



INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE INGENIERÍA MÉCANICA Y ELÉCTRICA

**"SIMULACIÓN EN SOFTWARE DEL CONTROL DE VELOCIDAD
DE AUTOMÓVILES BASADO EN SEÑALES VIALES
POR MEDIO DE MEMORIAS ASOCIATIVAS"**

TESIS

*QUE PARA OBTENER EL TITULO DE
INGENIERO EN COMUNICACIONES Y ELECTRÓNICA*

PRESENTAN:

MARTÍNEZ LECHUGA JESÚS

SÁNCHEZ SALINAS LUIS ALBERTO

ASESORES :

M. EN C. ELENA ACEVEDO MOSQUEDA

M. en C. MARCO ANTONIO ACEVEDO MOSQUEDA

ING. GENARO ZAVALA MEJÍA



MÉXICO D.F.

DICIEMBRE 2013

INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE INGENIERÍA MECÁNICA Y ELÉCTRICA
UNIDAD PROFESIONAL “ADOLFO LÓPEZ MATEOS”

TEMA DE TESIS

**QUE PARA OBTENER EL TÍTULO DE
POR LA OPCIÓN DE TITULACIÓN
DEBERA (N) DESARROLLAR**

**INGENIERO EN COMUNICACIONES Y ELECTRÓNICA
TESIS COLECTIVA Y EXAMEN ORAL INDIVIDUAL
C. JESUS MARTINEZ LECHUGA
C. LUIS ALBERTO SANCHEZ SALINAS**

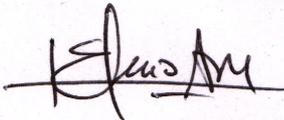
**“SIMULACIÓN EN SOFTWARE DEL CONTROL DE VELOCIDAD DE AUTOMÓVILES BASADO EN
SEÑALES VIALES POR MEMORIAS ASOCIATIVAS”**

**SIMULAR POR SOFTWARE EL CONTROL DE VELOCIDAD DE UN AUTOMÓVIL BASÁNDOSE EN LOS
SEÑALAMIENTOS VIALES, PARA DEMOSTRAR LA EFICACIA DE MEMORIAS ASOCIATIVAS
EMPLEADAS CON EL PROPÓSITO DE RECONOCER DICHS SEÑALAMIENTOS.**

- INTRODUCCIÓN
- ANTECEDENTES
- ESTADO DEL ARTE
- MARCO TEÓRICO Y DE REFERENCIA
- DESARROLLO DE LA SIMULACIÓN
- PRUEBAS Y RESULTADOS
- CONCLUSIONES

MÉXICO D.F. A 9 DE JUNIO DE 2015.

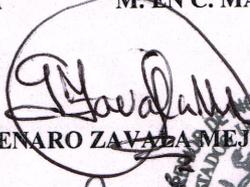
ASESORES



M. EN C. MARÍA ELENA ACEVEDO MOSQUEDA



M. EN C. MARCO ANTONIO ACEVEDO MOSQUEDA



ING. GENARO ZAVAZA MEJÍA



ING. PATRICIA LORENA RAMÍREZ RANGEL
JEFE DEL DEPARTAMENTO DE INGENIERÍA EN COMUNICACIONES Y ELECTRÓNICA

Contenido

INTRODUCCIÓN	i
Prólogo.....	ii
Justificación	iii
Planteamiento del Problema	iv
Hipótesis.....	v
Objetivo General	vi
Objetivos Particulares.....	vi
CAPÍTULO I: ANTECEDENTES Y ESTADO DEL ARTE	1
1.1 Antecedentes.....	2
1.1.1 <i>Control de Crucero</i>	2
1.1.2 <i>Control de Crucero Adaptativo</i>	2
1.2 Estado del Arte	3
1.2.1 <i>Funcionamiento del control de crucero</i>	3
1.2.2 <i>Controlador inteligente de velocidad</i>	5
1.2.3 <i>Funcionamiento del control inteligente de velocidad (CIV)</i>	6
1.2.4 <i>Sistema de información de señales de tráfico (RSI)</i>	7
CAPÍTULO II: MARCO TEÓRICO Y DE REFERENCIA.....	8
2.1 Conceptos básicos sobre memorias asociativas.....	9
2.2 Memorias Asociativas.....	12
2.2.1 <i>Lernmatrix de Steinbuch</i>	12
2.2.2 <i>Correlograph de Willshaw, Buneman y Longuet-Higgins</i>	13
2.2.3 <i>Linear Associator de Anderson-Kohonen</i>	14
2.2.4 <i>La memoria asociativa Hopfield</i>	15
2.3 Memorias Asociativas Morfológicas	17
2.3.1 <i>Memorias Heteroasociativas Morfológicas</i>	18
2.3.2 <i>Memorias Autoasociativas Morfológicas</i>	19
2.3.3 <i>Memorias Asociativas Alfa-Beta</i>	20
2.3.4 <i>Memorias Asociativas Media</i>	22
2.4 Señalamientos viales.....	24
2.4.1 <i>Señales preventivas</i>	24
2.4.2 <i>Señales restrictivas</i>	27

2.4.3 Señales restrictivas de velocidad.....	31
CAPÍTULO III: DESARROLLO DE LA SIMULACIÓN	32
3.1 Manipulación de imágenes por software	33
3.2 Implementación y pruebas de las memorias asociativas morfológicas en MATLAB.....	33
3.3 Implementación en lenguaje C# utilizando Visual Studio 2012	34
3.4 Ingreso y manipulación de imágenes en C#.....	34
3.4.1 Abrir imagen.....	34
3.4.2 Ajustar tamaño de imagen.....	35
3.4.3 Convertir a escala de grises	35
3.4.4 Obtener matriz equivalente.....	35
3.5 Implementación y aplicación de los algoritmos de las memorias asociativas morfológicas ...	35
3.5.1 Vectorizar matriz.....	36
3.5.2 Asociar con patrón de comportamiento	36
3.5.2.1 Patrones de comportamiento	37
3.5.3 Vector de referencia	39
3.5.4 Creación de la matriz máxima	39
3.5.5 Implementar una función de recuperación.....	41
3.6 Interfaz Gráfica de usuario.....	42
3.6.1 Primera etapa.....	42
3.6.2 Segunda etapa	43
3.6.3 Tercera etapa.....	43
3.6.3.1 Eventos condicionados por los temporizadores o timers.....	44
3.6.3.2 Eventos condicionados por botones y la barra deslizante.....	44
3.7 Conjunción del sistema.....	46
CAPÍTULO IV: PRUEBAS Y RESULTADOS.....	47
4.1 Pruebas y resultados del sistema	48
4.1.1 Pruebas al algoritmo.....	48
4.2 Resultados.....	51
4.2.1 resultados de las pruebas al algoritmo.....	51
CONCLUSIÓN.....	52
Conclusiones.....	53

ANEXOS.....	54
Anexo I: Códigos.....	55
1.1 Código de conversión de imagen a Vector	55
1.2 Código de creación de la matriz máxima	55
1.3 Código de Recuperación.....	56
Anexo II: Referencias de la investigación.....	57
II.1 Control crucero	57
II.2 Control de crucero adaptativo	57
II.3 Sistema de información de señales de tráfico.....	57
II.4 Límites máximos de velocidad.....	57
II.5 Normatividad de señales viales.....	57

