

DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

**“Implementación de una red Neuronal Artificial
para Autonomía de Sistemas de Monitoreo y
Seguridad”**

POR

CEDRIC OMAR HERNÁNDEZ RIESCO

TESIS

**PRESENTADA COMO REQUISITO PARCIAL PARA OBTENER EL
GRADO DE MAESTRO EN CIENCIAS EN INGENIERÍA ELÉCTRICA**

DIRECTOR DE TESIS

DR. FRANCISCO GERARDO FLORES GARCÍA

ISSN: 0188-9060

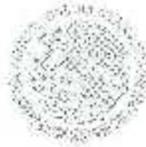


RIITEC: (05)-TMCIE-2015

Torreón, Coahuila. México
Agosto 2015

SEP

SECRETARÍA DE
EDUCACIÓN PÚBLICA



TECNOLÓGICO NACIONAL DE MÉXICO
Instituto Tecnológico de La Laguna

"2015, Año del Generalísimo José María Morelos y Pavón"

Torreón, Coah., 25/Agosto/2015

Dependencia: DEPI/CPCIE

Oficio: DEPI/CPCIE/158/2015

Asunto: Autorización de impresión
de tesis.

C. CEDRIC OMAR HERNÁNDEZ RIESCO
CANDIDATO AL GRADO DE MAESTRO EN CIENCIAS EN INGENIERÍA ELÉCTRICA.
PRESENTE

Después de haber sometido a revisión su trabajo de tesis titulado:

**"IMPLEMENTACIÓN DE UNA RED NEURONAL ARTIFICIAL
PARA AUTONOMÍA DE SISTEMAS DE MONITOREO Y SEGURIDAD"**

Habiendo cumplido con todas las indicaciones que el juraco revisor de tesis hizo, se le comunica que se le concede la autorización con número de registro **RIITEC: (08)-TMCIE-2015**, para que proceda a la impresión del mismo.

A T E N T A M E N T E
EDUCACIÓN TECNOLÓGICA FUENTE DE INNOVACIÓN

DR. JOSÉ LUIS MEZA MEDINA
Jefe de la División de Estudios de Posgrado e Investigación -
del Instituto Tecnológico de La Laguna

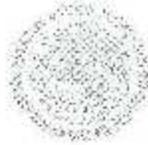
SECRETARÍA DE
EDUCACIÓN PÚBLICA
INSTITUTO TECNOLÓGICO
de La Laguna
Blvd. Revolución y Casa Cuernavaca s/n Col. Centro CP. 27000
Torreón, Coah. Tel. 7051313, e-mail: sepi@itl.laguna.edu.mx

JLMM/HR



SEP

SECRETARÍA DE
EDUCACIÓN PÚBLICA



TECNOLÓGICO NACIONAL DE MÉXICO
Instituto Tecnológico de La Laguna

"2015, Año del Generalísimo José María Morelos y Pavón"

Torreón, Coah., 24/Agosto/2015

DR. JOSE LUIS MEZA MEDINA
JEFE DE LA DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

Por medio de la presente, hacemos de su conocimiento que después de haber sometido a revisión el trabajo de tesis titulado:

"Implementación de una red neuronal artificial para autonomía de sistemas de monitoreo y seguridad"

Desarrollado por el **C. Cedric Omar Hernandez Riesco**, con número de control **M1313072** y habiendo cumplido con todas las correcciones que se le indicaron, estamos de acuerdo que se le conceda la autorización de la fecha de examen de grado para que proceda a la impresión de la misma.

ATENTAMENTE
EDUCACIÓN TECNOLÓGICA FUENTE DE INNOVACION



Dr. Francisco Flores Garcia
Asesor/Director de Tesis



M.C. Martín Vázquez Rueda
Coasesor de Tesis



Dr. Mario F. Cepeda Rubio
Comité Tutorial



Dr. Juan Silvestre Mijares
Comité Tutorial



Dedicatoria

A mis padres, Salvador Hernández y Magdalena Riesco, y a quienes creyeron en mí.

Agradecimientos

A mi asesor y tutor Dr. Francisco Gerardo Flores García por el apoyo y orientación que me ofreció, gracias a sus conocimientos y enseñanzas ha sido posible la realización de este proyecto.

A toda mi familia por haberme apoyado durante el transcurso de estos años.

Al Consejo Nacional de Ciencia y Tecnología por darme la oportunidad de continuar con mi preparación profesional y por el apoyo económico recibido.

A mis amigos que me ayudaron en los temas que no dominaba en esos momentos.

A mis compañeros de maestría por demostrarme su amistad y el apoyo durante nuestras clases juntos.

Resumen

Hoy en día hemos observado un incremento en el Índice delictivo en nuestra sociedad, por tal motivo hemos tenido la necesidad de cuidar nuestro patrimonio de alguna manera u otra. Por tal motivo las personas recurren a los sistemas de monitoreo y seguridad privada, pero para adquirir éste servicio se requiere una considerable inversión económica, que no todos logran solventar, ya que este tipo de servicios requieren una infraestructura para operar y personal que la maneje adecuadamente. En éste trabajo se desarrolla una alternativa a esos servicios, con un bajo costo y que esté al alcance de la mayoría.

En el actual proyecto se trató de implementar a estos sistemas de monitoreo y seguridad una red neuronal artificial llamada ART2 (*Adaptive Resonance Theory 2*); las redes neuronales artificiales son la emulación de las redes neuronales biológicas, que son las encargadas de procesar la información que recibe y envía el cerebro (hablando en la anatomía del ser humano).

La red neuronal artificial ART2 es una red no supervisada o auto-organizada, es decir, una red neuronal que aprende por sí sola. Esta red compara datos pasados con actuales y si encuentra similitud reconoce dichos datos, lo cual significa que, logra identificarlos y recordarlos. Si no logra encontrar una similitud con los patrones anteriores lo cataloga como uno nuevo y lo almacena en su memoria de largo plazo para que cuando vuelva a recibirlo, o uno parecido lo identifique.

Aunque la red ART2 sea aun teórica en éste proyecto se codificó en el lenguaje de programación Java por su versatilidad de multiplataforma (funciona en cualquier sistema operativo), y eso nos da la posibilidad de implementar la red en casi cualquier aparato electrónico, en éste caso utilizado para que por medio de cámaras la red pueda reconocer rostros y se convierta en el circuito de monitoreo, y el personal encargado de la seguridad.

Palabras clave: red neuronal artificial, ART2, Java.

Abstract

Today we have observed an increase in the crime rate in our society, for that reason we had the need to protect our heritage in some way or another. Thus people appeal to monitoring systems and private security, but to acquire this service is needed a considerable high economic investment, not all people are able to pay that is required. Since these services require an infrastructure to operate and to the specialist staff to operate it. In this work an alternative to these services is developed, with low cost and within the reach of most people.

In the current project has tried to implement to these security and monitoring systems, an artificial neural network called ART2 (Adaptive Resonance Theory 2); Artificial neural networks are the emulation of biological neural networks, which are responsible for processing the information it receives and sends the brain (speaking in human anatomy).

The artificial neural network ART2 is an unsupervised or self-organized network, i.e., a neural network that learns by itself. This network compare current data with past data and if it find similarity recognizes that data, i.e., it can identify and remember. If it cannot find a similarity with the previously observed data, is categorized as new data and stores it in your long term memory so that when it return to receive the same or similar data, identifies the data.

Although the network is still theoretical ART2 in this project was coded in the Java programming language for its versatility that is cross-platform (works on any operating system), and that gives us the possibility to implement the network in almost any electronic device, in this case was used in video cameras for it can recognize faces and practically becomes in the monitoring circuit, and the staff responsible for security.

Keywords: artificial neural network, ART2, Java.

Índice

Capítulo 1	1
1.1 Introducción.....	1
1.2 Objetivos.....	2
2 Capítulo 2	3
2.1 Marco Teórico.....	3
2.1.1 Sistemas de monitoreo y seguridad.....	3
2.1.2 Red Neuronal Biológica.....	4
2.1.3 Red Neuronal Artificial.....	5
2.1.4 Java.....	12
2.1.5 Procesamiento digital de una imagen.....	12
2.1.6 Antecedentes.....	17
2.1.7 Estado del arte.....	18
11 Capítulo 3	19
3.1 Desarrollo e implementación del proyecto.....	19
3.1.1 Etapa de diseño.....	19
3.1.2 Etapa de desarrollo.....	23
15 Capítulo 4	29
4.1 Pruebas y resultados.....	29
4.2 Conclusiones.....	43
18 Capítulo 5	45
5.1 Referencias Bibliográficas.....	45

Índices de Figuras

Figura 2.1.- Esquema de un circuito cerrado de vigilancia	3
Figura 2.2.- Ejemplo del trabajo del personal de vigilancia	5
Figura 2.3.- Partes de una neurona	5
Figura 2.4.- Comunicación entre neuronas	6
Figura 2.5.- Perceptrón simple	8
Figura 2.6.- Red neuronal Adaline (ADaptative Linear Element)	9
Figura 2.7.- Esquema del Perceptrón multicapa.....	10
Figura 2.8.- Esquema de la red de Memorias asociativas	11
Figura 2.9.- Red Backpropagation	11
Figura 2.10.- Demostración de la interpretación de la información de un mapa auto-organizado	12
Figura 2.11.- Estructura de imagen raster	14
Figura 2.12.- Curva de contraste	16
Figura 3.1.- Ejemplo del reconocimiento facial	19
Figura 3.2.- Estructura básica de una red ART2	21
Figura 3.3.- Diagrama de flujo de la red ART2	22
Figura 3.4.- Asistente para la creación del nuevo proyecto	24
Figura 3.5.- Diagrama de flujo condensado del proceso de la red ART2	27
Figura 4.1.- Distancia máxima y posición que debe tener el rostro para que la red pueda identificarlo	30
Figura 4.2.- Resultado a la exposición a poca luz aproximada a una candela	31
Figura 4.3.- Resultado a la exposición de la luz del medio día	31
Figura 4.4.- Pestaña de inicio del programa	32
Figura 4.5.- Pestaña de una de las cámaras	33
Figura 4.6.- Pestaña de la configuración del programa	34
Figura 4.7.- Modificación de la base de datos de manera manual	35
Figura 4.8.- Vista de la cámara con el nombre modificado	35
Figura 4.9.- Esquema de la red neuronal ART2 modificada	36
Figura 4.10.- Tiempos registrados por la red para reconocer rostros	37
Figura 4.11.- Rostros para el entrenamiento de la red a comparar.....	38
Figura 4.12.- Imágenes en escala de grises	38
Figura 4.13.- Imágenes en blanco y negro delimitadas por bordes	39
Figura 4.14.- Imagen tratada para la prueba de la red	39

Figura 4.15.- Resultado del entrenamiento de la red	40
Figura 4.16.- Respuesta dada por la red al introducir la imagen de prueba	41
Figura 4.17.- Entrenamiento de la red con 30 capas ocultas	41
Figura 4.18.- Respuesta de la red en la segunda corrida	42

Capítulo 1

1.1 Introducción

Hoy en día, los sistemas de monitoreo y seguridad son servicios de alta demanda, requeridos tanto en empresas como en zonas residenciales. Debido al incremento de la actividad delictiva en los últimos años.

Este tipo de servicios suelen ser costosos ^[1,2], ya que requieren una infraestructura y recurso humano especializados; debido a esto, no es sencillo sustentar la contratación y el mantenimiento de los mismos.

Por este motivo se plantea la incorporación de una red neuronal artificial a estos sistemas para disminuir el costo y que de esta manera sean más accesibles a toda clase de público.

Los costos de estos servicios se ven reflejados principalmente en el recurso humano, ya que es necesario el pago de salarios de manera periódica dicese semanal, quincenal o mensual. Porque queda a juicio de una persona decidir quién no está autorizado y quién sí al visualizar a alguien por medio de una cámara en un sistema de monitoreo. Así mismo como una persona no puede estar demasiado tiempo viendo monitores o haciendo trabajos repetitivos sin descanso, es necesario la rotación de personal; entonces al hacer esos movimientos no se puede asegurar un trabajo ciento por ciento eficaz.

De esta manera al implementar una red neuronal artificial se puede dispensar hasta cierto punto de la presencia del recurso humano en estos servicios. Porque una red neuronal tiene la suficiente capacidad para reconocer patrones, en este caso rostros, en un sistema de monitores y determinar si cierto objetivo es alguien autorizado o no.

Teniendo en cuenta lo que una red neuronal artificial puede lograr, se visualiza un ahorro en el costo de estos servicios y de esta forma que estén al alcance del público en general.

Para alcanzar este cometido se plantea implementar este sistema por medio del lenguaje de programación Java. El cual es un lenguaje multiplataforma, es decir, que se incorpora en cualquier sistema operativo, además de que millones de dispositivos lo usan como sistema principal para su funcionamiento.

Gracias a la flexibilidad que ofrece este lenguaje es posible desarrollar un sistema base para ser combinado con el diseño de una red neuronal artificial y resulte en un modelo para crear programas hasta cierto punto autónomos y sean empleados en los sistemas de monitoreo y seguridad.

1.2 Objetivos

Objetivo principal: Implementar la red neuronal artificial "Teoría de Resonancia Adaptativa 2" (o por sus siglas en inglés "Adaptive Resonance Theory 2" (ART2)) en sistemas de monitoreo y seguridad para convertirlos en sistemas inteligentes y que estén a la par con la tecnología que actualmente se produce mundialmente.

Objetivo secundario #1: Reducir los costos de estos sistemas a precios accesibles para toda la población en general.

