estos dos grupos en el centro, esto no se es bueno ya que estas características siguen traslapando unas clases con otras.

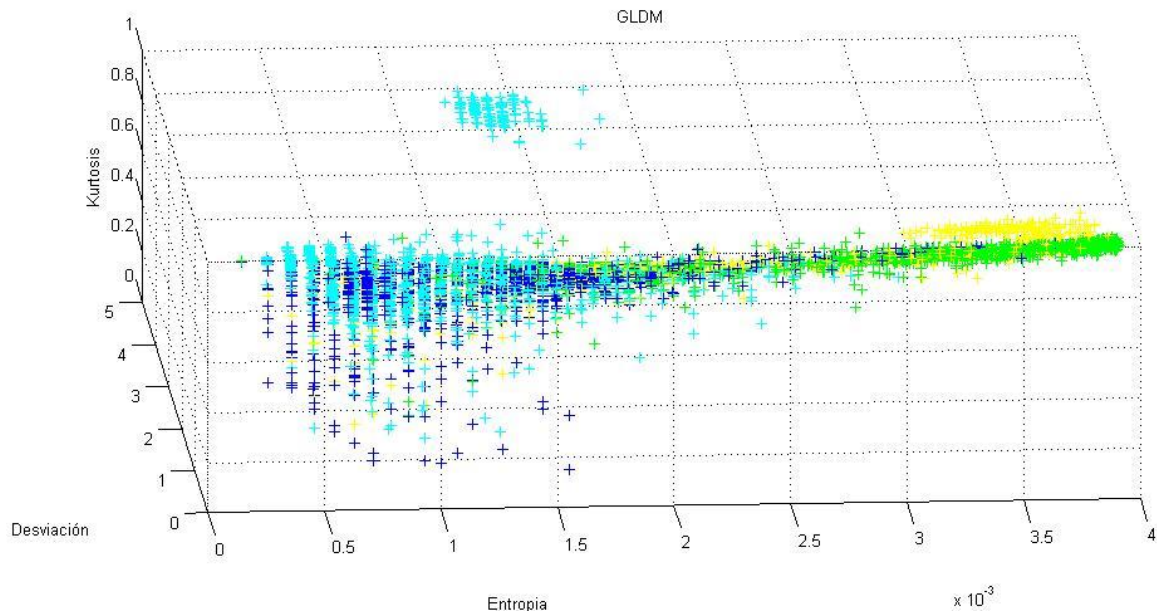


Ilustración 26 GLDM Kurtosis, Desviación y Entropía

Por último mostramos la gráfica de las dos características que hemos elegido como parte de nuestro vector característico. La ilustración 27 nos muestra la desviación, estándar y la media calculada de la matriz GLDM, y que como se puede observar separan mejor las clases que cualquiera de las vistas anteriormente.

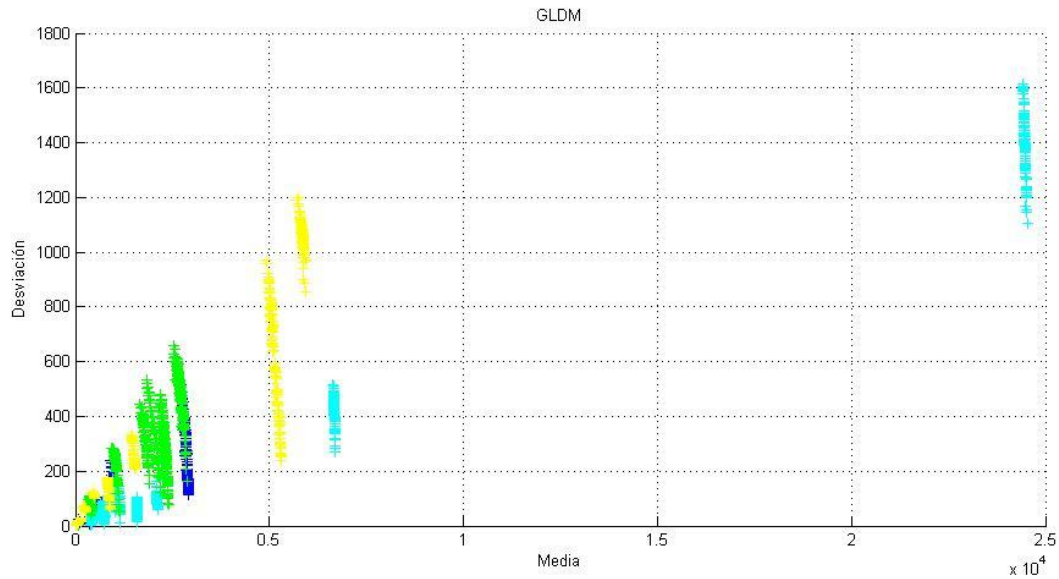
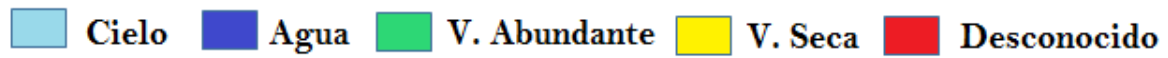