Índice de Figuras

Figura 1. Control de Crucero	3
Figura 2 Control de crucero adaptativo.....	5
Figura 3 Sistema de información de señales de tráfico de Volvo (exterior)	7
Figura 4 Sistema de información de señales de tráfico de Volvo (interior).....	7
Figura 5 Memoria Asociativa.....	9
Figura 6. Señales Preventivas	26
Figura 7. Señales Restrictivas.....	30
Figura 8. Patrón de comportamiento (Señales Restrictivas)	36
Figura 9. Patrón de Comportamiento (Señales Preventivas)	36
Figura 10. Señales reductoras de velocidad.....	37
Figura 11. Señales con Alerta de Precaución.....	38
Figura 12. Señales con Alerta de Reducción de velocidad.....	38
Figura 13. Diagrama de flujo de la Matriz Máxima	40
Figura 14. Diagrama de Flujo de la función de Recuperación	41
Figura 15. Interfaz primera etapa	42
Figura 16. Interfaz Segunda Etapa	43
Figura 17. Interfaz Tercera Etapa.....	45
Figura 18. Clase Morfológica.....	46
Figura 19. Interfaz de Pruebas con $x=255$	48
Figura 20. Prueba para $x=50$	48

Índice de Tablas

Tabla 1. Operación Alfa.....	20
Tabla 2. Operación Beta.....	21
Tabla 3. Dimensiones del Tablero de las Señales Preventivas	25
Tabla 4. Ubicación longitudinal de las Señales Preventivas.....	25
Tabla 5. Dimensiones del tablero de las Señales Restrictivas	28
Tabla 6. Dimensiones del tablero adicional de las señales restrictivas.....	29
Tabla 7. Valores de los elementos del vector de referencia	39
Tabla 8. Valores de los elementos del vector de referencia	39
Tabla 9. Pruebas de Recuperación para 50 imágenes.....	49
Tabla 10. Pruebas de Recuperación para 75 imágenes.....	49
Tabla 11. Pruebas para un patrón de comportamiento de 12 elementos	50
Tabla 12. Pruebas para un patrón de 100 elementos	50

INTRODUCCIÓN

Prólogo

El presente trabajo surge del interés en ayudar a reducir los accidentes automovilísticos. Primeramente, se hará mención de los sistemas de control para automóviles que tienen el propósito de evitar accidentes; esto con la finalidad de mostrar el marco de referencia que se utilizó para idear el sistema. La información se encuentra en el capítulo I.

El diseño del trabajo está basado en redes neuronales, por ello, el capítulo II explica la teoría sobre tipos de memorias asociativas; ya que es el marco teórico en el cuál se basa el funcionamiento del sistema simulado. También, en el mismo capítulo se hace mención de las normas establecidas por la SCT (secretaría de comunicaciones y transportes) referente a los señalamientos viales en México; para los cuales el sistema efectuará una acción determinada que responda acorde a dichos señalamientos.

Ya teniendo el marco teórico y de referencia, se hace el desarrollo del simulador. En el capítulo III se explican la forma en cómo se diseñó el algoritmo que reconoce los señalamientos viales, la interfaz gráfica, y la forma de respuesta del simulador al presentársele una señal vial o accionar un evento de la interfaz gráfica.

Finalmente, el capítulo IV trata de las diversas pruebas que se hicieron al simulador para determinar la forma y condiciones en la cuales el simulador trabaja de forma eficaz.

Justificación

Se cree que ciencia y tecnología deben ser aplicadas principalmente para cuidado y protección de los seres vivos, más que para otras cosas. Se eligió el tema de control de velocidad en automóviles porque, viviendo en ciudad, a veces se es testigo de accidentes vehiculares y continuamente se escucha en los noticieros sobre este tipo de incidentes. Los accidentes vehiculares son unas de las principales causas de muerte en el mundo; y por ello es necesario implementar sistemas que ayuden a reducir las probabilidades de accidentes. Además, se cree que el control de velocidad referido en esta tesis también ayudaría a que el tránsito de vehículos fuera fluido; ya que los congestionamientos viales repercuten negativamente en muchos aspectos.

Se espera que la tesis sea del interés de las dependencias gubernamentales relacionadas con cuestiones de tránsito y vialidad, de las empresas automotrices, de las industrias camioneras y también de aquellas personas u organismos que se empeñan en seguridad y protección vial; porque se cree que con su ayuda el desarrollo de ésta idea seguramente traerá el beneficio de mayor protección y seguridad al viajar en vehículos vía terrestre.

Planteamiento del Problema

Una de las principales causas de accidentes automovilísticos es el desacato de las señales de vialidad, sobre todo por exceder los límites de velocidad. El INEGI estima que en México ocurren alrededor de 300 mil accidentes terrestres al año.

Pueden ser diversos los motivos por el cual no se tomen en cuenta los señalamientos viales, por ejemplo: cansancio, problemas de visibilidad e incluso negligencia. Por el hecho de que sea un humano quien lleve el control de un automóvil, no es posible que en todo momento se atiendan debidamente los señalamientos viales.

Aunque muchos vehículos incorporan sistemas de seguridad activa y pasiva como los sistemas de control de estabilidad, bolsas de aire, entre otros; esto no es suficiente, ya que estos sistemas actúan de último momento, cuando un evento drástico ha sucedido.

Otros sistemas de control como el de control de crucero, permiten al conductor regular la velocidad de una forma manual, lo cual deja abiertas las posibilidades de un accidente por distracción. En el caso del sistema de control crucero adaptativo ha demostrado su eficacia para evitar accidentes, pero es un sistema que ejerce control cuando está por suceder un evento riesgoso.

Además, en la actualidad los automóviles son uno de los principales medios de transporte, las grandes ciudades crecen cada vez más y por ello todos los movimientos se concentran en ellas; generando así mayor dificultad para el tránsito de vehículos, trayendo como consecuencias afectaciones en la salud y en lo económico.

Hipótesis

De acuerdo al problema es imperioso el uso debido de los señalamientos viales; porque atendiendo dichos señalamientos, el tránsito de vehículos será más fluido, se optimizaría el tiempo y sobre todo se reduciría la probabilidad de accidentes viales.

Para ello es necesario que en todo momento se atienda a los señalamientos viales. Esto lo hará el automóvil de forma automática; para ello se le debe implementar un sistema que cuente con un algoritmo inteligente basado en memorias asociativas, en específico, de las memorias morfológicas; éste se encargará del reconocimiento de los señalamientos viales.

La simulación por software que se presenta en esta tesis demostrará la eficacia del algoritmo basado en memorias asociativas para el control de velocidad de un automóvil por medio de señalamientos viales. Para ello, se supone que el algoritmo clasificará correctamente los señalamientos viales

En todo momento se atenderán las señales viales y se establecerá el límite de velocidad (cuando se presenten las señales restrictivas de velocidad), además se emitirá una alerta cuando se detecten señales preventivas para que el usuario esté atento de las circunstancias del camino.

Cuando se presenta una señal restrictiva, la velocidad se ajusta de acuerdo al valor de velocidad indicado; por lo que el usuario puede reducir su velocidad, pero no podrá incrementarla sobre ese valor.

Por el hecho de respetar restricciones de velocidad, el sistema evita o al menos prolonga el tiempo de que suceda algún evento riesgoso; ventaja que se tiene sobre los sistemas de control crucero y control crucero adaptativo que actúan de último momento.

Objetivo General

Diseñar un software de simulación para el control de velocidad de un automóvil, basándose en los señalamientos viales, para demostrar la eficacia de memorias asociativas empleadas, con el propósito de reconocer dichos señalamientos.