Objetivo secundario #2: Reducir al mínimo la intervención humana en esta clase de sistemas para aumentar su eficacia correspondiente al tiempo de vigilancia e identificación de personal.

Objetivos secundario #3: Desarrollar una de las primeras formas codificadas en un lenguaje de programación de alto nivel de este tipo de red neuronal artificial.

Objetivo secundario #4: Comprobar la sinergia de tecnología económica con las capacidades que puede ofrecer este tipo de red neuronal artificial.

Capítulo 2

2.1 Marco Teórico

- 2.1.1 Sistemas de monitoreo y seguridad

Los sistemas de monitoreo y seguridad son servicios impartidos por empresas generalmente privadas que consisten de dos partes, los aparatos electrónicos y el recurso humano^[3]. Entre los aparatos electrónicos comúnmente encontramos cámaras de video, monitores para visualizar lo que la cámara capta, una consola donde se almacenan las fotos y los videos grabados por las cámaras, alarmas (pueden ser por tiempo, detectoras de movimiento, etc.) y cerrojos (en los cuales se encuentran seguros magnéticos y de pasador mecánico). Todo este equipo está conectado entre sí, y se le conoce como sistema de circuito cerrado de televisión o CCTV (véase Figura 2.1).

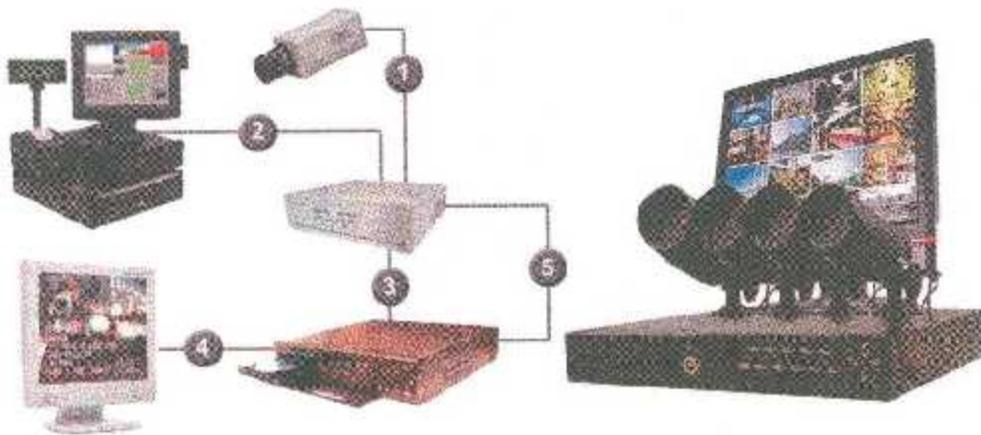


Figura 2.1.- Esquema de un circuito cerrado de vigilancia de televisión o CCTV.

Existen tres tipos de instalaciones para los sistemas de CCTV:

- La conexión analógica o convencional, es la instalación en la cual todos los componentes del CCTV están conectados por medio de cables y su única función es mostrar en monitores estáticos lo que las cámaras captan.

- La conexión basada en tecnología IP, esta instalación comunica a cada elemento del CCTV de manera inalámbrica a través de un servidor que puede ser local o remoto, esta instalación permite visualizar el audio y video que capturan las cámaras en cualquier equipo de cómputo que esté conectado en la misma red que los equipos.
- La conexión híbrida, esta instalación combina conexiones analógicas e inalámbricas, sirve para monitorear varias locaciones remotas y concentrar el flujo de información por medio de una base central.

Todas estas instalaciones pueden manejar diferentes tipos de cámaras según las necesidades del cliente, hay del tipo caja estática, con zoom, móviles y con posicionador.

Por otro lado tenemos el recurso humano el cual se encarga de manejar todo el equipo antes mencionado. Éste generalmente consta de un grupo de dos o tres personas a las cuales se les asigna el puesto de guardias, estas personas son las encargadas de estar viendo los monitores y por medio del equipo determinar que cerrojos se abren y se cierran, identificar a las personas grabadas por las cámaras y en su criterio queda quien está autorizado y quién no (véase Figura 2.2). Para lograrlo este recurso tiene que tener rotaciones para que las personas puedan descansar y su juicio no quede nublado por el agotamiento físico y mental, estas rotaciones son turnos que pueden ser tres o cuatro al día. Todo esto en conjunto forma el sistema de monitoreo y seguridad que su objetivo es mantener una locación vigilada y segura.

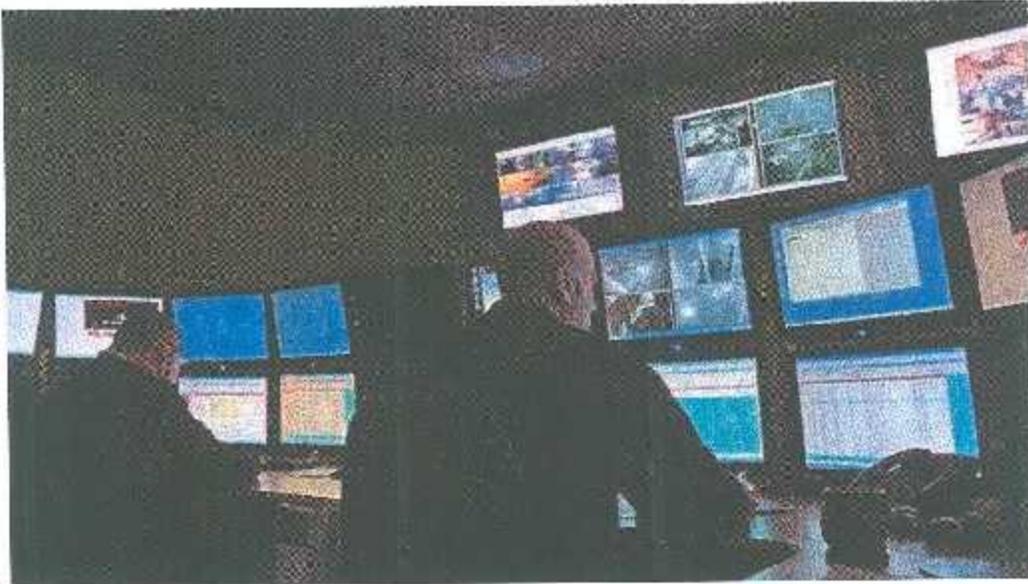


Figura 2.2.- Ejemplo del trabajo del personal de vigilancia.

- 2.1.2 Red Neuronal Biológica

La red neuronal biológica es un conjunto de neuronas distribuidas por todo el organismo de un ser vivo. La neurona es una célula, proyectándose desde el cuerpo celular, tiene miles de ramificaciones llamadas dendritas, además de tener una prolongación llamada axón que se divide a su vez en muchas ramificaciones^[4] llamado botón (véase Figura 2.3).



Figura 2.3.- Partes de una neurona.

El propósito de esta red es llevar información de un lado a otro del organismo en forma de impulsos eléctricos que se dan a consecuencia de una reacción

química en las dendritas y retransmitida por el axón hacia otra neurona y así sucesivamente hasta llegar al órgano deseado (véase Figura 2.4).

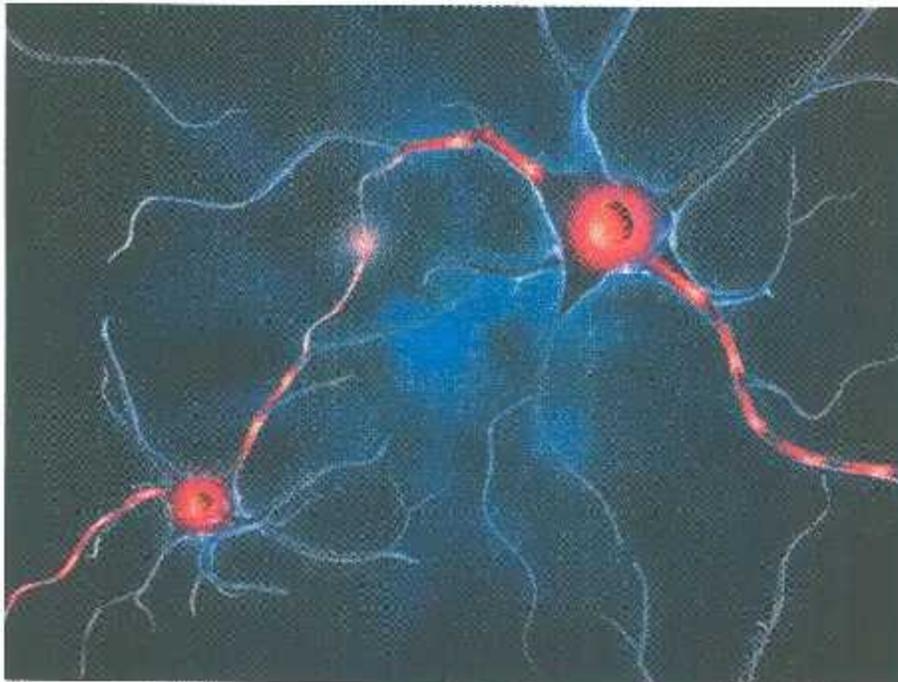


Figura 2.4.- Comunicación entre neuronas.

- 2.1.3 Red Neuronal Artificial

Una red neuronal artificial son modelos matemáticos o algoritmos que se inspiran en el modelo de una red neuronal biológica para el procesamiento de la información, es decir, trata de emular el cómo procesa el cerebro la información (dando de ejemplo el cerebro humano), aplicado a los sistemas computacionales [5].

Existen varios tipos de redes neuronales artificiales, que se clasifican dependiendo su forma de "aprendizaje", esto es la manera como una red neuronal procesa la información recibida y el resultado que esta obtendría. Hay distintos tipos de aprendizajes, entre ellos el supervisado, el no supervisado y redes híbridas.

El aprendizaje supervisado es aquel donde se espera una respuesta predeterminada a partir de una entrada definida (refiriéndose a la entrada como

cualquier tipo de información), es decir, una entrada específica que recibe la red neuronal artificial al ser procesada debe dar siempre la misma respuesta o la más cercana posible. Por ejemplo si la red neuronal estuviera implementada en un coche y a la red se le indica que si su entrada es la luz roja de un semáforo, la red debe hacer que el vehículo se detenga. Así que siempre que perciba una luz roja la red efectuaría la acción de detener el automóvil.

Las redes neuronales artificiales basadas en el aprendizaje no supervisado (también llamado autosupervisado) son aquellas que no reciben una entrada específica y al ser procesada dan una respuesta que no se puede precisar si está correcta o errónea sino que esta respuesta indica similitud o familiaridad de entradas pasadas con respecto a la entrada actual, es decir, que este tipo de red compara la entrada actual con entradas anteriores y determina si son iguales o similares, pertenecen a la misma categoría o son distintas entre sí. Por ejemplo a la red se le muestra una pelota de baloncesto naranja, guardará esa información y al momento de presentarle otra pelota de baloncesto pero ahora de color negro, la respuesta de la red será que son pelotas similares por tener la misma forma y marcas pero el color es distinto; después se le presenta un balón de fútbol americano, la respuesta de la red será que pertenece a la misma categoría (por la similitud ya que el balón de fútbol americano es ovoide y el de baloncesto es esférico), pero si después se le presenta un bate de béisbol, la respuesta de la red será que este objeto es completamente diferentes a los demás y que no corresponden a la misma categoría.

Las redes híbridas son aquellas que combinan ciertas características de las redes antes mencionadas para complementarse entre sí, como por ejemplo que una entrada sea procesada con una parte autoorganizada y su respuesta sea la entrada de una parte supervisada, haciendo de esta manera que la red primero clasifica varias entradas determinando si son similares, de la misma categoría o diferentes y después dependiendo del resultado asignándoles un valor predeterminado que siempre será el mismo. Un ejemplo sería poner a una red neuronal artificial a catalogar los objetos de una casa, la parte no supervisada determinaría que objetos pertenecen a la sala, comedor, cocina, recamaras, etc., y la parte supervisada se encargaría de poner el nombre adecuado a cada objeto, como sofá, estufa, cama, mesa, silla, etc.

En la actualidad hay muchos modelos de redes neuronales, unos ya implementados y otros aun teóricos. Como por ejemplo las siguientes:

- Perceptrón [16]
- Adaline [18]
- Perceptrón multicapa [16]
- Memorias asociativas
- Máquina de Boltzmann
- Máquina de Cauchy
- Propagación hacia atrás (backpropagation) [18]
- Redes de Elman
- Redes de Hopfield
- Red de contrapropagación
- Redes de neuronas de base radial
- Redes de neuronas de aprendizaje competitivo
- Mapas Autoorganizados (RNA) (Redes de Kohonen)
- Crecimiento dinámico de células
- Gas Neuronal Creciente
- Redes ART (Adaptive Resonance Theory)

Enseguida se muestran esquemas de algunas redes neuronales artificiales mencionadas anteriormente:

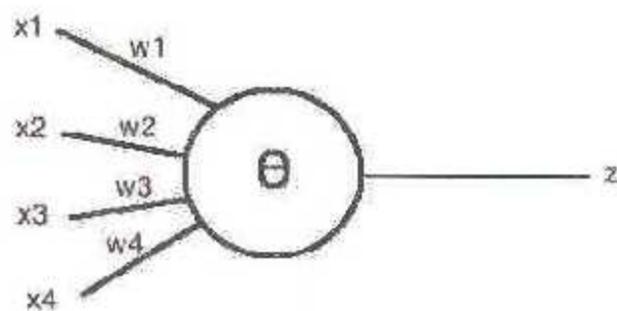


Figura 2.5.- Perceptrón simple.