Ilustración 27 GLDM, Media y Desviación

6.4 Pruebas con red de base radial y Feedforward

Se obtuvo un conjunto de frames de diferentes videos de la base de datos de Mexculture en formato MP4 para evaluar la clasificación de cada red las cuales se muestran a continuación:

 Cielo Agua V. Abundante V. Seca Desconocido

FeedForward



Ilustración 29

Base Radial

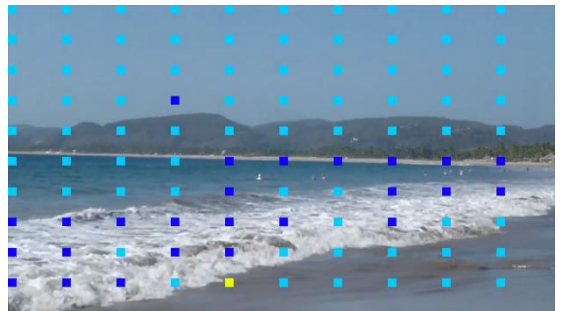


Ilustración 28



Ilustración 31



Ilustración 30

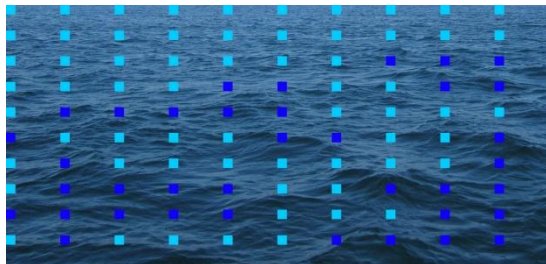


Ilustración 32

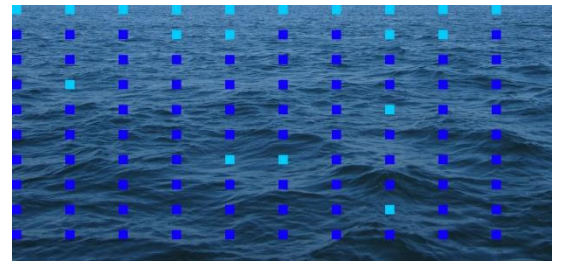


Ilustración 33

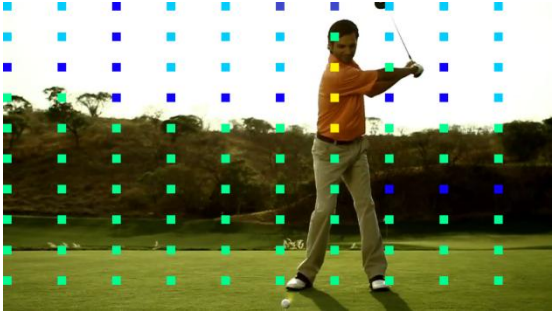


Ilustración 34

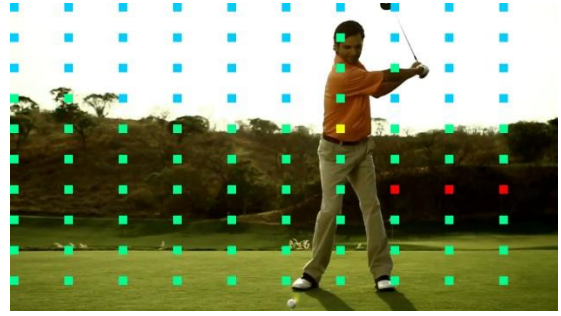


Ilustración 35