Objetivos Particulares

- ▶ Determinar que señales viales reconocerá el software
- ▶ Proponer patrones de comportamiento de velocidad pertinentes para cada señal
- ▶ Implementar y utilizar el algoritmo de memorias asociativas morfológicas
- ▶ Diseñar la interfaz gráfica de usuario del simulador

CAPÍTULO I:

ANTECEDENTES Y ESTADO DEL ARTE

1.1 Antecedentes

1.1.1 Control de Crucero

En los vehículos actuales se utiliza un sistema llamado “control de crucero”, lo que este sistema hace es mantener un velocidad prefijada por el conductor durante un viaje, generalmente en carretera, frenando automáticamente cuando es necesario. El sistema se desactiva cuando el conductor pisa el pedal de freno y en algunos modelos también se puede desactivar con el pedal de embrague. Existen algunos sistemas que permiten aumentar la velocidad al pisar el acelerador y cuando este se suelta la velocidad vuelve a ser la programada previamente.

La principal ventaja de este sistema es que le evita al conductor la fatiga de ir manteniendo la velocidad y lo enfoca más en mantener la dirección, lo cual es particularmente útil en viajes muy largos.

El control de crucero moderno fue inventado como tal por Ralph Teetor en 1945, el primer automóvil en utilizar este sistema fue el “Imperial” de 1958, llamado autopiloto este sistema calculaba la velocidad del vehículo con base en las rotaciones del eje del diferencial y la regulaba a través de un solenoide.

1.1.2 Control de Crucero Adaptativo

Este sistema incorpora un radar para mantener una distancia de seguridad con respecto al automóvil que viaja adelante.

Si el vehículo se aproxima a algún auto que circule a menor velocidad por delante, el sistema desacelera el coche hasta igualar la velocidad, con el fin de mantener la distancia. Este sistema también actúa cuando un auto se incorpora al mismo carril a una velocidad menor.

El sistema vuelve a alcanzar la velocidad prefijada una vez que el camino por delante se encuentra despejado.

1.2 Estado del Arte

1.2.1 Funcionamiento del control de crucero



Figura 1. Control de Crucero

Utilizar el control de crucero es fácil: suele haber un mando del vehículo para seleccionar la velocidad deseada, y el coche se encarga, él solito, de **aplicar la potencia necesaria para mantener esa velocidad sin que debamos pisar ningún pedal**. El funcionamiento interno parece también extremadamente simple: si la velocidad cae por debajo del valor deseado, el control de crucero aparentemente *pisa* pedal del acelerador; si la velocidad es demasiado alta, hace uso del freno motor. Eso sí, al contrario que el su hermano mayor, **el control activo de velocidad**, el control de crucero estándar no suele hacer uso de los frenos normales.

Sin embargo, hay algunas sutilezas a tener en cuenta. Un control de crucero que se limitara a pisar el acelerador a tope y soltarlo de golpe al alcanzar la velocidad deseada sería muy incómodo y probablemente inseguro. Estaríamos todo el rato experimentando acelerones bruscos, con las correspondientes sacudidas. Además, el cambio en el reparto

del peso que se produce al acelerar o dejar de acelerar bruscamente podría hacernos perder el control del vehículo en las curvas, creando un **grave problema de seguridad vial**. La solución a todo esto es una tecnología que existe desde finales del siglo XIX: **el regulador PID** (Proporcional integral derivativo), que también aparece en otros dispositivos (termostatos y robótica son los ejemplos más conocidos).

El principio es muy sencillo. Cada cierto tiempo (varias veces por segundo), el coche compara la velocidad actual con la deseada, y decide qué potencia debe suministrar el motor. La novedad del regulador PID es que, además de utilizar la velocidad actual del vehículo, también utiliza el historial de velocidades. Es decir, la respuesta del control de crucero no sólo depende de la velocidad actual, sino **también de la velocidad que el coche tenía momentos atrás**.

1.2.2 Controlador inteligente de velocidad



Figura 2 Control de crucero adaptativo

Dentro de los **reguladores inteligentes de velocidad** tenemos el control de crucero, propiamente dicho, que es un sistema que permite establecer una velocidad determinada a la cual deseamos que circule nuestro vehículo sin necesidad de que presionemos el acelerador y el **control de crucero activo** que puede, además de funcionar como un control de crucero normal, realizar funciones de frenado o aceleración dependiendo de la situación del tráfico y siempre hasta determinadas cotas de distancia.

El funcionamiento del **control de crucero adaptativo** o ACC es muy simple. Si disponemos de dicho dispositivo en nuestro vehículo nos mantendrá informado en todo momento ya que su activación es automática. Esto significa que **mediante señales de radar detecta y nos avisa** sobre la presencia de otros vehículos en nuestro camino y así poder reaccionar ante los mismos para evitar un posible alcance. Después y una vez salvado el obstáculo, el vehículo volverá a acelerar hasta situarse a la velocidad predefinida, pudiéndose desconectar bien por el conductor o bien automáticamente y en todo caso, cuando el pedal de freno sea pisado, pulsando el botón “Off” del panel de instrumentos o cuando el vehículo circule a baja velocidad.

Dispositivo que alerta al conductor sobre situaciones críticas

Existen en la actualidad varios **sistemas de control de crucero adaptativos** como, por citar algunos, el *Distronic Plus* de **Mercedes-Benz** con su programa de velocidad activa, el *AdaptiveCruise Control 2010* en el **Ford Taurus**, *ACInndrive* de **Porsche**.

No obstante, independientemente del tipo de sistema que usemos cabe recordar que son de **ayuda a la conducción** y que no sustituyen en ningún caso al ser humano, entre otras

cuestiones, porque los mandos del vehículo son dirigidos o programados por el conductor y no al revés.

El Control Inteligente de Velocidad (CIV) es un sistema de ayuda al conductor que permite calcular y mantener las distancias de seguridad con respecto a otros vehículos, todo un avance tecnológico y un elemento de comodidad añadida.

Actualmente este mecanismo suele incorporarse opcionalmente en los vehículos y tienen la gran ventaja de que analizan continuamente y en tiempo real los alrededores del vehículo y pueden incluso corregir diversos parámetros de marcha.

1.2.3 Funcionamiento del control inteligente de velocidad (CIV)

Los vehículos equipados con Control Inteligente de Velocidad llevan incorporados una serie de sensores basados en tecnologías de radar o de infrarrojos que exploran el espacio por delante del coche midiendo la distancia y la velocidad del resto de vehículos.

Con estos datos una computadora **calcula una distancia de seguridad** y actúa electrónicamente sobre el acelerador o sobre el sistema de frenado para intentar mantener dicho valor de distancia de seguridad.

En caso de una reducción significativa de la distancia de seguridad, el sistema emite una señal acústica de advertencia al conductor. Este mecanismo únicamente frena o ralentiza la marcha del vehículo de manera moderada, es decir, no están diseñados para efectuar una frenada de emergencia con la máxima deceleración, como hacen los frenos.

Además, este sistema funciona normalmente **a partir de una velocidad mínima** (unos 30 km/h), por debajo de la cual debe ser siempre el conductor quien controle la velocidad del vehículo. Y, al igual que ocurre en los sistemas tradicionales de control de la velocidad de crucero, el sistema CIV se desactiva inmediatamente en caso de que el conductor pise el pedal de freno.

1.2.4 Sistema de información de señales de tráfico (RSI)

Es un sistema que la empresa automotriz Volvo ha implementado en sus automóviles modelo V40.

A través de una cámara frontal se detectan las señales de límites de velocidad y de prohibido adelantarse, posteriormente despliega las señales en el tablero para mantener informado al conductor. Si el conductor lo desea, el sistema puede avisarle mediante una advertencia visual si se está excediendo el límite de velocidad.