El modelo de red neuronal del Perceptrón es el más simple de todos (véase Figura 2.5), porque maneja una sola neurona donde recibe los datos de entrada

x_n por medio de los pesos w_n dando como resultado una sola respuesta z por parte de la neurona.

Al igual que el Perceptrón la red neuronal Adaline es constituida por una sola neurona, pero a diferencia de éste, Adaline tiene un factor de corrección a la salida de la neurona (véase Figura 2.6), es decir, que adapta sus pesos para que la respuesta obtenida por la neurona sea lo más cercana a la respuesta deseada.

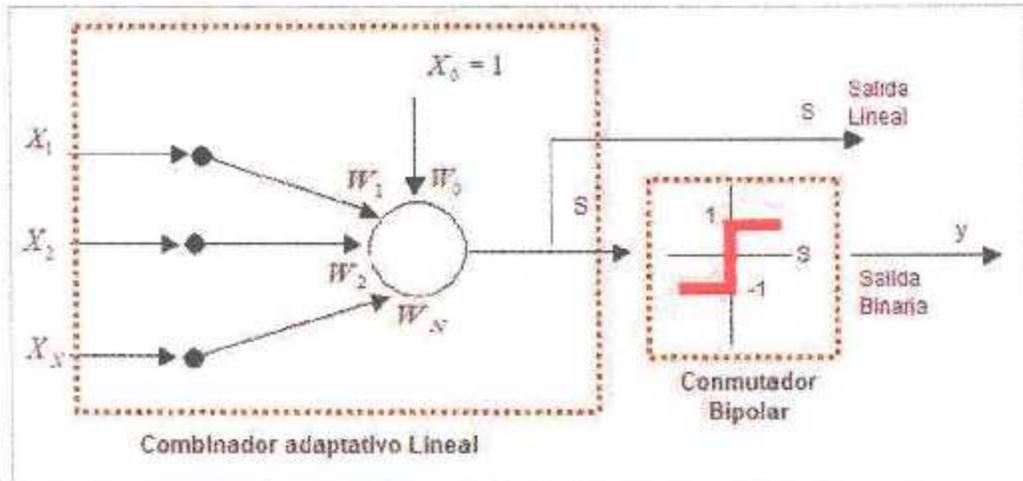


Figura 2.6.- Red neuronal Adaline (ADaptative LINear Element)

La red neuronal del Perceptrón multicapa funciona de igual manera que el Perceptrón pero con la novedad que puede resolver problemas que no son linealmente separables gracias a que su estructura junta varias neuronas para formar capas que son las encargadas de procesar la información (véase Figura 2.7).

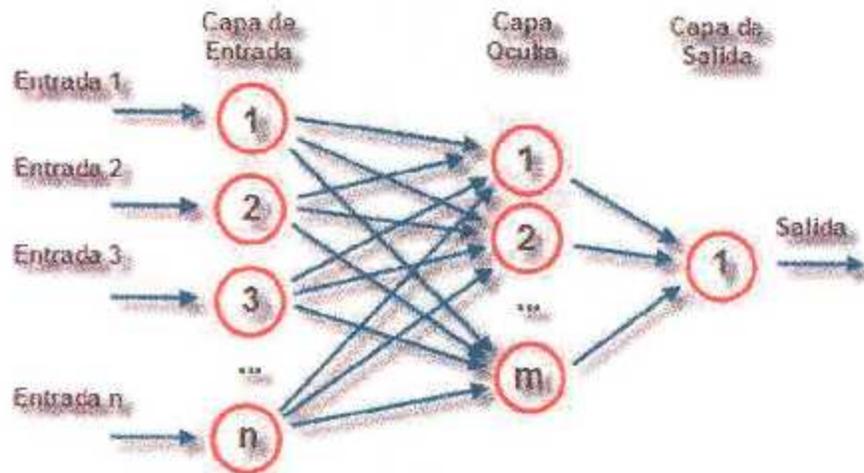


Figura 2.7.- Esquema del Perceptrón multicapa.

La red neuronal de memorias asociativas es una red que trata de emular cómo el cerebro humano almacena y recupera información por medio de asociar esos datos con otra, a diferencia de las computadoras convencionales que requieren la ubicación exacta del elemento guardado que se quiere recuperar, esta red no lo hace de esta manera, sino que gracias a la asociación de los datos con otros elementos puede recobrarlos sin importar cuando se almaceno esa información (véase Figura 2.8).

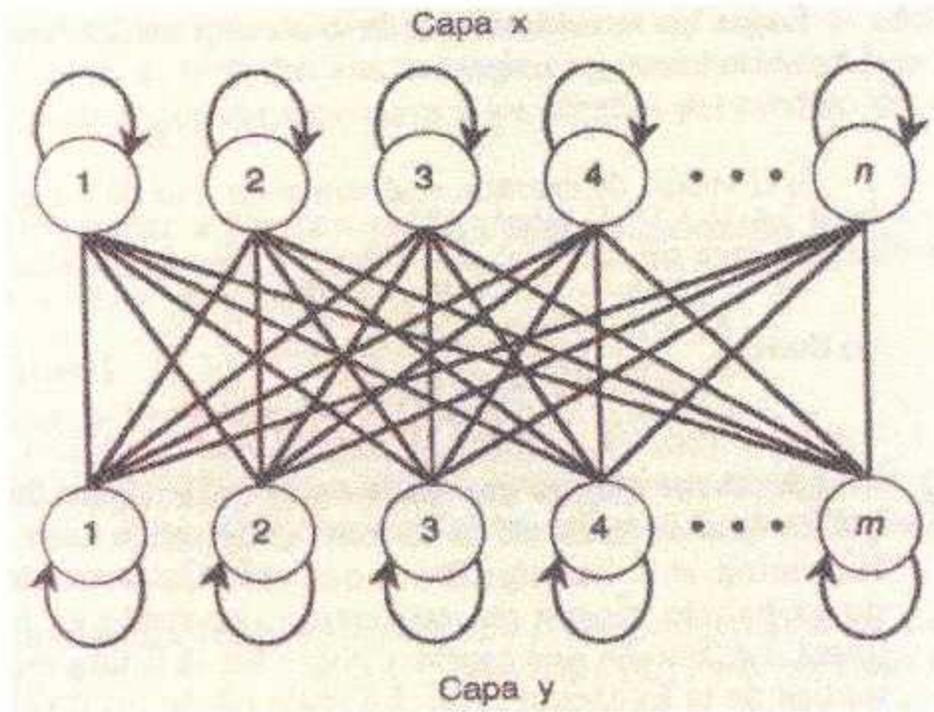


Figura 2.8.- Esquema de la red de Memorias asociativas.

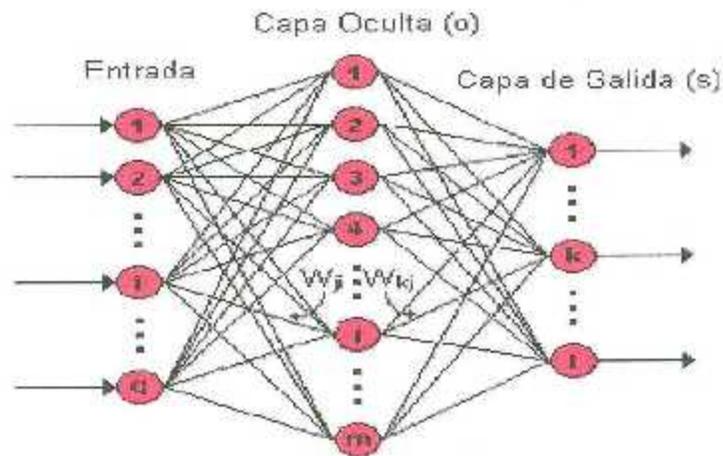


Figura 2.9.- Red Backpropagation.

La red Backpropagation trabaja de manera similar a la red de Perceptrón multicapa pero con la diferencia que al momento de generar la salida la red calcula una señal de error que retroalimenta a las demás neuronas empezando

de la capa final hacia atrás, pero las neuronas solo reciben una parte de la señal, esta parte es proporcional a la contribución que tuvieron para generar ese error, de este modo se actualizan los pesos de cada capa para que la siguiente vez que se reciba la misma información o una semejante la red pueda proporcionar una salida lo más cercana posible a la deseada (véase Figura 2.9).

Los mapas auto-organizados o mapas de Kohonen es una red que difiere de las demás porque su elemento, la neurona, no sigue una misma estructura que las demás, en esta red las neuronas, también llamadas nodos, usan una función de vecindad para procesar la información, es decir, asocian neuronas similares y discriminan neuronas diferentes para formar un mapa de la información recaudada (resultados experimentales de datos al azar, véase Figura 2.10):

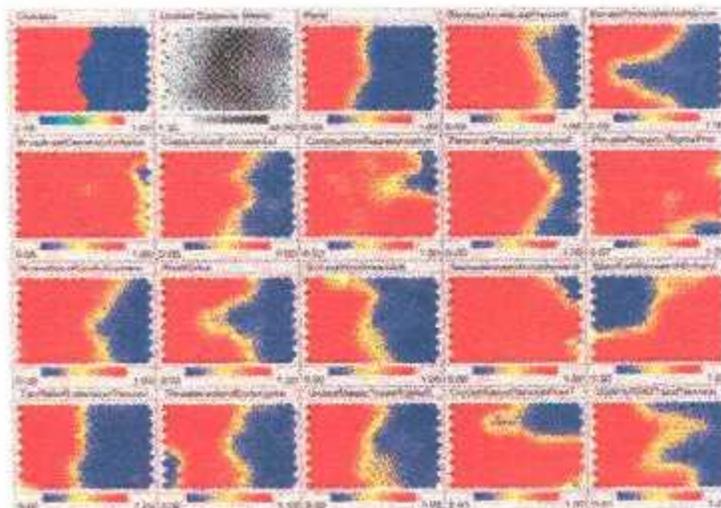


Figura 2.10.- Demostración de la interpretación de la información de un mapa auto-organizado.

La ventaja de las redes neuronales es que son tolerante a fallos, lo que quiere decir que continúa funcionando aun si reciben algún daño, además de trabajar en tiempo real, es decir, que pueden hacer muchas tareas a la vez sin perder ningún dato de información y por último son de fácil inserción dentro de la tecnología existente, se pueden implementar de manera sencilla en casi cualquier aparato electrónico.

Algunas aplicaciones^[6] que tienen las redes neuronales artificiales están enfocadas en análisis y procesados de señales como son la predicción, el

modelado de un sistema y filtrado de ruido; otras aplicaciones son en el reconocimiento de imágenes y de patrones, en estas áreas las redes clasifican los objetivos detectados por medio de sensores, interpretan la información captada y la transforman en una representación diferente que requiere menos memoria; también tiene aplicaciones en el área de social referente a diagnósticos médicos en los cuales la red simula la evolución de ciertas enfermedades y da la posibilidad a su investigación sin arriesgar vidas humanas; por último tiene aplicaciones orientados a la robótica y servo-control ayudando a encontrar un método de cálculo computacional aceptable para compensar las variaciones físicas que se producen en el sistema.

- 2.1.4 Java

Java es un lenguaje de programación de propósito general, concurrente, orientado a objetos, que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible y una plataforma informática comercializada por primera vez en 1995 por Sun Microsystems y que en la actualidad se encuentra en desarrollo por la compañía ORACLE. Java es la base para prácticamente todos los tipos de aplicaciones de red, además del estándar global para desarrollar y distribuir aplicaciones móviles y embebidas, juegos, contenido basado en web y software de empresa ^[6].

El lenguaje Java se creó con cinco objetivos principales:

1. Debería usar el paradigma de la programación orientada a objetos.
2. Debería permitir la ejecución de un mismo programa en múltiples sistemas operativos.
3. Debería incluir por defecto soporte para trabajo en red.
4. Debería diseñarse para ejecutar código en sistemas remotos de forma segura.
5. Debería ser fácil de usar y tomar lo mejor de otros lenguajes orientados a objetos, como C++ (lenguaje de programación).

Este lenguaje es multiplataforma lo que quiere decir que puede ser implementado en cualquier sistema operativo.

- 2.1.5 Procesamiento digital de una imagen

Una imagen en forma genérica es una representación en 2 dimensiones de un objeto de 2 o 3 dimensiones. Esta representación puede ser definida por colores o diferentes niveles de gris. Las imágenes pueden ser divididas en tres tipos las análogas, digitales y vectoriales. El propósito del procesamiento digital de imágenes es poder distinguir objetos que no se pueden o es dificultoso ver directamente en la imagen raster (es una estructura o fichero de datos que representa una rejilla rectangular de píxeles o puntos de color). La forma que tiene una imagen raster es una secuencia de líneas ordenadas en columnas (véase Figura 2.11).