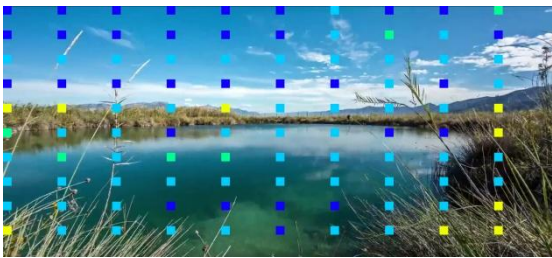


Ilustración 36

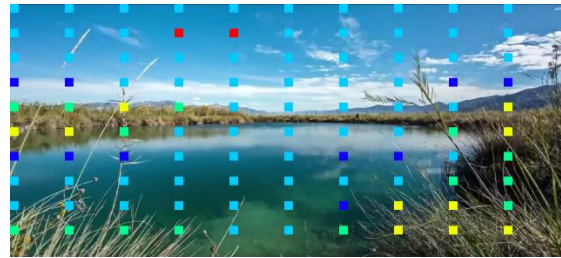


Ilustración 37

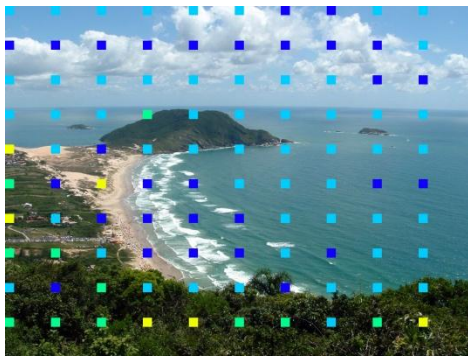


Ilustración 38

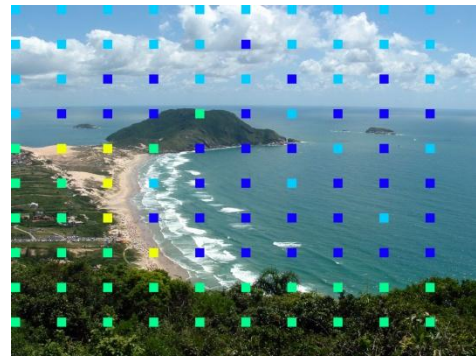


Ilustración 39

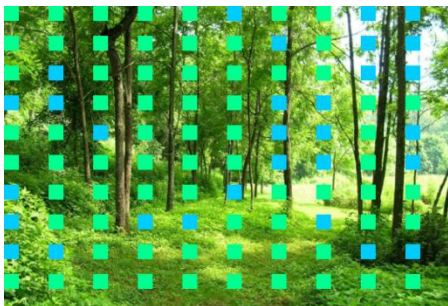


Ilustración 40

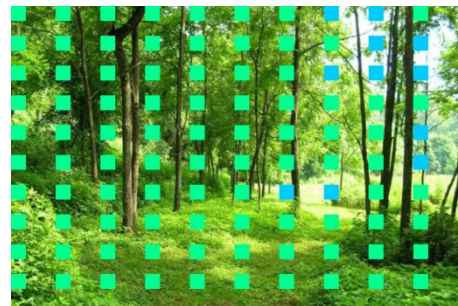


Ilustración 41

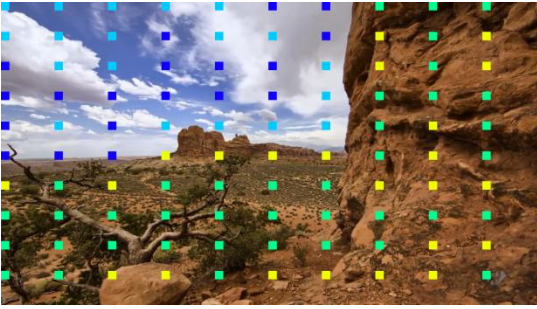


Ilustración 42

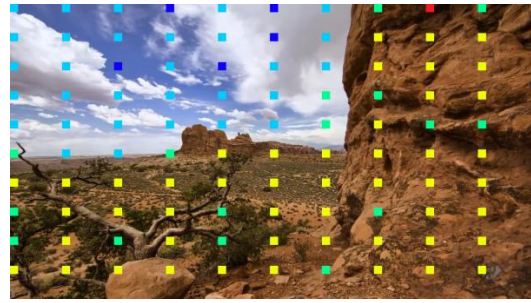


Ilustración 43

6.5 Matriz de confusión

Se probaron 650 imágenes para ver la confusión que genera cada red probada en este trabajo terminal, a continuación se presentan la matriz de confusión para cada una de las redes neuronales.