Figura 3 Sistema de información de señales de tráfico de Volvo (exterior)



Figura 4 Sistema de información de señales de tráfico de Volvo (interior)

CAPÍTULO II:

MARCO TEÓRICO Y DE REFERENCIA

2.1 Conceptos básicos sobre memorias asociativas

El propósito fundamental de una memoria asociativa es recuperar correctamente patrones completos a partir de patrones de entrada, los cuales pueden estar alterados con ruido aditivo, sustractivo o combinado.

Una **Memoria Asociativa** puede formularse como un sistema de entrada y salida, idea que se esquematiza a continuación:

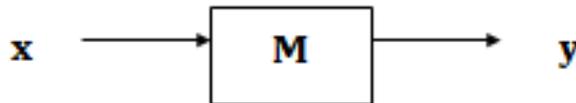


Figura 5 Memoria Asociativa

En este esquema, los patrones de entrada y salida están representados por vectores columna denotados por \mathbf{x} y \mathbf{y} , respectivamente. Cada uno de los patrones de entrada forma una asociación con el correspondiente patrón de salida, la cual es similar a la una pareja ordenada; por ejemplo, los patrones \mathbf{x} y \mathbf{y} del esquema anterior forman la asociación (\mathbf{x}, \mathbf{y}) .

No obstante que a lo largo de las dos secciones restantes del presente capítulo se respetarán las notaciones originales de los autores de los modelos presentados aquí, a continuación se propone una notación que se usará en la descripción de los conceptos básicos sobre memorias asociativas, y en el resto de los capítulos de esta tesis.

Los patrones de entrada y salida se denotarán con las letras negrillas, \mathbf{x} y \mathbf{y} , agregándoles números naturales como superíndices para efectos de discriminación simbólica. Por ejemplo, a un patrón de entrada \mathbf{x}^1 le corresponderá el patrón de salida \mathbf{y}^1 , y ambos formarán la asociación $(\mathbf{x}^1, \mathbf{y}^1)$; del mismo modo, para un número entero positivo k específico, la asociación correspondiente será $(\mathbf{x}^k, \mathbf{y}^k)$.

La memoria asociativa **M** se representa mediante una matriz, la cual se genera a partir de un conjunto finito de asociaciones conocidas de antemano: este es el **conjunto fundamental de aprendizaje**, o simplemente **conjunto fundamental**.

El conjunto fundamental se representa de la siguiente manera:

$$\{(\mathbf{x}^\mu, \mathbf{y}^\mu) \mid \mu = 1, 2, \dots, p\}$$

Donde p es un número entero positivo que representa la cardinalidad del conjunto fundamental.

A los patrones que conforman las asociaciones del conjunto fundamental se les llama **patrones fundamentales**. La naturaleza del conjunto fundamental proporciona un importante criterio para clasificar las memorias asociativas:

Una memoria es **Autoasociativa** si se cumple que $\mathbf{x}^\mu = \mathbf{y}^\mu \forall \mu \in \{1, 2, \dots, p\}$, por lo que uno de los requisitos que se debe de cumplir es que $n = m$.

Una memoria **Heteroasociativa** es aquella en donde $\exists \mu \in \{1, 2, \dots, p\}$ para el que se cumple que $\mathbf{x}^\mu \neq \mathbf{y}^\mu$. Nótese que puede haber memorias heteroasociativas con $n = m$.

En los problemas donde intervienen las memorias asociativas, se consideran dos fases importantes: La fase de aprendizaje, que es donde se genera la memoria asociativa a partir de las p asociaciones del conjunto fundamental, y la fase de recuperación que es donde la memoria asociativa opera sobre un patrón de entrada, a la manera del esquema que aparece al inicio de esta sección.

A fin de especificar las componentes de los patrones, se requiere la notación para dos conjuntos a los que llamaremos arbitrariamente A y B . Las componentes de los vectores columna que representan a los patrones, tanto de entrada como de salida, serán elementos del conjunto A , y las entradas de la matriz **M** serán elementos del conjunto B .

No hay requisitos previos ni limitaciones respecto de la elección de estos dos conjuntos, por lo que no necesariamente deben ser diferentes o poseer características especiales. Esto significa que el número de posibilidades para escoger A y B es infinito.

Por convención, cada vector columna que representa a un patrón de entrada tendrá n componentes cuyos valores pertenecen al conjunto A , y cada vector columna que representa a un patrón de salida tendrá m componentes cuyos valores pertenecen también al conjunto A . Es decir:

$$\mathbf{x}^\mu \in A^n \text{ y } \mathbf{y}^\mu \in A^m \forall \mu \in \{1, 2, \dots, p\}$$

La j -ésima componente de un vector columna se indicará con la misma letra del vector, pero sin negrilla, colocando a j como subíndice ($j \in \{1, 2, \dots, n\}$ o $j \in \{1, 2, \dots, m\}$ según corresponda). La j -ésima componente del vector columna \mathbf{x}^μ se representa por: x_j^μ

Con los conceptos básicos ya descritos y con la notación anterior, es posible expresar las dos fases de una memoria asociativa:

1. **Fase de Aprendizaje** (Generación de la memoria asociativa). Encontrar los operadores adecuados y una manera de generar una matriz \mathbf{M} que almacene las p asociaciones del conjunto fundamental $\{(\mathbf{x}^1, \mathbf{y}^1), (\mathbf{x}^2, \mathbf{y}^2), \dots, (\mathbf{x}^p, \mathbf{y}^p)\}$, donde $\mathbf{x}^\mu \in A^n$ y $\mathbf{y}^\mu \in A^m \forall \mu \in \{1, 2, \dots, p\}$. Si $\exists \mu \in \{1, 2, \dots, p\}$ tal que $\mathbf{x}^\mu \neq \mathbf{y}^\mu$, la memoria será heteroasociativa; si $m = n$ y $\mathbf{x}^\mu = \mathbf{y}^\mu \forall \mu \in \{1, 2, \dots, p\}$, la memoria será autoasociativa.
2. **Fase de Recuperación** (Operación de la memoria asociativa). Hallar los operadores adecuados y las condiciones suficientes para obtener el patrón fundamental de salida \mathbf{y}^μ , cuando se opera la memoria \mathbf{M} con el patrón fundamental de entrada \mathbf{x}^μ ; lo anterior para todos los elementos del conjunto fundamental y para ambos modos: autoasociativo y heteroasociativo.

Se dice que una memoria asociativa **M** exhibe **recuperación correcta** si al presentarle como entrada, en la fase de recuperación, un patrón \mathbf{x}^ω con $\omega \in \{1, 2, \dots, p\}$, ésta responde con el correspondiente patrón fundamental de salida \mathbf{y}^ω .

Una memoria asociativa bidireccional también es un sistema de entrada y salida, solamente que el proceso es bidireccional. La dirección hacia adelante se describe de la misma forma que una memoria asociativa común: al presentarle una entrada \mathbf{x} , el sistema entrega una salida \mathbf{y} . La dirección hacia atrás se lleva a cabo presentándole al sistema una entrada \mathbf{y} para recibir una salida \mathbf{x} .

2.2 Memorias Asociativas

A continuación, en esta sección haremos un breve recorrido por los modelos de memorias asociativas, con objeto de establecer el marco de referencia en el que surgieron las memorias asociativas bidireccionales.