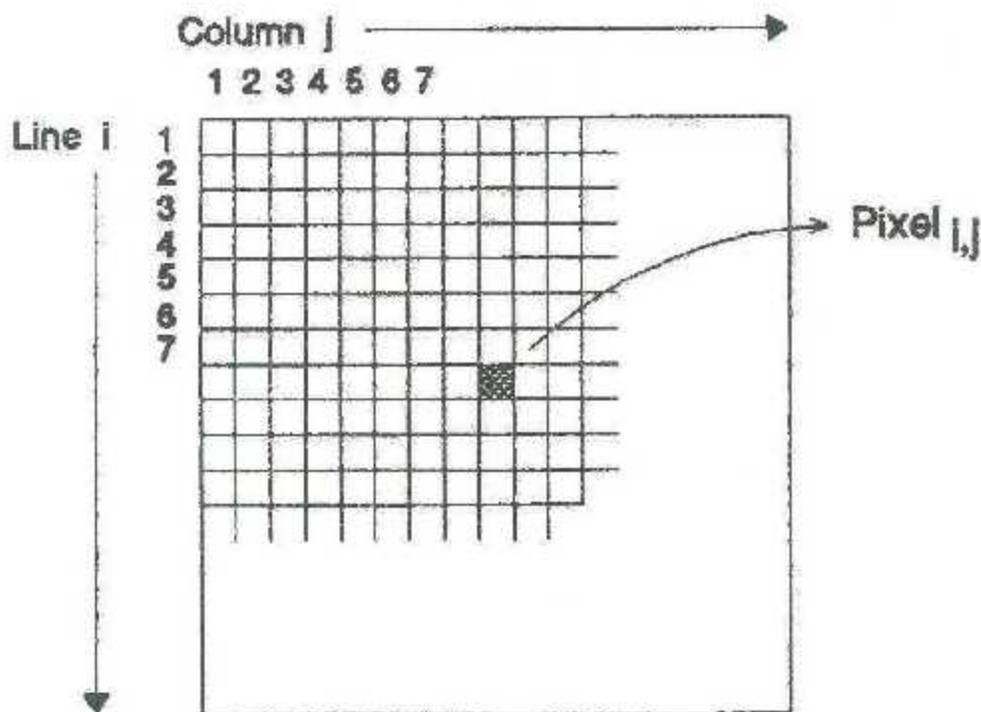


Figura 2.11.- Estructura de imagen raster.

Como se puede observar es una matriz bidimensional donde cada pixel representa un valor de intensidad y es un número digital entero. Todos los conjuntos de datos de teledetección consisten de bandas espectrales simples, los datos multispectrales o multitemporales consisten en varias bandas compuestas por bandas simples. El procesamiento de imágenes de una banda es fundamental para el procesamiento de cualquier tipo de imágenes raster, además el procesamiento de imágenes multitemporales se hace procesando varias bandas secuencialmente. Lo primero que se hace es modificar el brillo y contraste de la imagen, para esto es necesario conocer las estadísticas de la imagen. El proceso de mejora de contraste suele llamarse ensanche de contraste. El método básico consiste en estirar (stretch) el rango de valores de gris al máximo posible (255), esto se puede hacer con una simple función lineal (linear stretch) que usa la ecuación de una recta:

$$y = mx + b \quad \text{ec. 2.1}$$

Donde x es el valor del pixel de entrada (los datos originales); y el de salida (imagen corregida); a es la pendiente de la línea y b es el offset. La pendiente determina el contraste y el offset el brillo de salida (véase Figura 2.12).

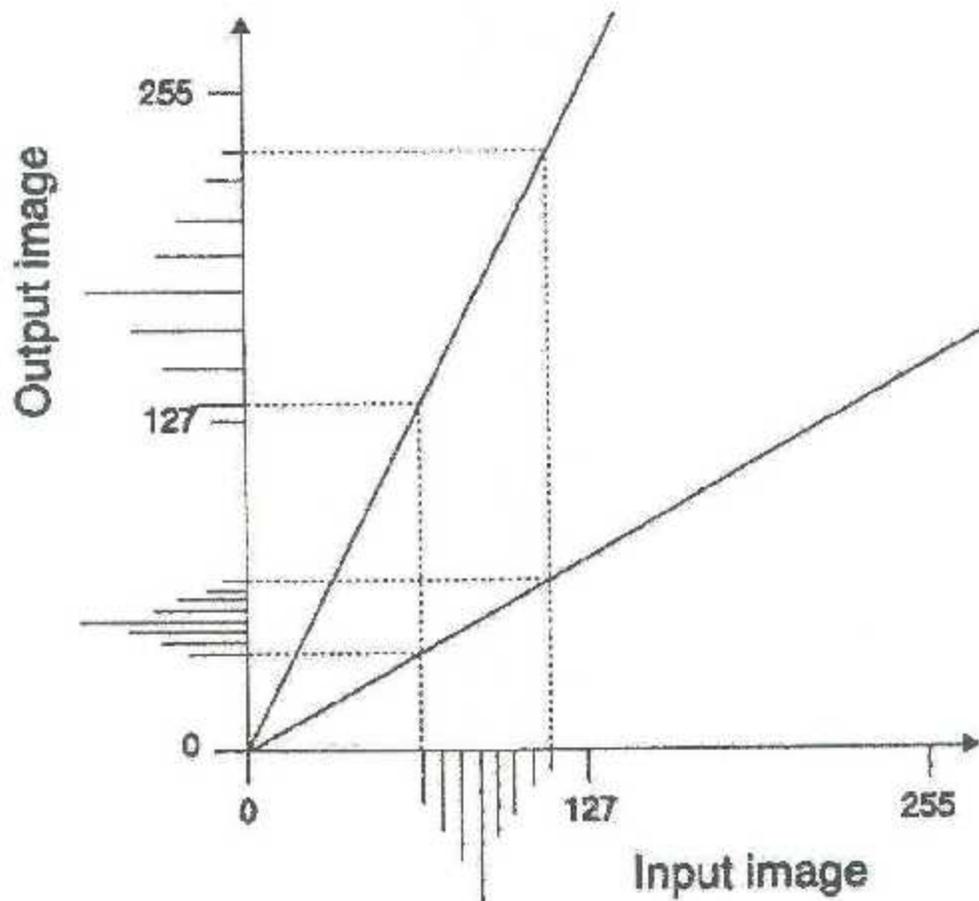


Figura 2.12.- Curva de contraste.

Lo siguiente es el filtrado, la idea de hacer un filtrado de la imagen es reforzar o suavizar los contrastes espaciales existentes entre los valores de los píxeles, la idea es transformar esos valores para que se asemejen o se diferencien más a los píxeles vecinos. El proceso de la convolución espacial consiste en mover una pequeña ventana o núcleo del filtro (filter kernel) a través de toda la imagen y calculando el valor del nuevo píxel a través de todos los valores de los píxeles dentro de ese kernel. Este kernel es una matriz de números que se usa para calcular el valor del nuevo píxel en función de los valores de sus vecinos y esa matriz, conviene aclarar que la matriz se desplaza un píxel a cada paso en la imagen de entrada y que sólo se obtiene el valor para el píxel central de la imagen de salida y luego se aplica al segundo grupo de entrada y así sucesivamente va calculando los puntos de salida.

Y por último se puede hacer algunas transformaciones simples en la imagen, éstas se pueden hacer aplicando las operaciones matemáticas básicas, para ello necesitamos un mínimo de 2 bandas para poder procesarlas. Las operaciones más usadas son:

Sumas: La suma de 2 o más bandas espectrales se usa para reducir el ruido propio de la imagen, se pueden generar imágenes pancromáticas usando las bandas rojo, verde y azul de Landsat TM (1,2 y 3).

Diferencias: Se usan básicamente para determinar cambios, por ejemplo imágenes tomadas en distintas fechas.

Cocientes: Los cocientes sirven para detectar o acentuar características especiales como el contenido de hierro en rocas o suelos o la vegetación. También se los usa para disminuir la influencia del relieve. Un punto a tener en cuenta al realizar un cociente es que no se produzca una división por cero, lo cual es bastante normal en el caso de lagos o superficies oscuras por lo que conviene adicionar 1 siempre al divisor.

También se encuentra la transformación RGB – IHS, que es un cambio de coordenadas de RGB a IHS, del sistema de colores al sistema intensidad, saturación y tono. La intensidad es el brillo, la fuerza del color, Hue o tono es el color propiamente dicho y la saturación es la cantidad de color.

• 2.1.6 Antecedentes

ART son las siglas en inglés de Teoría de la Resonancia Adaptativa (Adaptive Resonance Theory), desarrollada por Stephen Grossberg y Gail Carpenter en 1986, la cual fue una red neuronal diseñada para tener cuatro características principales: plasticidad, la habilidad de cambiar o deformarse; estabilidad, la habilidad de recordar lo aprendido anteriormente; resonancia, la aplicación repetida de una señal para reflejar la comunicación continua entre dos neuronas; y adaptación, el cambio formulado por el resultado de la resonancia.

La red neuronal ART2 es llamada así por ser la segunda versión de sí misma desarrollada en 1987. La diferencia entre las dos versiones es que la primera ^[10] trabaja con entradas binarias solamente, es decir, solamente con 1 y 0, además

que las capas de la primera versión no empiezan con la misma información o parámetros. En cambio la segunda versión trabaja con entradas reales (dícese cualquier valor de entrada) y las capas comienzan con los mismos parámetros. Además de estas dos versiones hay otras pero no serán mencionadas.

- 2.1.7 Estado del arte

En la actualidad se han encontrado varios modelos de aplicación de esta red neuronal artificial, tales como para análisis de datos, reconocimiento de patrones y predicciones de eventos [17]; modelos propios de cada autor en sus respectivos proyectos de investigación, pero nada estandarizado en un lenguaje de programación de alto nivel. Por tal motivo resulta difícil ver en circulación algún programa de computadora que nos ayude aplicar de manera concisa y práctica esta red neuronal.

Por lo que aunque exista información circulando a cerca de esta red, los modelos probados no sobresalen más allá de la teoría o modelos en fase de experimentación.

Capítulo 3

3.1 Desarrollo e implementación del proyecto

- 3.1.1 Etapa de diseño

En esta etapa se eligió la plataforma donde se programaría la red neuronal artificial, siendo en este caso el lenguaje de programación Java, se escogió por la flexibilidad que tiene el lenguaje y su capacidad de adaptarse en casi cualquier componente electrónico como se había mencionado antes.

Para trabajar con cámaras de video y que éstas fueran capaces de reconocer rostros se necesitaban algoritmos de reconocimiento facial, para lograr esto se integró al código una API llamada OpenCV. La API es una interfaz de programación de aplicaciones (API, por sus siglas del inglés: Application Programming Interface) provista por los creadores del lenguaje de programación Java, que da a los programadores los medios para desarrollar aplicaciones Java. La API OpenCV ^{[7][15]} es un conjunto de códigos de programación en los que se desarrollan algoritmos para reconocimientos de patrones tales como rostros, formas geométricas, etc., la cual fue diseñada por la compañía Intel. Esta API está dirigida a los lenguajes de programación Java, C++ y C#.

La tarea específica que esta API realiza en este programa es la detección de uno o más rostros (véase Figura 3.1) cuando estos pasan enfrente de la cámara y guardando esos rostros en el disco duro de la computadora.



Figura 3.1.- Ejemplo del reconocimiento facial.

Las clases que se usan de la API de OpenCV son las siguientes:

- VideoCapture: Esta función es la encargada de enumerar y controlar las cámaras que estén conectadas al equipo, además de guardar la imagen que se va a procesar.
- Mat: Esta función convierte la imagen en matrices numéricas para poder procesarlas.
- Imgproc: Esta función sirve para procesar la imagen, es decir, cambiar el tamaño, ajustes de brillos, aplicar filtros, etc.
- CascadeClassifier: Esta función nos permite aplicar los algoritmos de reconocimiento de patrones a las imágenes ya procesadas.
- MatOfRect: Esta función nos permite separar por medio de una matriz las coordenadas de los elementos reconocidos por medio de la función CascadeClassifier.

Aunque sólo se usa una pequeña porción de las capacidades de la API, es muy importante dentro del programa para su función.

La red neuronal que se utilizaría para la implementación de este proyecto se optó por una red neuronal artificial no supervisada, ya que estas no requieren entrenamiento previo para empezar a trabajar y por ser redes que trabajan de manera continua.

En este caso se eligió la red neuronal artificial ART2 (que son siglas en inglés de "Adaptive Resonance Theory 2" o "Teoría de Resonancia Adaptativa 2"), esta red neuronal aparentemente en la actualidad sigue siendo teórica, es decir, no se ha visto implementada en ningún código de programación o en aparato electrónico.

La red ART2 ^[8] ^[9] está conformada por dos capas de neuronas las cuales son la capa de entrada y la capa de salida respectivamente (véase Figura 14). Las dos capas empiezan con la misma información o parámetros. Al recibir una entrada esta se aloja en una de las neuronas de la capa de entrada y esa neurona es enviada a la capa de salida y es comparada con todas las neuronas de esa capa. Al haber una coincidencia la resultante es la neurona ganadora la cual regresa a la capa de entrada para actualizar todas las neuronas de entrada y emitir un

resultado. Si no hay coincidencia de algún tipo la neurona enviada a la capa de salida se almacena y se forma una nueva neurona en esa capa (véase Figura 3.2 y 3.3).

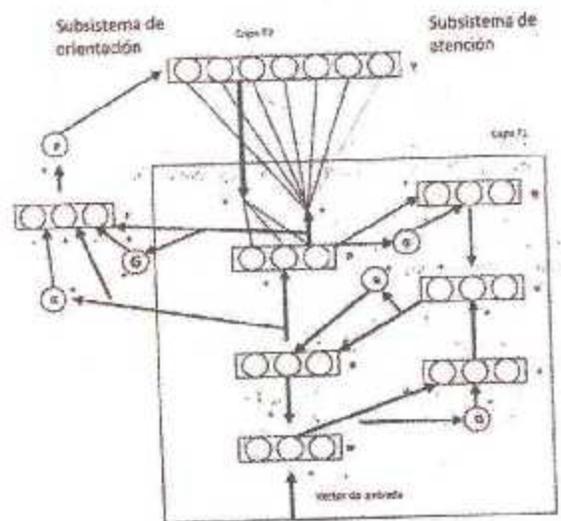


Figura 3.2.- Estructura básica de una red ART2.