Para la red neuronal FeedForward da como resultado la siguiente matriz:

FeedForward				
	Agua	Cielo	Abundante	Seca
Agua	10	35	9	19
Cielo	56	106	1	0
Abundante	2	3	9	21
Seca	3	29	16	10
Desconocido	16	1	3	1

Tabla 16. Matriz de confusión para la red feedforward

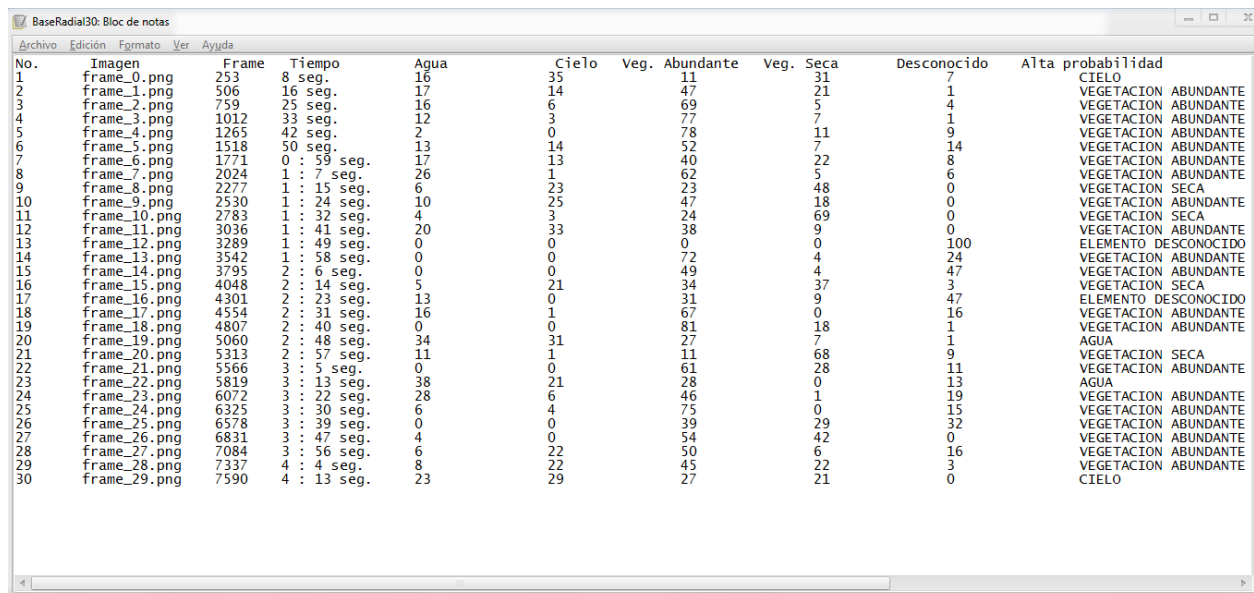
Se creó la matriz de confusión para la red de base radial la cual es la siguiente:

Base Radial					
	Agua	Cielo	Abundante	Seca	Desconocido
Agua	29	28	4	0	0
Cielo	9	131	1	0	12
Abundante	0	0	28	0	0
Seca	0	2	3	35	0
Desconocido	2	8	5	1	2

Tabla 17. Matriz de confusión para la red de base radial

6.6 Estructura del archivo del resultado final

La salida del sistema nos dará como resultado el archivo mostrado en la siguiente imagen en el cual se muestran los porcentajes de las imágenes analizadas así como el nombre del frame, número de frame y el tiempo en el cual fue encontrado.