Las memorias asociativas que se presentarán en esta sección, son los modelos más representativos que sirvieron de base para la creación de modelos matemáticos que sustentan el diseño y operación de memorias asociativas más complejas. Para cada modelo se describe su fase de aprendizaje y su fase de recuperación.

Se incluyen cuatro modelos clásicos basados en el anillo de los números racionales con las operaciones de multiplicación y adición: *Lernmatrix*, *Correlograph*, *Linear Associator* y Memoria Hopfield, además de tres modelos basados en paradigmas diferentes a la suma de productos, a saber: memorias asociativas Morfológicas, memorias asociativas Alfa-Beta y memorias asociativas Media.

2.2.1 *Lernmatrix de Steinbuch*

Karl Steinbuch fue uno de los primeros investigadores en desarrollar un método para codificar información en arreglos cuadrículados conocidos como *crossbar*. La importancia de la *Lernmatrix* se evidencia en una afirmación que hace Kohonen en su artículo de 1972, donde apunta que las matrices de correlación, base fundamental de su innovador trabajo, vinieron a sustituir a la *Lernmatrix* de Steinbuch.

La *Lernmatrix* es una memoria heteroasociativa que puede funcionar como un clasificador de patrones binarios si se escogen adecuadamente los patrones de salida; es un sistema de entrada y salida que al operar acepta como entrada un patrón binario $\mathbf{x}^\mu \in A^n$, $A = \{0,1\}$ y produce como salida la clase $\mathbf{y}^\mu \in A^p$ que le corresponde (de entre p clases diferentes), codificada ésta con un método que en la literatura se le ha llamado *one-hot*. El método funciona así: para representar la clase $k \in \{1, 2, \dots, p\}$, se asignan a las componentes del vector de salida \mathbf{y}^μ los siguientes valores: $y_k^\mu = 1$, y $y_j^\mu = 0$ para $j = 1, 2, \dots, k-1, k+1, \dots, p$.

Algoritmo de la Lernmatrix

- *Fase de Aprendizaje*

Se genera el esquema (*crossbar*) al incorporar la pareja de patrones de entrenamiento $(\mathbf{x}^\mu, \mathbf{y}^\mu) \in A^n \times A^p$. Cada uno de los componentes m_{ij} de \mathbf{M} , la *Lernmatrix* de Steinbuch, tiene valor cero al inicio, y se actualiza de acuerdo con la regla $m_{ij} + \Delta m_{ij}$, donde:

$$\Delta m_{ij} = \begin{cases} +\varepsilon & \text{si } x_j^\mu = 1 = y_i^\mu \\ -\varepsilon & \text{si } x_j^\mu = 0 \text{ y } y_i^\mu = 1 \\ 0 & \text{en otro caso} \end{cases}$$

Donde ε una constante positiva escogida previamente: es usual que ε sea igual a 1.

- *Fase de Recuperación*

La i -ésima coordenada y_i^ω del vector de clase $\mathbf{y}^\omega \in A^p$ se obtiene como lo indica la siguiente expresión, donde \vee es el operador *máximo*:

$$y_i^\omega = \begin{cases} 1 & \text{si } \sum_{j=1}^n m_{ij} \cdot x_j^\omega = \vee_{h=1}^p \left[\sum_{j=1}^n m_{hj} \cdot x_j^\omega \right] \\ 0 & \text{en otro caso} \end{cases}$$

2.2.2 *Correlograph de Willshaw, Buneman y Longuet-Higgins*

El *correlograph* es un dispositivo óptico elemental capaz de funcionar como una memoria asociativa. En palabras de los autores "el sistema es tan simple, que podría ser construido en cualquier laboratorio escolar de física elemental".

Algoritmo del Correlograph

- *Fase de Aprendizaje*

La *red asociativa* se genera al incorporar la pareja de patrones de entrenamiento $(\mathbf{x}^\mu, \mathbf{y}^\mu) \in A^n \times A^m$. Cada uno de los componentes m_{ij} de la *red asociativa* \mathbf{M} tiene valor cero al inicio, y se actualiza de acuerdo con la regla:

$$m_{ij} = \begin{cases} 1 & \text{si } y_i^\mu = 1 = x_j^\mu \\ \text{valor anterior} & \text{en otro caso} \end{cases}$$

- *Fase de Recuperación*

Se le presenta a la *red asociativa* \mathbf{M} un vector de entrada $\mathbf{x}^\omega \in A^n$. Se realiza el producto de la matriz \mathbf{M} por el vector \mathbf{x}^ω y se ejecuta una operación de umbral, de acuerdo con la siguiente expresión:

$$y_i^\omega = \begin{cases} 1 & \text{si } \sum_{j=1}^n m_{ij} \cdot x_j^\omega \geq u \\ 0 & \text{en otro caso} \end{cases}$$

Donde u es el valor de umbral. Una estimación aproximada del valor de umbral u se puede lograr con la ayuda de un número indicador mencionado en el artículo de Willshaw *et al.* de 1969: $\log_2 n$

2.2.3 *Linear Associator de Anderson-Kohonen*

El *Linear Associator* tiene su origen en los trabajos pioneros de 1972 publicados por Anderson y Kohonen.

Para presentar el *Linear Associator* consideremos de nuevo el conjunto fundamental:

$$\{(\mathbf{x}^\mu, \mathbf{y}^\mu) \mid \mu = 1, 2, \dots, p\} \text{ con } A = \{0, 1\}, \mathbf{x}^\mu \in A^n \text{ y } \mathbf{y}^\mu \in A^m$$

Algoritmo del Linear Associator

- *Fase de Aprendizaje*

1) Para cada una de las p asociaciones $(\mathbf{x}^\mu, \mathbf{y}^\mu)$ se encuentra la matriz $\mathbf{y}^\mu \cdot (\mathbf{x}^\mu)^t$ de dimensiones $m \times n$

2) Se suman la p matrices para obtener la memoria

$$\mathbf{M} = \sum_{\mu=1}^p \mathbf{y}^\mu \cdot (\mathbf{x}^\mu)^t = [m_{ij}]_{m \times n}$$

de manera que la ij -ésima componente de la memoria \mathbf{M} se expresa así:

$$m_{ij} = \sum_{\mu=1}^p y_i^\mu x_j^\mu$$

- *Fase de Recuperación*

Esta fase consiste en presentarle a la memoria un patrón de entrada \mathbf{x}^ω , donde $\omega \in \{1, 2, \dots, p\}$ y realizar la operación

$$\mathbf{M} \cdot \mathbf{x}^\omega = \left[\sum_{\mu=1}^p \mathbf{y}^\mu \cdot (\mathbf{x}^\mu)^t \right] \cdot \mathbf{x}^\omega$$

2.2.4 La memoria asociativa Hopfield

El artículo de John J. Hopfield de 1982, publicado por la prestigiosa y respetada *National Academy of Sciences* (en sus *Proceedings*), impactó positivamente y trajo a la palestra internacional su famosa memoria asociativa.

En el modelo que originalmente propuso Hopfield, cada neurona x_i tiene dos posibles estados, a la manera de las neuronas de McCulloch-Pitts: $x_i = 0$ y $x_i = 1$; sin embargo, Hopfield observa que, para un nivel dado de exactitud en la recuperación de patrones, la capacidad de almacenamiento de información de la memoria se puede incrementar por un factor de 2, si se escogen como posibles estados de las neuronas los valores $x_i = -1$ y $x_i = 1$ en lugar de los valores originales $x_i = 0$ y $x_i = 1$.