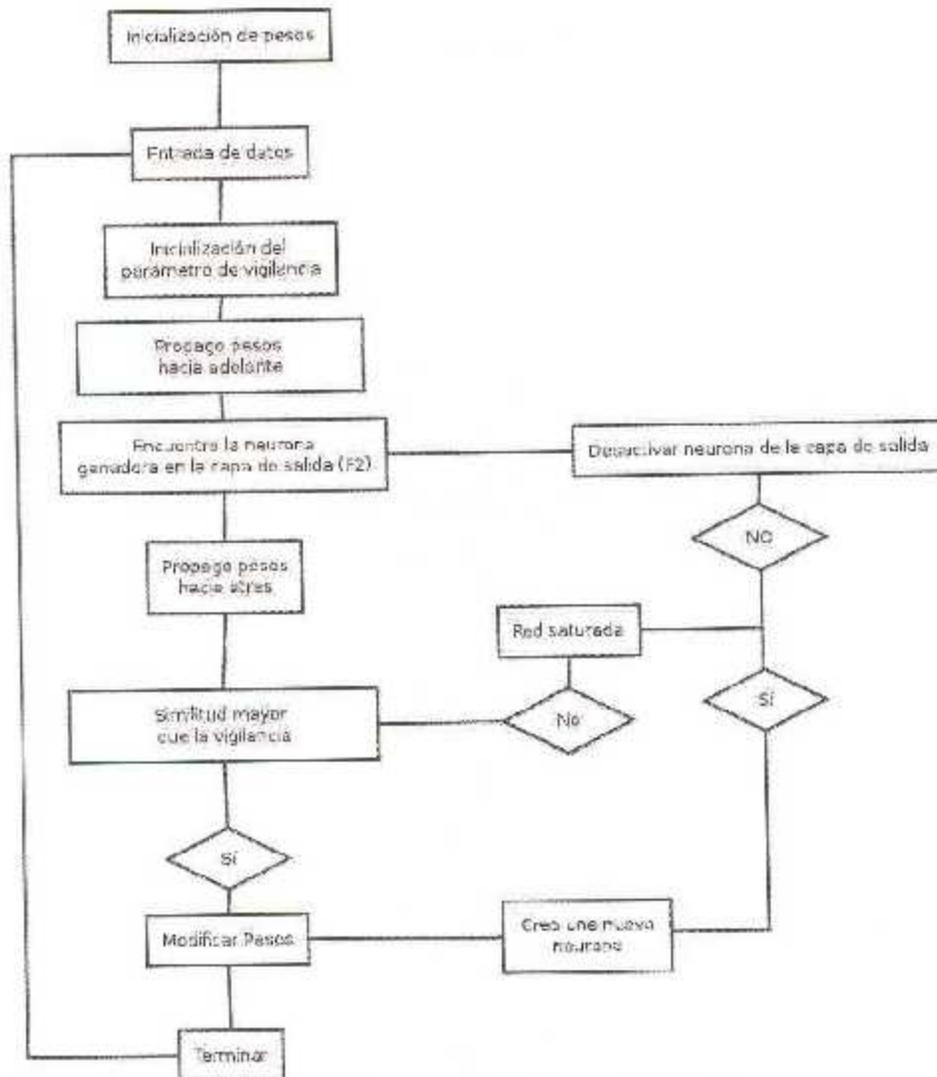


Figura 3.3.- Diagrama de flujo de la red ART2.

Esta red neuronal artificial tiene una característica adicional, la cual es que maneja una memoria a largo plazo, esto quiere decir que tiende a guardar toda la información que recaba para poder acceder a ella cuando lo requiera, a diferencia con otras redes neuronales que tienen que ser entrenadas cada vez que son arrancadas y al momento de dejar de operar borran toda información.

Para ayudar a la memoria de largo plazo de la red ART2 se acoplo a ella una base de datos que se desarrolló en la plataforma MySQL ^[11]. Esta plataforma de

código abierto permite crear base de datos de manera sencilla, tanto en línea como de manera local y es compatible con el lenguaje de Java.

Gracias a esta base de datos la red es capaz, al iniciar cada vez el programa, de acceder a los parámetros con los cuales va a llenar sus capas y empezar a trabajar.

- 3.1.2 Etapa de desarrollo

En esta fase se utilizó el programa Eclipse^[2] compuesto por un conjunto de herramientas de programación de código abierto multiplataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido". El cual nos permite desarrollar el lenguaje de programación de Java de manera sencilla y nos ayuda a compilarlo (reorganizar el programa en un paquete que pueda ser ejecutado en cualquier sistema operativo).

Para empezar se tiene que crear un nuevo proyecto de Java en Eclipse y determinar cómo se llamará, que versión de Java se utilizará y si se requiere agregar librerías externas para que el proyecto funcione. En este caso el nombre del proyecto es "Atalaya" (véase Figura 3.4), se utiliza la última versión de Java compilación 1.8.0_20 y se agregaron de librerías externas las API de OpenCV y de MySQL.

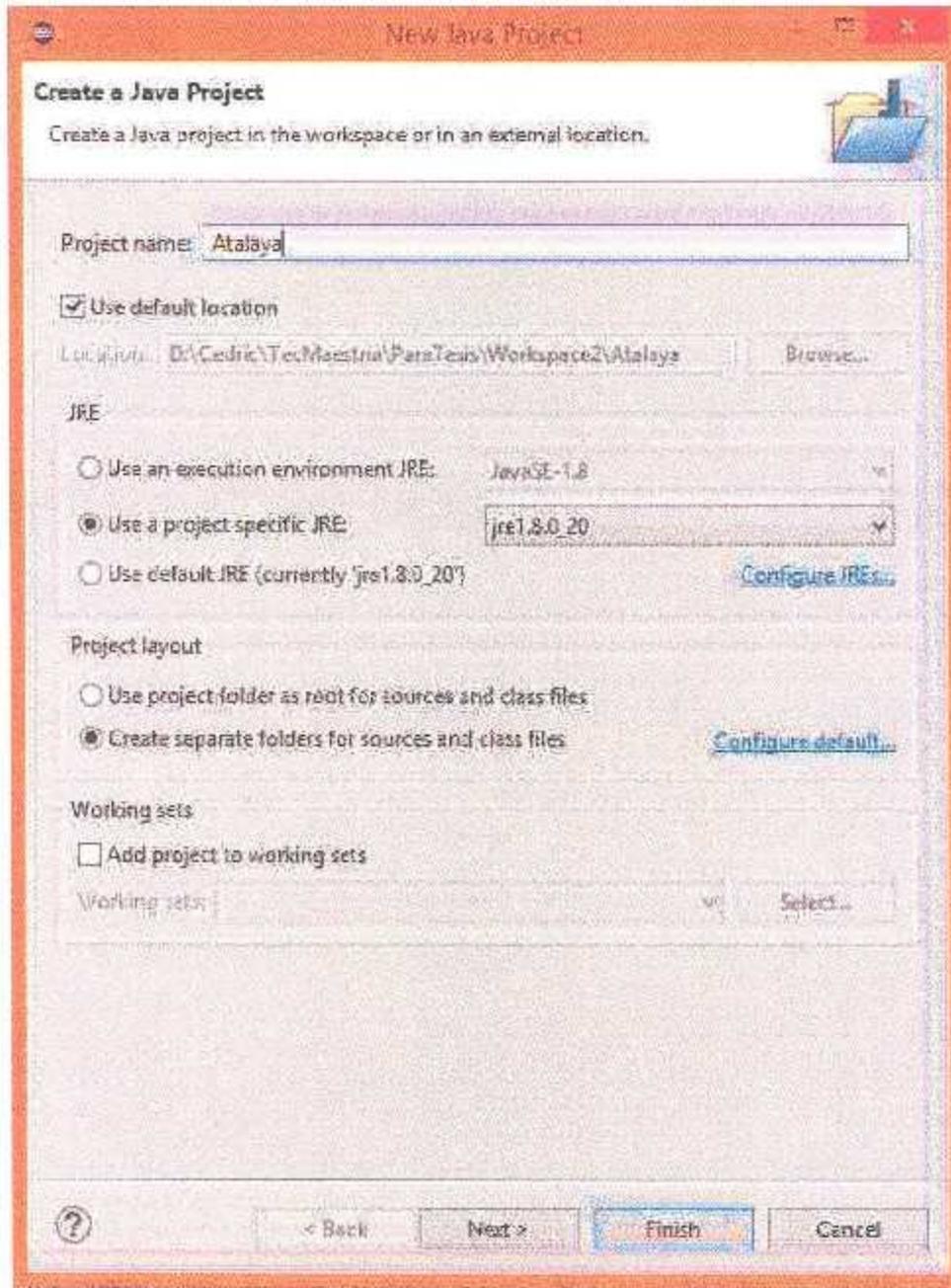


Figura 3.4.- Asiste para la creación del nuevo proyecto.

A continuación se define el nombre de la clase principal del programa y si serán agregadas clases externas. Esta clase principal es la que permite que el programa pueda ser ejecutado y la que nos da acceso a las demás clases del programa.

Para entender lo que es una clase [3] hay que recordar que el lenguaje Java es un lenguaje orientado a objetos; un objeto en el ámbito de la programación es todo aquello que se puede manipular y con lo que se puede interactuar hablando de software, como por ejemplo las ventanas, botones, iconos, etc., que se muestran en el monitor de un ordenador. Las clases por otro lado son lo que definen a un objeto, es una porción de código que contienen las propiedades y métodos del objeto. Las propiedades se refieren por lo general al aspecto visual del objeto en pantalla, como su tamaño, color, forma y animación. Los métodos (también llamados funciones) son el trabajo que realiza el objeto en un programa; un ejemplo sería el botón con una "X" en las ventanas, su función es cerrar la ventana al momento de dar clic sobre él.

La clase principal del programa es la que decide en qué orden y en qué momento las demás clases se van a ejecutar; además de poner en pantalla la ventana principal del programa.

En este programa destacan tres clases:

- Una clase destinada para la programación de la red neuronal ART2.
- Una clase encargada de la manipulación y control de las cámaras conforme a los algoritmos de la API OpenCV para el reconocimiento facial.
- Por último la clase que conecta y manipula la base de datos.

El proceso de este programa sería el siguiente:

En primer lugar se activan las cámaras que serán utilizadas; por medio de éstas y a través de los algoritmos de reconocimiento facial se detecta un rostro; éste es guardado en la memoria temporal de la computadora (memoria RAM), y se desglosa en una matriz de píxeles que sirven como entrada a la red neuronal.

Después, la red ART2 llena sus capas con la información adquirida según la base de datos, alimentada con los rostros digitalizados hasta ese momento.

A continuación la matriz de entrada es evaluada por cada una de las neuronas que contiene la capa de entrada por medio de la siguiente fórmula:

$$\sum_{i=1}^N x_i - N \sum_{j=1}^M y_j \quad (\text{ec. 3.1})$$

Donde N es el número total de elementos de la matriz de entrada (en este caso el número total de píxeles de la imagen del rostro capturado por la cámara), x_i cada elemento de la matriz de entrada, M es el número total de los elementos de la matriz de la neurona actual que ésta evaluando a la matriz de entrada y y_j cada elemento de esa matriz.

Cada evaluación genera un resultado que son comparados entre sí, para determinar cuál valor es el mayor de todos y así encontrar la neurona ganadora; en caso de que algunas neuronas empaten, se consideran todas las neuronas empatadas como las ganadoras. En caso de que todos los valores sean iguales la red determina que es una matriz nueva y pasa a guardarla en la base de datos.

En seguida al tener identificadas la o las neuronas ganadoras estas pasan al test de semejanza comparándose con la matriz de entrada que es determinada con la siguiente ecuación:

$$\text{relación de semejanza} = \frac{\|X * Y\|}{\|X\|} = \frac{\|\sum_{i=1}^N x_i * Y_{ij}\|}{\|\sum_{i=1}^N x_i\|} \quad (\text{ec. 3.2})$$

Donde X es la matriz de entrada y la Y la matriz de la neurona de la capa de salida que se compara con la matriz de entrada.

El resultado de la relación de semejanza es comparado con un valor de vigilancia para verificar si la neurona de la capa de salida es semejante a la matriz de entrada; en caso de ser similares esa neurona es escogida como la respuesta que dará la red; en caso contrario la red guarda la información como si fuera un dato nuevo. La siguiente ecuación determina lo anterior dicho:

$$\begin{cases} \text{relación de semejanza} > \rho = \text{valor aceptado} \\ \text{relación de semejanza} < \rho = \text{valor rechazado} \end{cases} \quad (\text{ec. 3.3})$$

Siendo p el valor de vigilancia, se le da un valor entre cero y uno que refleja que tan preciso se desea que trabaje la red neuronal siendo cero nada preciso y uno cien por ciento preciso.

Finalmente el resultado de este proceso determina si el rostro capturado representa a una persona que ya es reconocida por la red o si es ajena a su base de datos y procede a almacenarla (véase Figura 3.5).