No.	Imagen	Frame	Tiempo	Agua	Cielo	Veg. Abundante	Veg. Seca	Desconocido	Alta probabilidad
1	frame_0.png	253	8 seg.	16	35	11	31	7	CIELO
2	frame_1.png	506	16 seg.	17	14	47	21	1	VEGETACION ABUNDANTE
3	frame_2.png	759	25 seg.	16	6	69	5	4	VEGETACION ABUNDANTE
4	frame_3.png	1012	33 seg.	12	3	77	7	1	VEGETACION ABUNDANTE
5	frame_4.png	1265	42 seg.	2	0	78	11	9	VEGETACION ABUNDANTE
6	frame_5.png	1518	50 seg.	13	14	52	7	14	VEGETACION ABUNDANTE
7	frame_6.png	1771	0 : 59 seg.	17	13	40	22	8	VEGETACION ABUNDANTE
8	frame_7.png	2024	1 : 7 seg.	26	1	62	5	6	VEGETACION ABUNDANTE
9	frame_8.png	2277	1 : 15 seg.	6	23	23	48	0	VEGETACION SECA
10	frame_9.png	2530	1 : 24 seg.	10	25	47	18	0	VEGETACION ABUNDANTE
11	frame_10.png	2783	1 : 32 seg.	4	3	24	69	0	VEGETACION SECA
12	frame_11.png	3036	1 : 41 seg.	20	33	38	9	0	VEGETACION ABUNDANTE
13	frame_12.png	3289	1 : 49 seg.	0	0	0	0	100	ELEMENTO DESCONOCIDO
14	frame_13.png	3542	1 : 58 seg.	0	0	72	4	24	VEGETACION ABUNDANTE
15	frame_14.png	3795	2 : 6 seg.	0	0	49	4	47	VEGETACION ABUNDANTE
16	frame_15.png	4048	2 : 14 seg.	5	21	34	37	3	VEGETACION SECA
17	frame_16.png	4301	2 : 23 seg.	13	0	31	9	47	ELEMENTO DESCONOCIDO
18	frame_17.png	4554	2 : 31 seg.	16	1	67	0	16	VEGETACION ABUNDANTE
19	frame_18.png	4807	2 : 40 seg.	0	0	81	18	1	VEGETACION ABUNDANTE
20	frame_19.png	5060	2 : 48 seg.	34	31	27	7	1	AGUA
21	frame_20.png	5313	2 : 57 seg.	11	1	11	68	9	VEGETACION SECA
22	frame_21.png	5566	3 : 5 seg.	0	0	61	28	11	VEGETACION ABUNDANTE
23	frame_22.png	5819	3 : 13 seg.	38	21	28	0	13	AGUA
24	frame_23.png	6072	3 : 22 seg.	28	6	46	1	19	VEGETACION ABUNDANTE
25	frame_24.png	6325	3 : 30 seg.	6	4	75	0	15	VEGETACION ABUNDANTE
26	frame_25.png	6578	3 : 39 seg.	0	0	39	29	32	VEGETACION ABUNDANTE
27	frame_26.png	6831	3 : 47 seg.	4	0	54	42	0	VEGETACION ABUNDANTE
28	frame_27.png	7084	3 : 56 seg.	6	22	50	6	16	VEGETACION ABUNDANTE
29	frame_28.png	7337	4 : 4 seg.	8	22	45	22	3	VEGETACION ABUNDANTE
30	frame_29.png	7590	4 : 13 seg.	23	29	27	21	0	CIELO

7 Conclusiones

En el proyecto AVIN se implementó una serie de descriptores de rasgos, para poder obtener el mejor vector característico que pudiera representar cada una de las clases a reconocer, es importante mencionar, que la serie de descriptores existentes para este tipo de clases es muy grande, pero a pesar de ello, no existe aún un conjunto de estos descriptores que permitan de manera adecuada la separación absoluta de las clases y con ello su posterior clasificación. Un caso similar es el del clasificador, del cual solo se probaron dos tipos, aunque también existen muchos más como redes de Hop Field, memorias asociativas, algoritmos genéticos, clasificadores basados en métricas, etc... Al final se ha optado por los que a nuestra consideración son los más adecuados, tanto en las características como en los clasificadores, todo esto de acuerdo a las pruebas que se realizaron de cada uno de ellos.