Al utilizar el conjunto $\{-1, 1\}$ y el valor de umbral cero, la fase de aprendizaje para la memoria Hopfield será similar, en cierta forma, a la fase de aprendizaje del *Linear Associator*. La intensidad de la fuerza de conexión de la neurona x_i a la neurona x_j se representa por el valor de m_{ij} , y se considera que hay simetría, es decir, $m_{ij} = m_{ji}$. Si x_i no

está conectada con x_j entonces $m_{ij} = 0$; en particular, no hay conexiones recurrentes de una neurona a sí misma, lo cual significa que $m_{ij} = 0$. El estado instantáneo del sistema está completamente especificado por el vector columna de dimensión n cuyas coordenadas son los valores de las n neuronas.

La memoria Hopfield es autoasociativa, simétrica, con ceros en la diagonal principal. En virtud de que la memoria es autoasociativa, el conjunto fundamental para la memoria Hopfield es $\{\mathbf{x}^\mu, \mathbf{x}^\mu \mid \mu = 1, 2, \dots, p\}$ con $\mathbf{x}^\mu \in A^n$ y $A = \{-1, 1\}$

Algoritmo Hopfield

- *Fase de Aprendizaje*

La fase de aprendizaje para la memoria Hopfield es similar a la fase de aprendizaje del *Linear Associator*, con una ligera diferencia relacionada con la diagonal principal en ceros, como se muestra en la siguiente regla para obtener la ij -ésima componente de la memoria Hopfield \mathbf{M} :

$$m_{ij} = \begin{cases} \sum_{\mu=1}^p x_i^\mu x_j^\mu & \text{si } i \neq j \\ 0 & \text{si } i = j \end{cases}$$

- *Fase de Recuperación*

Si se le presenta un patrón de entrada $\tilde{\mathbf{x}}$ a la memoria Hopfield, ésta cambiará su estado con el tiempo, de modo que cada neurona x_i ajuste su valor de acuerdo con el resultado que arroje la comparación de la cantidad $\sum_{j=1}^n m_{ij} x_j$ con un valor de umbral, el cual normalmente se coloca en cero.

Se representa el estado de la memoria Hopfield en el tiempo t por $\mathbf{x}(t)$; entonces $x_i(t)$ representa el valor de la neurona x_i en el tiempo t y $x_i(t+1)$ el valor de x_i en el tiempo siguiente ($t+1$).

Dado un vector columna de entrada $\tilde{\mathbf{x}}$, la fase de recuperación consta de tres pasos:

- 1) Para $t = 0$, se hace $\mathbf{x}(t) = \tilde{\mathbf{x}}$; es decir, $x_i(0) = \tilde{x}_i, \forall i \in \{1, 2, 3, \dots, n\}$
- 2) $\forall i \in \{1, 2, 3, \dots, n\}$ se calcula $x_i(t+1)$ de acuerdo con la condición siguiente:

$$x_i(t+1) = \begin{cases} 1 & \text{si } \sum_{j=1}^n m_{ij} x_j(t) > 0 \\ x_i(t) & \text{si } \sum_{j=1}^n m_{ij} x_j(t) = 0 \\ -1 & \text{si } \sum_{j=1}^n m_{ij} x_j(t) < 0 \end{cases}$$

- 3) Se compara $x_i(t+1)$ con $x_i(t) \forall i \in \{1, 2, 3, \dots, n\}$. Si $\mathbf{x}(t+1) = \mathbf{x}(t)$ el proceso termina y el vector recuperado es $\mathbf{x}(0) = \mathbf{x}$. De otro modo, el proceso continúa de la siguiente manera: los pasos 2 y 3 se iteran tantas veces como sea necesario hasta llegar a un valor $t = \tau$ para el cual $x_i(\tau+1) = x_i(\tau) \forall i \in \{1, 2, 3, \dots, n\}$; el proceso termina y el patrón recuperado es $\mathbf{x}(\tau)$.

En el artículo original de 1982, Hopfield había estimado empíricamente que su memoria tenía una capacidad de recuperar $0.15n$ patrones, y en el trabajo de Abu-Mostafa & St. Jacques se estableció formalmente que una cota superior para el número de vectores de estado arbitrarios estables en una memoria Hopfield es n .

2.3 Memorias Asociativas Morfológicas

La diferencia fundamental entre las memorias asociativas clásicas (*Lernmatrix*, *Correlograph*, *Linear Associator* y Memoria Asociativa Hopfield) y las memorias asociativas morfológicas radica en los fundamentos operacionales de éstas últimas, que son las operaciones morfológicas de *dilatación* y *erosión*; el nombre de las memorias asociativas morfológicas está inspirado precisamente en estas dos operaciones básicas. Estas memorias rompieron con el esquema utilizado a través de los años en los modelos de memorias asociativas clásicas, que utilizan operaciones convencionales entre vectores y matrices para la fase de aprendizaje y suma de productos para la recuperación de patrones. Las memorias asociativas morfológicas cambian los productos por sumas y las sumas por máximos o mínimos en ambas fases, tanto de aprendizaje como de recuperación.

Hay dos tipos de memorias asociativas morfológicas: las memorias *max*, simbolizadas con **M**, y las memorias *min*, cuyo símbolo es **W**; en cada uno de los dos tipos, las memorias pueden funcionar en ambos modos: heteroasociativo y autoasociativo.

Se definen dos nuevos productos matriciales:

El *producto máximo* entre **D** y **H**, denotado por $\mathbf{C} = \mathbf{D} \nabla \mathbf{H}$, es una matriz $[c_{ij}]_{m \times n}$ cuya *ij*-ésima componente c_{ij} es

$$c_{ij} = \bigvee_{k=1}^r (d_{ik} + h_{kj})$$

El *producto mínimo* de **D** y **H** denotado por $\mathbf{C} = \mathbf{D} \Delta \mathbf{H}$, es una matriz $[c_{ij}]_{m \times n}$ cuya *ij*-ésima componente c_{ij} es

$$c_{ij} = \bigwedge_{k=1}^r (d_{ik} + h_{kj})$$

Los productos máximo y mínimo contienen a los operadores máximo y mínimo, los cuales están íntimamente ligados con los conceptos de las dos operaciones básicas de la morfología matemática: *dilatación* y *erosión*, respectivamente.

2.3.1 Memorias Heteroasociativas Morfológicas

Algoritmo de las memorias morfológicas max

- *Fase de Aprendizaje*

Para cada una de las p asociaciones $(\mathbf{x}^\mu, \mathbf{y}^\mu)$ se usa el producto mínimo para crear la matriz $\mathbf{y}^\mu \Delta (-\mathbf{x}^\mu)^t$ de dimensiones $m \times n$, donde el negado transpuesto del patrón de entrada \mathbf{x}^μ se define como $(-\mathbf{x}^\mu)^t = (-x_1^\mu, -x_2^\mu, \dots, x_n^\mu)$.

Se aplica el operador máximo \bigvee a las p matrices para obtener la memoria **M**.

$$\mathbf{M} = \bigvee_{\mu=1}^p [\mathbf{y}^\mu \Delta (-\mathbf{x}^\mu)^t]$$

- *Fase de Recuperación*

Esta fase consiste en realizar el producto mínimo Δ de la memoria \mathbf{M} con el patrón de entrada \mathbf{x}^ω , donde $\omega \in \{1, 2, \dots, p\}$, para obtener un vector columna \mathbf{y} de dimensión m :

$$\mathbf{y} = \mathbf{M}\Delta\mathbf{x}^\omega$$

Las fases de aprendizaje y de recuperación de las **memorias morfológicas *min*** se obtienen por dualidad.