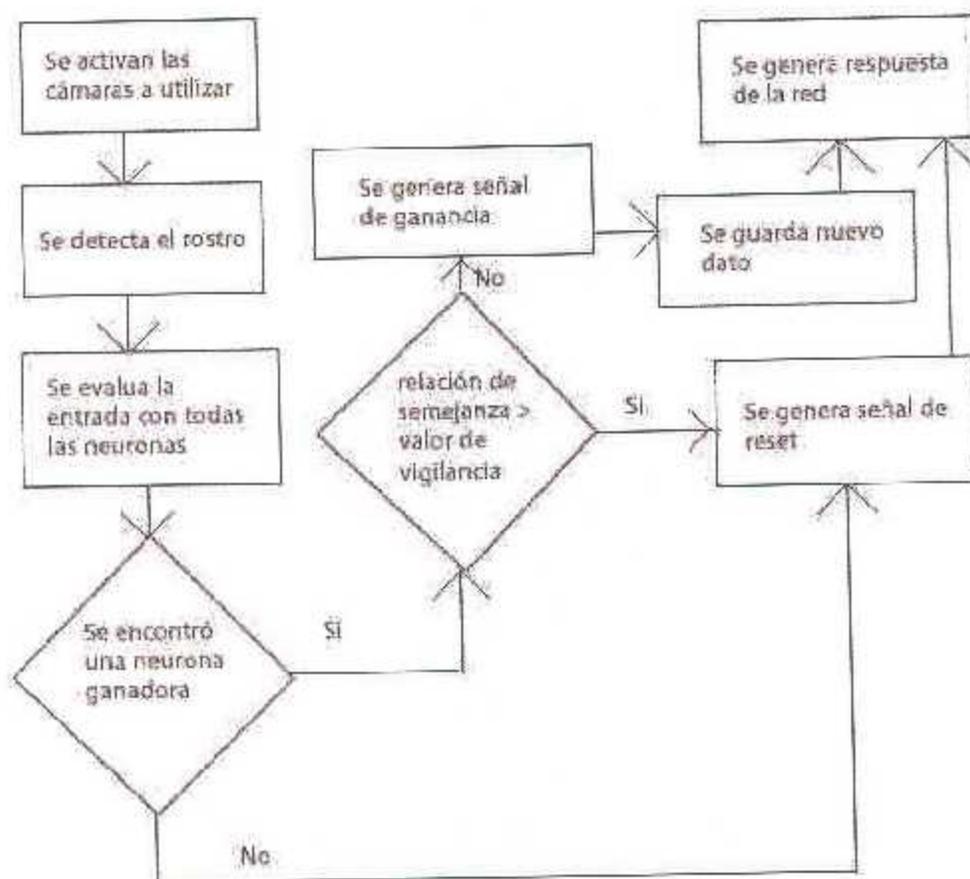


Figura 3.5.- Diagrama de flujo condensado del proceso de la red ART2.

Ya con la información en la base de datos el usuario puede modificar los valores predeterminados que la red asigna a los datos guardados, por valores que pueda identificar mejor, por ejemplo cambiar el valor que da la red al rostro de una persona por el nombre del individuo.

Además de lo anterior se incluyen en el programa dos modos en que la red puede operar: doméstico y empresarial que se detallan a continuación.

- A) Modo doméstico: Éste modo programa a la red para identificar círculos internos que pertenecen al domicilio tales como de las personas que viven en él y sus familiares externos; de igual manera trata de identificar a vecinos y amigos de los inquilinos. Toda persona ajena a lo anterior, la red lo considera como visita o persona desconocida. Éste modo también funciona como alarma convencional.

- B) Modo empresarial: Éste modo prepara a la red para la identificación del personal de la empresa y a que áreas de la misma pueden acceder. Limita el acceso a personal no autorizado a las áreas restringidas si es que las hubiera. Esencialmente este modo actúa como la vigilancia tradicional, es decir, como guardia de seguridad de una empresa.

Capítulo 4

4.1 Pruebas y resultados

Para empezar las pruebas a la red neuronal se tuvieron que hacer unas modificaciones; según la teoría de la red neuronal ART2, la red está diseñada para comparar matrices de las mismas dimensiones. Por ejemplo una imagen de un rostro de 300 x 400 píxeles en la matriz de entrada sólo podría compararse con imágenes de 300 x 400 píxeles que se encuentren en la base de datos de la red y no con las demás que sean de diferentes dimensiones. Para resolver dicho problema se implementó en el algoritmo de la red la prueba no paramétrica de KRUSKAL-WALLIS [14]. Esta prueba determina si los datos de una muestra independiente tienen relación con los datos de otra muestra independiente sin importar el tamaño de las mismas.

Al momento de tener las neuronas ganadoras, si la entrada llega a compararse con una neurona de diferente dimensión, el valor de la *relación de semejanza* es sustituido por el valor de la prueba no paramétrica de Kruskal-Wallis que se rige por la siguiente ecuación:

$$H = \frac{12}{N(N+1)} \sum_{i=1}^r \frac{R_i^2}{n_i} - 3(N+1) \quad (\text{ec. 4.1})$$

Donde H es el valor de la prueba, N es el número total de datos entre las dos matrices, R es la sumatoria de rangos de cada matriz (los rangos son valores consecutivos asignados a cada elemento de las dos matrices) y n es el número total de elementos de cada matriz.

Ese valor no es comparado con el valor de vigilancia, en vez de eso es comparado con el valor de X^2 (chi cuadrada). Este valor representa el límite crítico de la prueba no paramétrica, debajo de límite el valor es aceptado, pero al sobrepasarlo el valor es rechazado. Al tener esta modificación la red puede ahora comparar valores de diferentes dimensiones y tener menos fallos.

La red se probó con cámaras de baja resolución (una cámara Acteck® Lynx View HD ATW-1200), captando imágenes con una máxima dimensión de 600 x 480

pixeles; se probó registrando actividad de una persona frente a ella con poco movimiento y el rostro siempre al frente en dirección de la cámara; también se probó en un lugar concurrido con la cámara fija, recibiendo la imagen de los rostros de las personas que pasaban con un ángulo poco mayor a los tres cuartos con respecto a su vista frontal y con la cámara en movimiento captando rostros de varios ángulos.

El resultado de esta prueba determinó que la red es capaz de identificar y reconocer rostros a un rango máximo de 4 metros de la cámara y de forma frontal a la cámara hasta un máximo de tres cuartos (véase Figura 4.1).

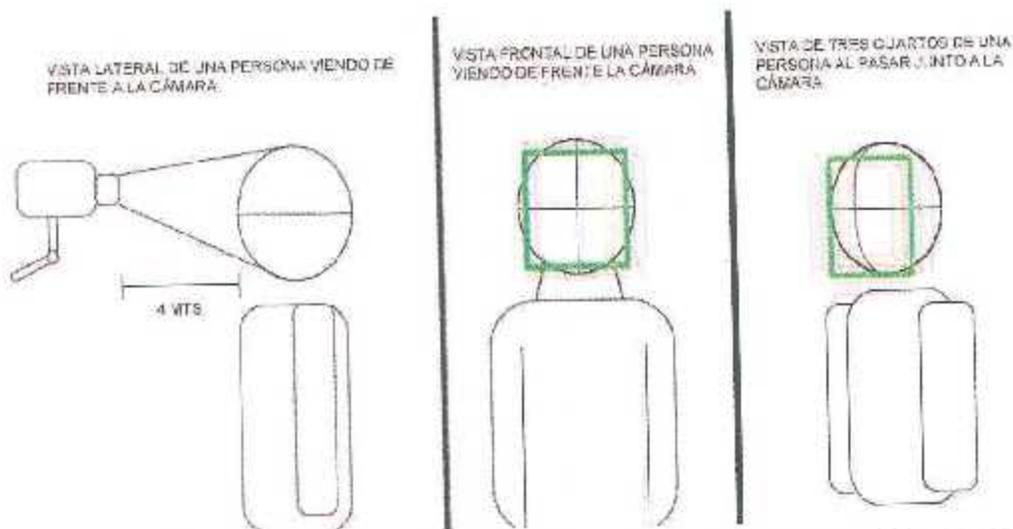


Figura 4.1.- Distancia máxima y posición que debe tener el rostro para que la red pueda identificarlo.

También se probó la cantidad de luz requerida sobre el rostro para que la red pueda reconocerlo. Tomando como parámetro desde un cuarto cerrado a oscuras hasta grabar a la intemperie en un día soleado a medio día.

Llegando a los resultados siguientes: la luz mínima requerida es la unidad de una candela (candela es la unidad para medir la intensidad de luz emitida por un objeto, véase Figura 4.2), que sería equivalente a estar en presencia de la luz que proporciona una vela hasta el aproximado de 300 candelas que sería la luz emitida por la luz de un día soleado pero sin ningún reflejo de algún objeto (véase Figura 4.3), como por ejemplo los espejos, vidrios y pisos reflejantes (pisos que reflejan la luz muy intensamente).



Figura 4.2.- Resultado a la exposición a poca luz aproximada a una candela.

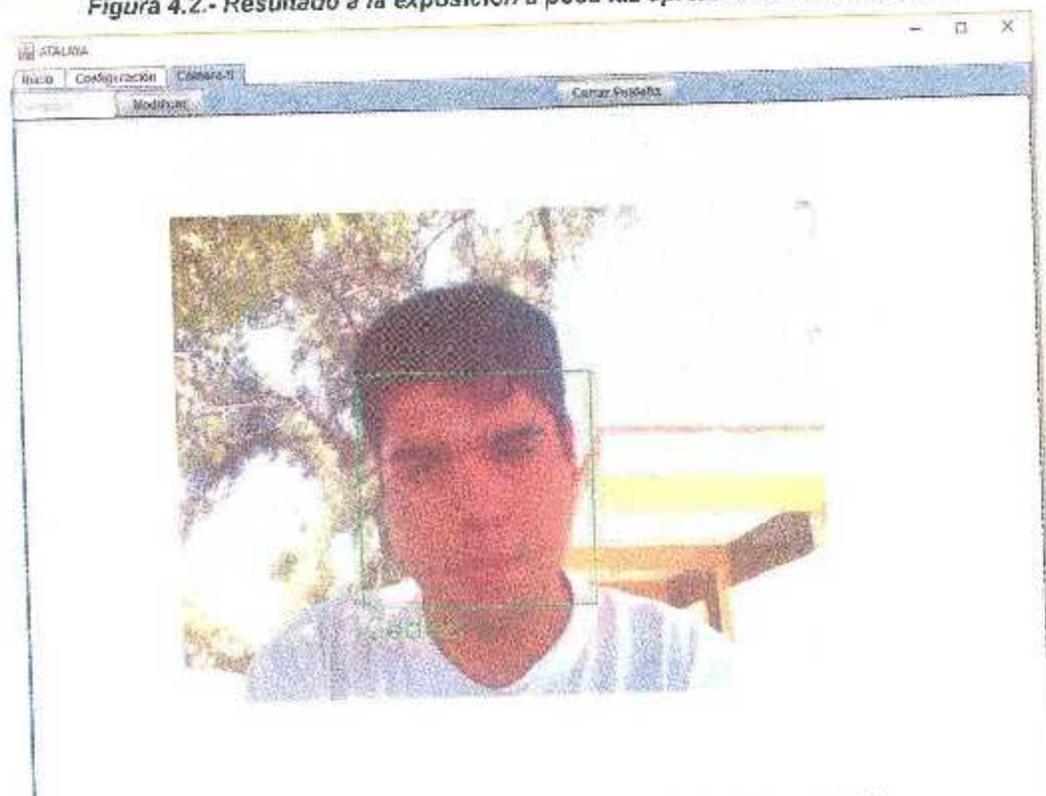


Figura 4.3.- Resultado a la exposición de la luz del medio día.

A continuación se muestran algunas capturas de pantalla del programa en una etapa temprana de programación ya funcional.

La primera captura (véase Figura 4.4) nos muestra la pestaña principal del programa, donde se localiza el estatus de la base de datos, las cámaras activas y el modo operativo en que este el programa en ese momento:

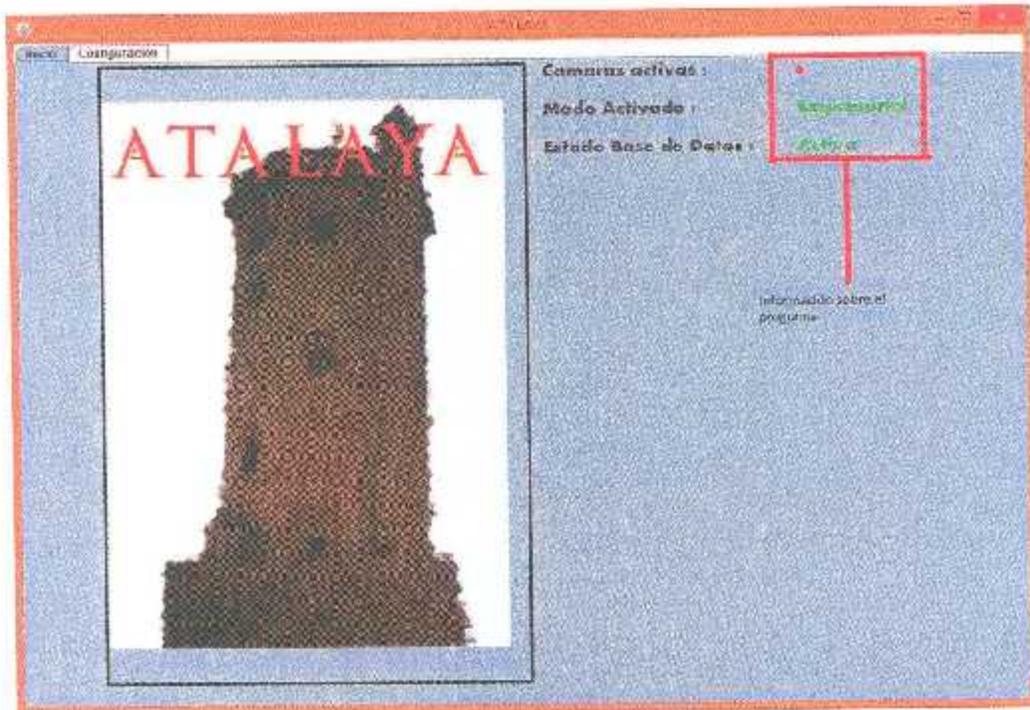


Figura 4.4.- Pestaña de inicio del programa.