Por otro lado, se modificaron algunas cosas del estudio de económico y del diseño del sistema, que no se habían considerado y que sobre la marcha se han desarrollado.

El proyecto AVIN llega hasta este punto a una fase de madurez en la cual se han probado y seleccionado los algoritmos adecuados para obtener el mayor porcentaje de clasificación correcta y se cuenta con una interfaz gráfica de usuario, fácil e intuitiva para poder manejar el sistema.

8 Trabajo a futuro

Como trabajo a futuro el primer paso es la identificación de escenas naturales, donde se pueda obtener como resultado el nombre de un ecosistema (playa, bosque, desierto, etc.).

Otro punto importante como trabajo a futuro es el reconocimiento de fauna en estos ecosistemas, pues muchas veces encontraremos esta fauna en las imágenes de escenas naturales.

Por último, pero no menos importante es el reconocer estas escenas naturales en distintas horas del día, pues no sería igual encontrar una imagen de desierto en el día, que en la noche y esto presenta un gran reto, para el reconocimiento de estas escenas.

Referencias

- [1] Ying Liu; Xiaofang Zhou, "Automatic texture segmentation for texture-based image retrieval," *Multimedia Modelling Conference, 2004. Proceedings. 10th International*, vol., no., pp.285,290, 5-7 Jan. 2004.
- [2] Geetha, M.K.; Palanivel, S., "HMM Based Automatic Video Classification Using Static and Dynamic Features," *Conference on Computational Intelligence and Multimedia Applications, 2007. International Conference on* , vol.3, no., pp.277,281, 13-15 Dec. 2007.
- [3] Santana, P.; Mendonca, R.; Barata, J., "Water detection with segmentation guided dynamic texture recognition," *Robotics and Biomimetics (ROBIO), 2012 IEEE International Conference on* , vol., no., pp.1836,1841, 11-14 Dec. 2012.
- [4] C. Schmid. Weakly supervised learning of visual models and its application to content-based image retrieval. "International Journal of Computer Vision", 56, no. 12:7-16, 2004.
- [5] J. Vogel. "Semantic Scene Modeling and Retrieval PhD Thesis". PhD thesis, Swiss Federal Institute of technology Zurich Germany, 2004.
- [6] J. Vogel and B. Schiele. "Semantic modeling of natural scenes for content-based image retrieval". *International Journal of Computer Vision*, 72(2), pages 133-157, 2007.
- [7] Solmaz, B.; Modiri, S.; Shah, M; "Classifying Web Videos using a Global Video Descriptor", *Machine Vision and Applications MVAP-D-12-00244*, 2012.
- [8] Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local svm approach. In: *Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04)*. Volume 3. (2004) 32–36.
- [9] Center for Research in Computer Vision (2012, Sep. 12) "UFC50-Action Recognition Data Set", University of Central Florida [Online]. Disponible en: <http://crcv.ucf.edu/data/UCF50.php> [Consultado: Feb. 20, 2015].
- [10] Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: HMDB: a large video database for human motion recognition. In: *ICCV*. (2011).
- [11] Sasidharan, R.; Menaka, D., "Dynamic texture segmentation of video using texture descriptors and optical flow of pixels for automating monitoring in different environments," *Communications and Signal Processing (ICCSP), 2013 International Conference on* , vol., no., pp.841,846, 3-5 April 2013 doi: 10.1109/iccsp.2013.6577175.
- [12] Mokri, S.S.; Ibrahim, N.; Hussain, A.; Mustafa, M.M, Motion detection using Horn Schunck algorithm and implementation, Vol. 01, pp. 83-87, 2009.
- [13] T. Ojala, M. Pietikäinen and T. Mäenpää. Multiresolution Gray Scale and Rotation Invariant Texture Analysis with Local Binary Patterns, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24(7), pp. 971- 987, 2002.
- [14] O. Valderrama, J.; Información tecnológica, Vol. (09), pp. 321-322, 1998.
- [15] Serrano J., "Recuperación de imágenes mediante rasgos descriptores globales y locales", Centro de Investigación en Computación-IPN, 2011.
- [16] Ministerio de educación, "La imagen digital", Instituto de Tecnologías educativas[Online]. Disponible en http://www.ite.educacion.es/formacion/materiales/86/cd/pdf/m2_caracteristicas_de_la_imagen_digital.pdf [Consultado: Abril 20, 2015]
- [17] Gonzalez R., Woods R., "Digital image processing", First Ed. Pp. 244-252, 1992.
- [18] Pajares G., de la Cruz J., "Visión por Computador, Imágenes digitales y aplicaciones", Primera Ed., Pp. 128-137, 2002.
- [19] Serna y Alvarado (2011, May.) "Descriptores de Imágenes", Recuperación de Información Multimodal [Online]. Disponible en