2.3.2 Memorias Autoasociativas Morfológicas

Para este tipo de memorias se utilizan los mismos algoritmos descritos anteriormente y que son aplicados a las memorias heteroasociativas; lo único que cambia es el conjunto fundamental. Para este caso, se considera el siguiente conjunto fundamental:

$$\{(\mathbf{x}^\mu, \mathbf{x}^\mu) \mid \mathbf{x}^\mu \in A^n, \text{ donde } \mu = 1, 2, \dots, p\}$$

2.3.3 Memorias Asociativas Alfa-Beta

Las memorias Alfa-Beta utilizan máximos y mínimos, y dos operaciones binarias originales α y β de las cuales heredan el nombre.

Para la definición de las operaciones binarias α y β se deben especificar los conjuntos A y B , los cuales son:

$$A = \{0, 1\} \quad \text{y} \quad B = \{0, 1, 2\}$$

La operación binaria $\alpha: A \times A \rightarrow B$ se define como:

x	y	$\alpha(x, y)$
0	0	1
0	1	0
1	0	2
1	1	1

Tabla 1. Operación Alfa

La operación binaria $\beta: B \times A \rightarrow A$ se define como:

x	y	$\beta(x, y)$
0	0	0
0	1	0
1	0	0
1	1	1
2	0	1
2	1	1

Tabla 2. Operación Beta

Los conjuntos A y B , las operaciones binarias α y β junto con los operadores \wedge (mínimo) y \vee (máximo) usuales conforman el sistema algebraico $(A, B, \alpha, \beta, \wedge, \vee)$ en el que están inmersas las memorias asociativas Alfa-Beta.

2.3.4 Memorias Asociativas Media

Las Memorias Asociativas Media utilizan los operadores A y B, definidos de la siguiente forma:

$$A(x, y) = x - y$$

$$B(x, y) = x + y$$

Las operaciones utilizadas se describen a continuación.

Sean $P = [p_{ij}]_{m \times r}$ y $Q = [q_{ij}]_{r \times n}$ dos matrices.

Operación \diamond_A : $P_{m \times r} \diamond_A Q_{r \times n} = [f_{ij}^A]_{m \times n}$ donde $f_{ij}^A = \mathbf{med}_{k=1}^r A(p_{ik}, q_{k,j})$

Operación \diamond_B : $P_{m \times r} \diamond_B Q_{r \times n} = [f_{ij}^B]_{m \times n}$ donde $f_{ij}^B = \mathbf{med}_{k=1}^r B(p_{ik}, q_{k,j})$

Algoritmo Memorias Media

- Fase de Recuperación

Paso 1. Para cada $\xi = 1, 2, \dots, p$, de cada pareja $(\mathbf{x}^\xi, \mathbf{y}^\xi)$ se construye la matriz:

$$[\mathbf{y}^\xi \diamond_A (\mathbf{x}^\xi)^t]_{m \times n}$$

Paso 2. Se aplica el operador media a las matrices obtenidas en el paso 1 para obtener la matriz \mathbf{M} , como sigue:

$$\mathbf{M} = \mathbf{med}_{\xi=1}^p \left[\mathbf{y}^\xi \diamond_A (\mathbf{x}^\xi)^t \right]$$

El ij -ésimo componente \mathbf{M} está dado como sigue:

$$m_{ij} = \mathbf{med}_{\xi=1}^p A(y_i^\xi, x_j^\xi)$$

- *Fase de Recuperación*

Se tienen dos casos:

Caso 1. Recuperación de un patrón fundamental. Un patrón \mathbf{x}^w , con $w \in \{1, 2, \dots, p\}$ se le presenta a la memoria \mathbf{M} y se realiza la siguiente operación:

$$\mathbf{M} \diamond_B \mathbf{x}^w$$

El resultado es un vector columna de dimensión n , con la i -ésima componente dada como:

$$(\mathbf{M} \diamond_B \mathbf{x}^w)_i = \mathbf{med}_{j=1}^n B(m_{ij}, x_j^w)$$

Caso 2. Recuperación de un patrón alterado. Un patrón $\tilde{\mathbf{x}}$, que es una versión alterada de un patrón \mathbf{x}^w , se le presenta a la memoria \mathbf{M} y se realiza la siguiente operación:

$$\mathbf{M} \diamond_B \tilde{\mathbf{x}}$$

De nuevo, el resultado es un vector de dimensión n , con la i -ésima componente dada como:

$$(\mathbf{M} \diamond_B \tilde{\mathbf{x}})_i = \mathbf{med}_{j=1}^n B(m_{ij}, \tilde{x}_j)$$

2.4 Señalamientos viales.

En México se han establecido normas de vialidad a través de la SCT (Secretaría de Comunicaciones y Transportes). En lo referente a la señales de vialidad, dichas normas se establecen con base en algunos protocolos establecidos por la ONU, algunas otras son tomados del sistema de vialidad de EUA; esto porque cuando empezaron a establecerse normas de vialidad en México se basaron en las normas estadounidenses, posteriormente la ONU estableció sus protocolos, pero no todo lo establecido allí aplicaba a las necesidades viales en México. Actualmente las normas mexicanas incluyen los lineamientos establecidos en la ONU y de los Estados Unidos con una doble finalidad: la adhesión al movimiento internacional para el facilitar el tránsito, independientemente del idioma, y lograr la uniformidad y efectividad de señales en toda la república mexicana.

A continuación veremos las características de las señales viales establecidos por las normas mexicanas que nos serán de utilidad para el propósito de nuestra tesis.

Las señales de vialidad se dividen en tres tipos:

Señales preventivas

Señales restrictivas

Señales informativas

De las anteriores solamente nos interesan las dos primeras, ya que son las señales directamente relacionadas con la seguridad y con las que estableceremos las acciones que se han de ejecutar por nuestra simulación.

2.4.1 Señales preventivas

Son tableros fijados en postes, con símbolos que tienen por objeto prevenir a los conductores de vehículos sobre la existencia de algún peligro en el camino y su naturaleza.

El tablero será cuadrado con las esquinas redondeadas y son colocadas con una diagonal en vertical. El radio para redondear las esquinas será de 4cm, quedando el radio interior para la curvatura del filete de 2 cm.

La siguiente tabla muestra las dimensiones de los tableros de las señales preventivas.

DIMENSIONES DEL TABLERO DE LAS SEÑALES PREVENTIVAS

Señal	U s o
Dimensiones cm	
61 x 61 (sin ceja)	En carreteras con ancho de corona menor de 6.00 m y calles urbanas.
71 x 71 (con ceja)	En carreteras con ancho de corona comprendido entre 6.00 y 9.00 m y avenidas principales urbanas.
86 x 86 (con ceja)	En carreteras con ancho de corona entre 9.00 y 12.00 m, vías rápidas urbanas y carreteras de cuatro carriles donde se puedan ubicar para el mismo sentido en ambos lados.
117 x 117 (con ceja)	En carreteras con cuatro carriles o más, con o sin separador central.

Tabla 3. Dimensiones del Tablero de las Señales Preventivas

Las señales preventivas se colocarán antes del riesgo que se trata señalar, a una distancia que depende de la velocidad. La siguiente tabla muestra esa relación.

UBICACION LONGITUDINAL DE LAS SEÑALES PREVENTIVAS

Velocidad* km/h	30	40	50	60	70	80	90	100	110
Distancia m	30	40	55	75	95	115	135	155	175

*En carreteras se utilizará la velocidad de proyecto; cuando se desconozca este dato, se utilizará la velocidad de marcha.