En la siguiente captura (véase Figura 4.5) se muestra como la red neuronal identifica el nuevo rostro asignándole un valor y mostrándolo en pantalla en tiempo real:

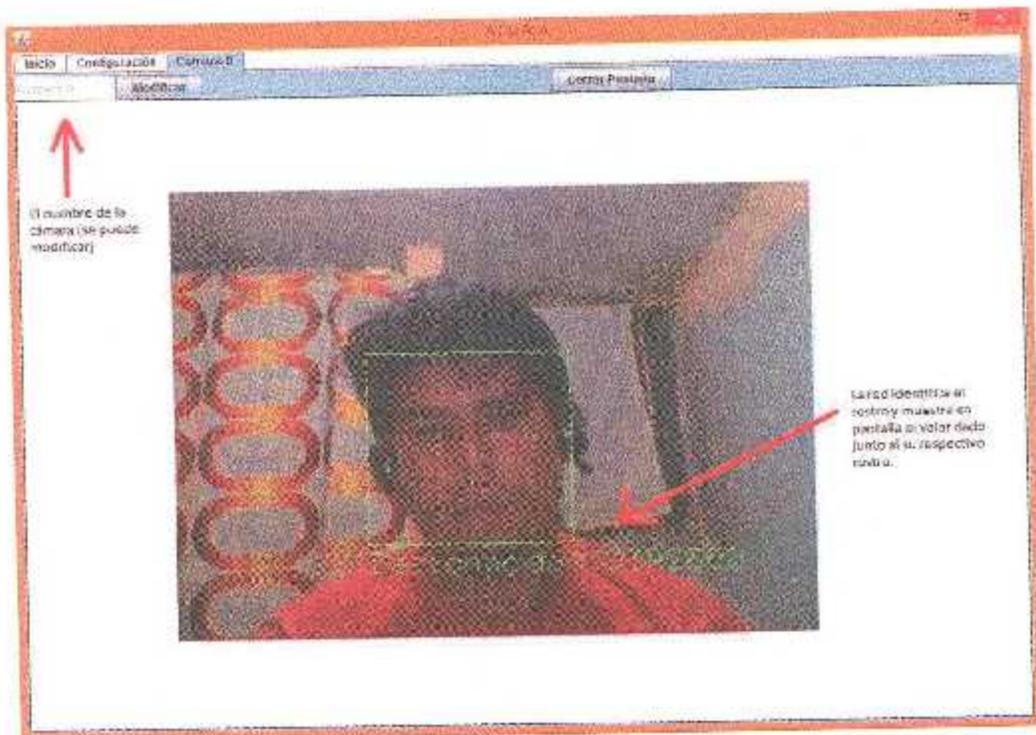


Figura 4.5.- Pestaña de una de las cámaras.

La siguiente captura (véase Figura 4.6) se muestra la pestaña de configuración del programa, en esta pestaña se determina cuantas cámaras están disponibles y cuáles son activadas, además de configurar el modo en que la red operará y una vista a la base de datos que la red está creando:

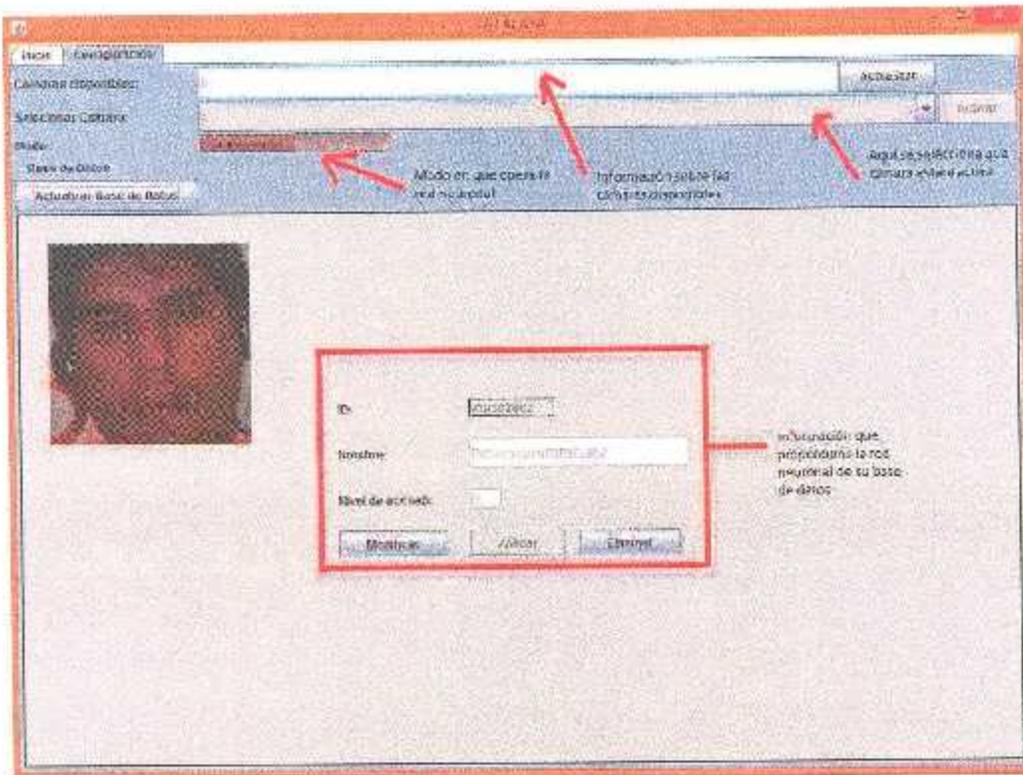


Figura 4.6.- Pestaña de la configuración del programa.

En esta pestaña se puede modificar el nombre que aparece en la vista de la cámara, el nivel de acceso (si la red está operando en modo empresarial) o el grado de parentesco (si la red está operando en modo domestico) como se muestra a continuación (véase Figura 4.7):

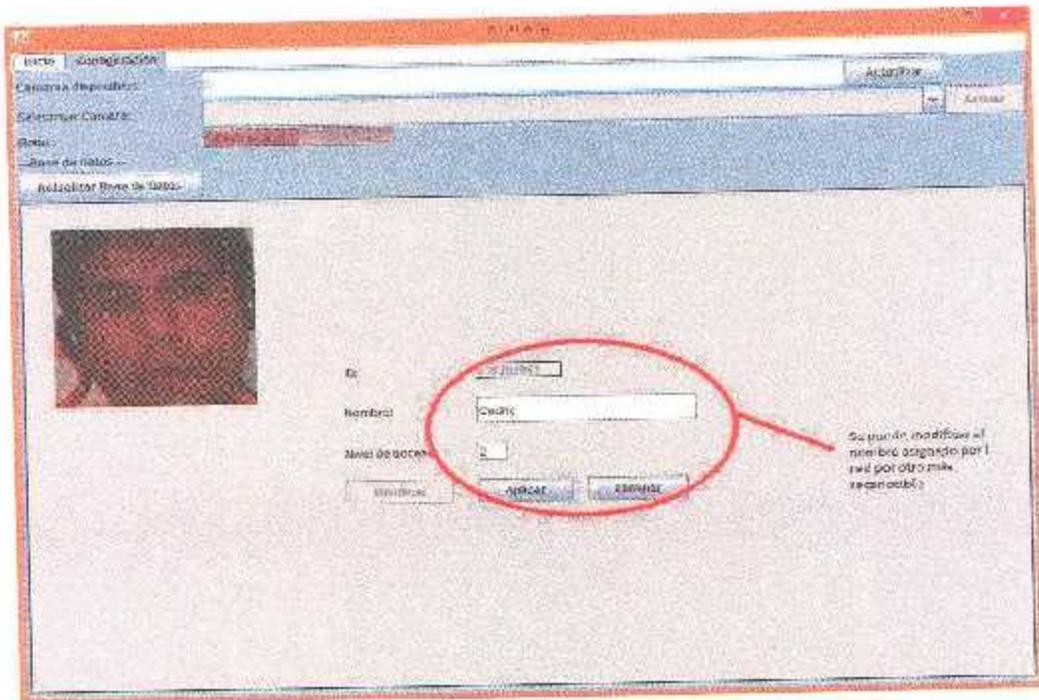


Figura 4.7.- Modificación de la base de datos de manera manual.

En esta última captura (véase Figura 4.8) se muestra como se ve en vista de la cámara cuando se ha modificado el nombre en la base de datos:



Figura 4.8.- Vista de la cámara con el nombre modificado.

Con los resultados obtenidos se puede apreciar en el siguiente esquema (vease Figura 4.9) que la estructura de la red queda prácticamente intacta, la única modificación es en la resonancia de los datos, que son ayudados gracias a la prueba no paramétrica de Kruskal-Wallis:

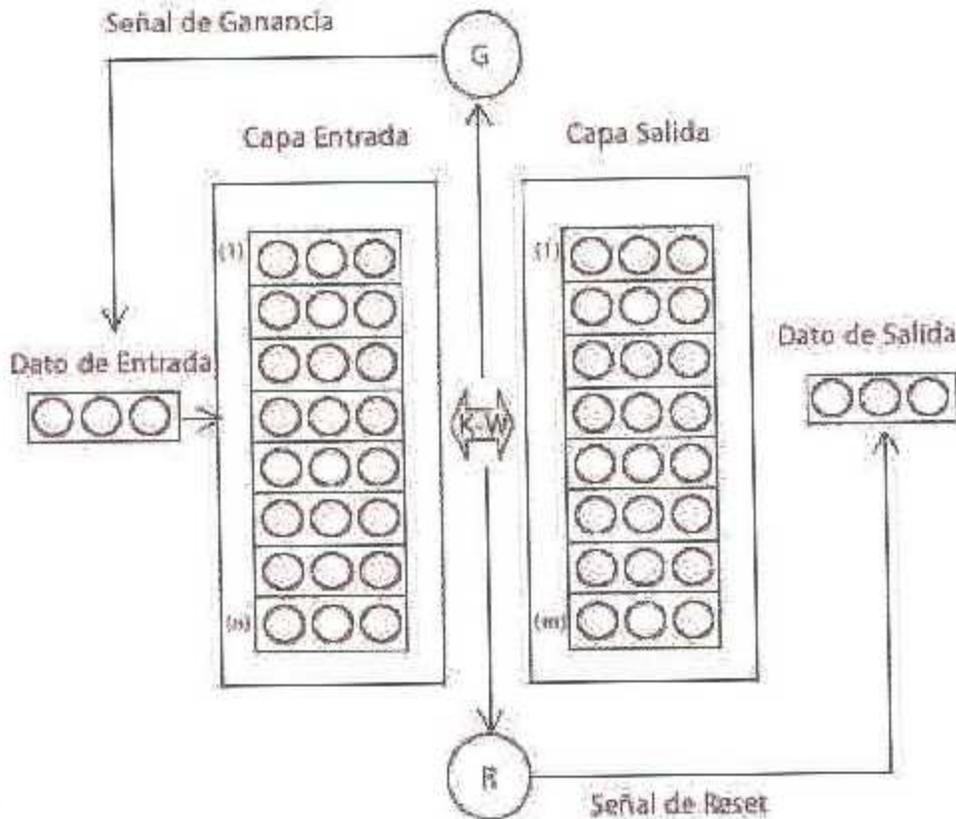


Figura 4.9.- Esquema de la red ART2 modificada.

Lo favorable de la red es que los pesos (normalmente valores que le indican a la neurona que tanta importancia tiene el valor de entrada que recibe) de las entradas son éstas mismas, esto quiere decir que cualquier dato de entrada tiene la misma relevancia para la red, así que al momento de mandar la información entre neuronas se envía la neurona que se recibió de entrada y se compara con todas. Al momento de encontrar coincidencia se manda una señal de reset, que es la que le indica a la red que no es necesario guardar esa información porque

ese dato ya se encuentra en la memoria de largo plazo; por otro lado si no hubo coincidencia alguna se manda una señal de ganancia que indica a la red que ese dato no se encuentra en su memoria y por lo tanto se procede a guardarla en la capa de salida como una categoría nueva y después se actualiza la capa de entrada con la misma información, como se mencionó anteriormente la red neuronal ART2 antes de comenzar un nuevo proceso las dos capas deben contener la misma información, y de esta manera repite el proceso para dar una respuesta.

Al tener esta característica de que las mismas neuronas sean los pesos, la comunicación entre neuronas es más rápida y toma menos tiempo en a la vez de recursos para el reconocimiento de los rostros, como se muestra a continuación varios tiempos registrados por la red, desde el momento que se identifica a un rostro hasta que en pantalla se pueda observar el nombre de la persona que se está reconociendo (véase Figura 4.10):

```

f Tiempo transcurrido para el reconocimiento: 849 milisegundos.
I Tiempo transcurrido para el reconocimiento: 752 milisegundos.
I Tiempo transcurrido para el reconocimiento: 773 milisegundos.
f Tiempo transcurrido para el reconocimiento: 980 milisegundos.
Tiempo transcurrido para el reconocimiento: 595 milisegundos.
Tiempo transcurrido para el reconocimiento: 489 milisegundos.
i Tiempo transcurrido para el reconocimiento: 398 milisegundos.
f Tiempo transcurrido para el reconocimiento: 379 milisegundos.
Tiempo transcurrido para el reconocimiento: 374 milisegundos.
Tiempo transcurrido para el reconocimiento: 381 milisegundos.
Tiempo transcurrido para el reconocimiento: 329 milisegundos.
Tiempo transcurrido para el reconocimiento: 322 milisegundos.
Tiempo transcurrido para el reconocimiento: 363 milisegundos.
Tiempo transcurrido para el reconocimiento: 367 milisegundos.
Tiempo transcurrido para el reconocimiento: 374 milisegundos.
Tiempo transcurrido para el reconocimiento: 482 milisegundos.
Tiempo transcurrido para el reconocimiento: 474 milisegundos.