<http://recuperaciondeinformacionmultimodal.blogspot.mx/2011/05/descriptores-de-imagenes.html> [Consultado: Abril 20, 2015].

- [20] Walpole, Ronald E., Probabilidad y estadística para ingenieros, 6ª ed., Prentice Hall Hispanoamericana, México, Pp. 4-6, 1999.
- [21] Pajares G., de la Cruz J., “Visión por Computador, Imágenes digitales y aplicaciones”, Primera Ed., Pp. 257-260, 2002.
- [22] L. Mantari, Medidas de Forma o Distribución, Universidad Peruana los Andes, 20 septiembre 2014, pp 3.
- [23] Camara A., “Método contextual para fusión de datos”, Programa de Maestría y Doctorado en ingeniería-UNAM, 2005.
- [24] Miriam Presutti, “La matriz de co-ocurrencia en la clasificación multiespectral :tutorial para la enseñanza de medidas texturales en cursos de grado universitario” 4ª Jornada de Educação em Sensoriamento Remoto no Âmbito do Mercosul – 11 a 13 de agosto de 2004 – São Leopoldo, RS, Brasil.
- [25] Batchelor Bruce; Waltz Frederick, “Intelligent Machine Vision, Techniques, Implementations and Applications”, Ed. Springer-Verlag London, 1st ed., pp 60-70.
- [26] R. P. Gabriel; V. Rodríguez; R. Cruz, “Redes Bayesianas para la clasificación de masas en mamografías” ,Universidad tecnológica de la mixteca, Temas de Ciencia y Tecnología vol. 17 número 51, pp 11 - 24, septiembre - diciembre 2013.
- [27] Universidad del Valparaíso, “Redes Neuronales Artificiales” [Online]. Disponible en: http://www.inf.utfsm.cl/~rsalas/Pagina_Investigacion/docs/Apuntes/Redes%20Neuronales%20Artificiales.pdf [Consultado May. 20, 2015].
- [28] H. Allende, C. Moraga y R. Salas (2002). “Artificial Neural Networks in Time Series Forecasting: A Comparative Analysis”. *Kybernetika*, Volume 38, number 6, pages 685-707. ISIISSN: 0023-5954.
- [29] González, J.R.H., Hernando, V.J.M., “Redes neuronales artificiales: fundamentos, modelos y aplicaciones”, Ed.RA-MA, 1994.
- [30] Strumillo, P., Kamiński, W., *Neural Networks and Soft Computing*, Ed. Physica-Verlag HD, vol. 19, pp. 108.
- [31] MacQueen, J. *Some methods for classification and analysis of multivariate observations. Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, Volume 1, Universidad de California Press, Berkeley, Calif., 1967.
- [32] BIODIVERSIDAD MEXICANA. "Ecosistemas de México", biodiversidad.gob.mx. [Online]. Disponible en: <http://www.biodiversidad.gob.mx/ecosistemas/ecosismex.html>. [Consultado: May. 25, 2015].
- [33] Boehm, B., “Using the WINWIN Spiral Model: A Case Study”, en *Computer*, vol. 3, núm.7, julio de 1988, pp. 33-34.
- [34] R. Pressman, *Ingeniería del Software, un enfoque práctico*, Ed. Mac Graw Hill, 6th ed., pp. 58
- [35] I. Sommerville, *Ingeniería del Software*, Ed. Pearson Addison Wesley, 7th ed., pp. 68-69
- [36] Boehm, B, "*Spiral Development: Experience, Principle ,and Refinements*", Special Report CMU/SEI-2000-SR-008, July 2000.

[37] R. Pressman, *Ingeniería de Software, un enfoque práctico*, Ed. Mac Graw Hill, 6ta ed.