```

Figura 4.10.- Tiempos registrados por la red para reconocer rostros.

Los tiempos van desde el valor más bajo de 322 milisegundos (0.3 segundos aproximadamente) hasta el valor más alto de 980 milisegundos (0.9 segundos aproximadamente), dando una media de 501 milisegundos que sería el tiempo promedio que tarda en reconocer un rostro.

Para tener una idea de que tan rápido procesa la información ésta red neuronal la compararemos con una red neuronal supervisada llamada backpropagation; la red la obtendremos de la caja de herramientas de Matlab.

Para entrenar a la red ingrese tres rostros de los cuales uno es el mío, para ver si la red puede reconocermé de los otros dos (véase Figura 4.11).



Figura 4.11.- Rostros para el entrenamiento de la red a comparar.

Pero para que la red pueda hacer los cálculos más rápido tenemos que procesar las imágenes para que sean más sencillas para la red, primero aplicamos un filtro para que estén en escala de grises (véase Figura 4.12).



Figura 4.12.- Imágenes en escala de grises.

Y por último aplicamos otro filtro que quede de manera analógica (blanco y negro) pero viendo sus bordes (véase Figura 4.13).



Figura 4.13.- Imágenes en blanco y negro delimitadas por bordes.

Al igual que las imágenes que serían utilizadas en el entrenamiento de la red, cargue otra imagen que sería con la cual se haría la prueba de reconocimiento; también se le aplicaron los mismos filtros que las anteriores (véase Figura 4.14).



Figura 4.14.- Imagen tratada para la prueba de la red.

El entrenamiento de la red fue con 10 capas ocultas y alcanzó el máximo de iteraciones en 1:54 minutos sin llegar a converger (véase Figura 4.15), dando

unos resultado poco acertados (véase Figura 4.16), porque el resultado ideal sería un vector [1 1 1 1 1 1] que indicaría que es mi rostro el reconocido, otros resultados serían [0 0 0 0 0 0] y [1 1 1 1 0 0 0], para la imagen de la mujer y el hombre respectivamente.

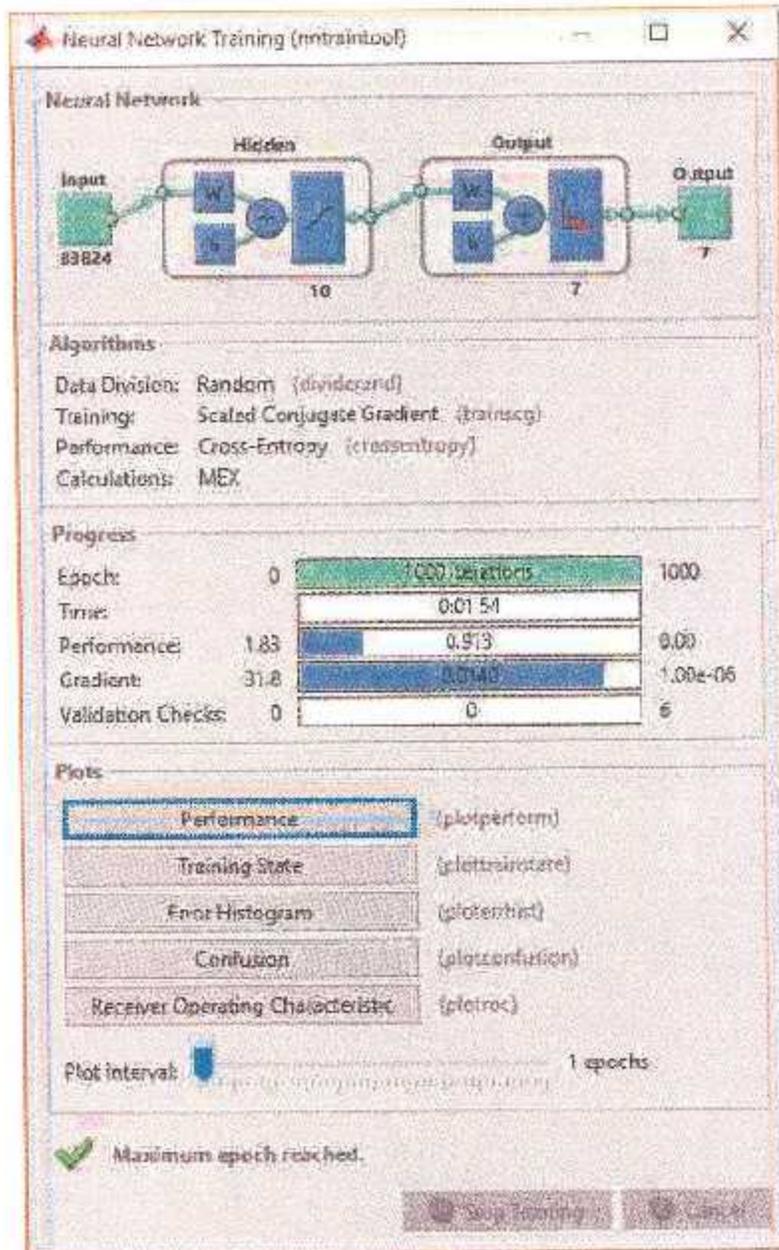


Figura 4.15.- Resultado del entrenamiento de la red.

```

>> Y = net(d4);
>> Y

Y =

    0.0210
    0.0434
    0.5447
    0.3895
    0.0004
    0.0007
    0.0002

```

Figura 4.16.- Respuesta dada por la red al introducir la imagen de prueba.

Otra corrida de la red neuronal fue configurada con 30 capas ocultas (véase Figura 4.17) alcanzando sus máximas iteraciones en 3:14 minutos sin llegar a converger nuevamente pero dando una respuesta más acertada que la anterior (véase Figura 4.13).

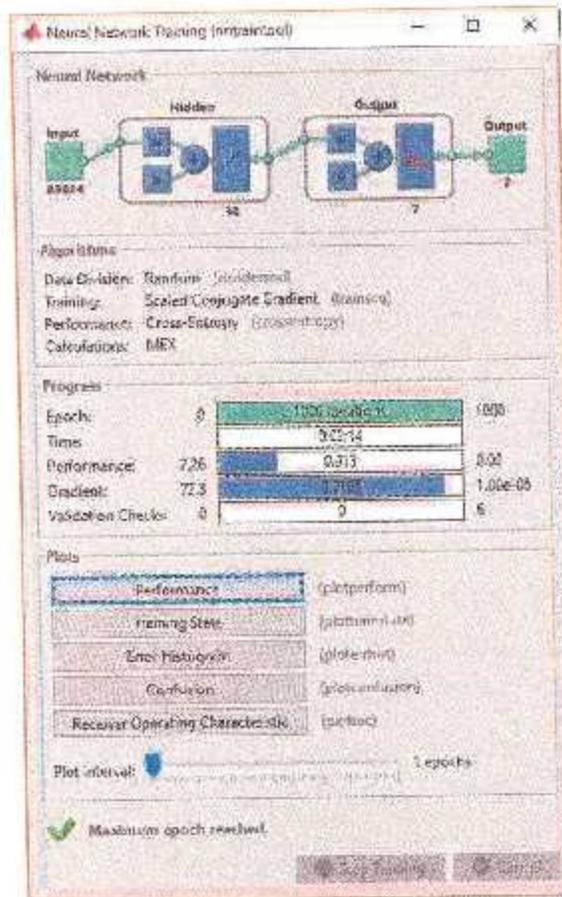


Figura 4.17.- Entrenamiento de la red con 30 capas ocultas.

```
>> X2 = net2(04);  
>> Y2  
  
Y2 =  
  
0.3070  
0.0730  
0.3380  
0.4806  
0.0000  
0.0000  
0.0000
```

Figura 4.18.- Respuesta de la red en la segunda corrida.

Este proceso se le conoce como entrenamiento por lote, donde por medio de diferentes corridas se encuentra la configuración más óptima de red. Por último al igual que con todas las redes, los resultados pueden variar dependiendo de la capacidad que tenga la máquina que ejecute la red.

4.2 Conclusiones

Sabemos que hoy en día los altos índices delictivos van en aumento y que proteger la integridad como el patrimonio personal, es una tarea difícil y que conlleva una gran inversión económica, por lo que si tuviéramos la opción de obtener esa seguridad que deseamos por una inversión mínima y con poco esfuerzo, sería un problema menos de que preocuparse. Derivado de lo anterior, se tiene la propuesta de este sistema de seguridad y vigilancia el cual se obtendría con un bajo costo para la protección de tan preciados bienes.

Este trabajo da sus primeros pasos en lo que respecta a la adaptación de las teorías de redes neuronales artificiales en los lenguajes de programación vigentes, en este caso la teoría de la red neuronal ART2 codificada en el lenguaje de programación Java e implementándola en cámaras.

Con los resultados mostrados la red demuestra funcionar bien y tener una gran eficacia al identificar rostros conocidos por la mayoría de coincidencias que ha tenido.

Lo que limita a la red son pocos factores que podrían mejorar más adelante. Los factores que influyen a esa limitación son los siguientes:

- La baja resolución de las cámaras empleadas en las pruebas, ya que este factor fue determinado así, porque si la red es capaz de identificar los rostros con tan pocos píxeles, es casi certero que la red pueda maximizar la identificación con cámaras de mayor calidad de imagen.
- Los algoritmos utilizados para el reconocimiento facial, ya que estos algoritmos tienden a cometer errores o funcionar de manera errónea como cualquier código de programación. Ya que la red depende totalmente de estos algoritmos porque son los que suministran la información que será la entrada de la red, si el algoritmo falla o funciona mal, lo mismo le pasa a la red.

- La capacidad de la memoria total que ocupa y la velocidad de procesamiento que requiere para realizar sus tareas pueden ser una limitante para realizarlas. La red tiende a guardar todo lo que ve, por consiguiente ocupa memoria física de una computadora; así que el espacio establecido para la red en un disco duro con forme pase el tiempo no será suficiente, ya que al manejar cantidades colosales de información requiere a su debido tiempo más capacidad de procesamiento, el hardware que demande la red será mayor o de mejor calidad.

Pero aun con estas limitaciones que pueden ser subsanadas las capacidades de la red son mayores, ya que pueden aplicarse en otras áreas y de formas diferentes. De tal manera que sus usos son ilimitados y se mejoran conforme vaya pasando el tiempo.

Capítulo 5

5.1 Referencias Bibliográficas

- [1] - <http://www.steren.com.mx/catalogo/category.asp?f=10&sf=105&c=992>, lunes 6 de abril de 2015, 2:21 pm (Última visita realizada).
- [2] - <http://www.forodeseguridad.com/artic/miscel/6087.htm>, lunes 6 de abril de 2015, 2:23 pm (Última visita realizada).
- [3] - <http://www.alarmasseguridad.com/sistemas-seguridad/>, lunes 6 de abril de 2015, 5:01 pm (Última visita realizada).
- [4] - "El sistema nervioso: Desde las neuronas hasta el cerebro", Ernesto Bustamante Zuleta, Editorial Universidad de Antioquia, 1ª Edición Nov. 2007, Páginas 50 – 65.
- [5] - "Redes Neuronales: Conceptos Básicos y Aplicaciones", Damián Jorge Match, Universidad Tecnológica Nacional – Facultad Regional Rosario Departamento de Ingeniería Química Grupo de Investigación Aplicada a la Ingeniería Química (GIAIQ), Rosario, Marzo 2001.
- [6] - https://www.java.com/es/download/whatis_java.jsp, jueves 9 de abril de 2015, 2:02 pm (Última visita realizada).
- [7] - <http://opencv.org/>, lunes 13 de abril de 2015, 10:06 am (Última visita realizada).
- [8] - "ART 2: self-organization of stable category recognition codes for analog input patterns", Gail A. Carpenter and Stephen Grossberg, Applied Optics volume 26, number 23, December, 1987.
- [9] - "Competitive Learning: From Interactive Activation to Adaptive Resonance", Stephen Grossberg, Boston University, Cognitive Science 11, 1987, pags. 23-63.
- [10] - "ADAPTIVE RESONANCE THEORY", Gail A. Carpenter and Stephen Grossberg, Boston University, September, 1998.

[11] - <https://www.mysql.com/>, lunes 13 de abril de 2015, 11:15 am (Última visita realizada).

[12] - <http://www.eclipse.org/>, lunes 13 de abril de 2015, 11:38 am (Última visita realizada).

[13] - "Java2", Jorge Sánchez (www.jorgesanchez.net) año 2004.

[14] - "Use of ranks in one-criterion variance analysis", Kruskal; Wallis (1952). Journal of the American Statistical Association 47 (260): pags. 583–621.

[15] - <http://docs.opencv.org/modules/refman.html>, lunes 13 de abril de 2015, 11:38 am (Última visita realizada).

[16] – "Redes neuronales artificiales y sus aplicaciones", Xabier Basogain Olabe, Escuela Superior de Ingeniería de Bilbao, EHU, 2014.

[17]

<http://www.tesisenred.net/bitstream/handle/10803/9441/tjimm1de1.pdf?sequence=1>, recopilación de varios modelos aplicando la red ART2, viernes 21 de agosto de 2015, 1:31 am (Última visita realizada).

[18] – "ENTRENAMIENTO DE REDES NEURONALES BASADO EN ALGORITMOS EVOLUTIVOS", Luis Federico Bertona, Universidad de Buenos Aires, noviembre 2005.