Tabla 4. Ubicación longitudinal de las Señales Preventivas

Las señales se fijarán en uno o dos postes colocados a un lado de la carretera o sobre la banqueta.

Se colocarán de modo que su orilla interior quede a una distancia de 50 cm de la proyección vertical del hombro del camino.

En todas las carreteras la parte inferior del tablero de las señales quedar a 1.50m sobre el hombro del camino y en las zonas urbanas a 2.0m sobre el nivel de la banqueta.

El tablero de las señales deberá quedar siempre en posición vertical a 90° con respecto al eje del camino.

El color de fondo de las señales preventivas será amarillo tránsito. El color para los símbolos y caracteres y filete será negro.

Ahora se describe en qué consiste cada señal preventiva.

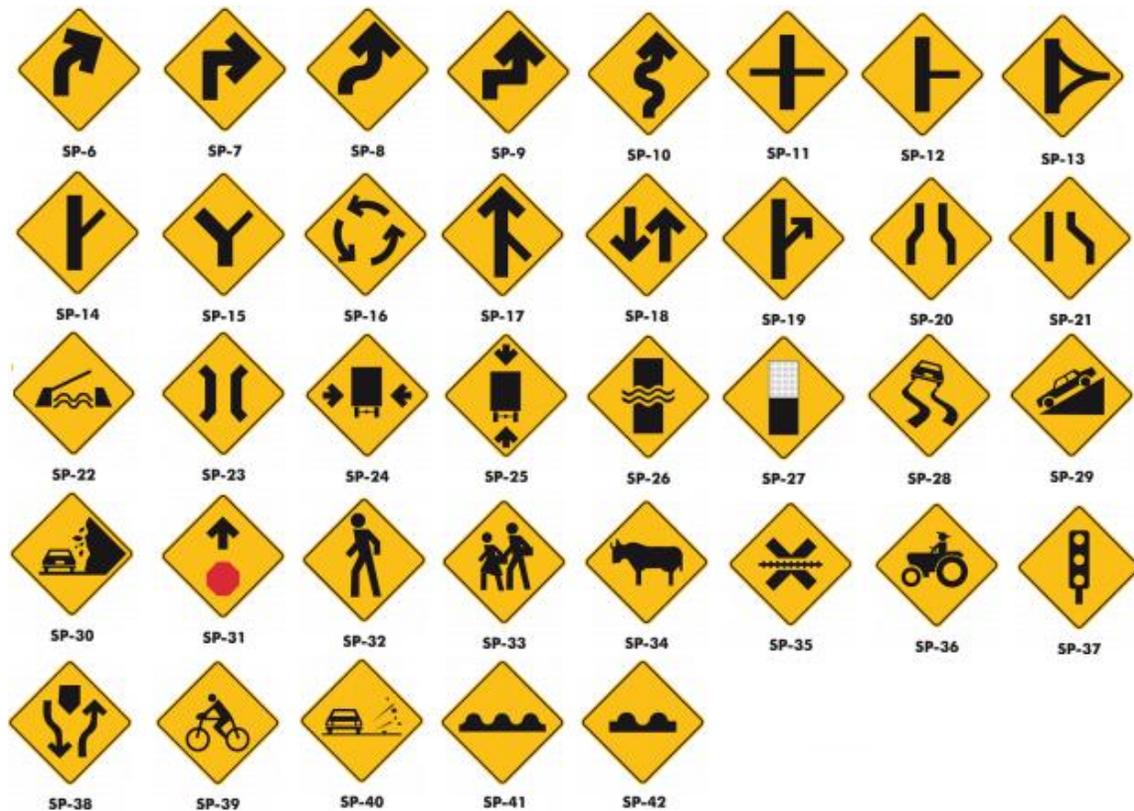


Figura 6. Señales Preventivas

SP-06 Curva
 SP-07 Codo
 SP-08 Curva Inversa
 SP-09 Codo Inverso
 SP-10 Camino Sinuoso
 SP-11 Cruce de Caminos
 SP-12 Entronque en T
 SP-13 Entronque en Delta
 SP-14 Entronque Lateral Oblicuo
 SP-15 Entronque en Y
 SP-16 Glorieta
 SP-17 Incorporación del Tránsito
 SP-18 Doble Circulación

SP-19 Salida
 SP-20 Estrechamiento Simétrico
 SP-21 Estrechamiento Asimétrico
 SP-22 Puente Móvil
 SP-23 Puente Angosto
 SP-24 Anchura Libre
 SP-25 Altura Libre
 SP-26 Vado
 SP-27 Termina Pavimento
 SP-28 Superficie Derrapante
 SP-29 Pendiente Peligrosa
 SP-30 Zona de Derrumbes
 SP-31 Alto Próximo

SP-32 Peatones
 SP-33 Escolares
 SP-34 Ganado
 SP-35 Cruce de Ferrocarril
 SP-36 Maquinaria Agrícola
 SP-37 Semáforo
 SP-38 Camino Dividido
 SP-38 B Camino Dividido
 SP-39 Ciclistas
 SP-40 Grava Suelta
 SP-41 Vibrador
 SP-42 Tope

2.4.2 Señales restrictivas

Son tableros fijados en postes, con símbolos y /o leyendas que tienen por objeto indicar al usuario, tanto en zona rural como urbana, la existencia de limitaciones físicas o prohibiciones reglamentarias que regulan al tránsito.

El tablero de las señales restrictivas será de forma cuadrada con las esquinas redondeadas, excepto las de “ALTO” y “CEDA EL PASO”.

Las señales que requieren una explicación complementaria, además del símbolo, llevarán un tablero adicional de forma rectangular con las esquinas redondeadas para formar un conjunto

El radio para redondear las esquinas será de 4cm quedando el radio interior para la curvatura del filete de 2 cm.

La señal de “ALTO” tendrá la forma octagonal, la señal “CEDA EL PASO” tendrá la forma de un triángulo equilátero, con un vértice hacia abajo.

La siguiente tabla muestra las dimensiones del tablero de las señales restrictivas.

DIMENSIONES DEL TABLERO DE LAS SEÑALES RESTRICTIVAS

Señal	U s o
Dimensiones cm	
61 x 61 (sin ceja)	En carreteras con ancho de corona menor de 6.00 m y calles urbanas.
71 x 71 (con ceja)	En carreteras con ancho de corona comprendido entre 6.00 y 9.00 m y avenidas principales urbanas.
86 x 86 (con ceja)	En carreteras con ancho de corona entre 9.00 y 12.00 m, vías rápidas urbanas y carreteras de cuatro carriles donde se puedan ubicar para el mismo sentido en ambos lados.
117 x 117 (con ceja)	En carreteras con cuatro carriles o más, con o sin separador central.
Alto 25 por lado (sin ceja)	En carreteras con ancho de corona hasta 9.00 m y en calles urbanas.
Alto 30 por lado (con ceja)	En carreteras con ancho de corona mayor de 9.00 m y avenidas principales urbanas.
Ceda el paso 70 x 70 x 70 (sin ceja)	En carreteras con ancho de corona hasta 9.00 m y en calles urbanas.
Ceda el paso 85 x 85 x 85 (con ceja)	En carreteras con ancho de corona mayor de 9.00 m y avenidas principales urbanas.

Tabla 5. Dimensiones del tablero de las Señales Restrictivas

DIMENSIONES DEL TABLERO ADICIONAL DE LAS SEÑALES RESTRINGIDAS

Dimensiones de la señal cm	Dimensiones del tablero cm		Altura de las letras mayúsculas cm	
	1 renglón	2 renglones	1 renglón	2 renglones
61 x 61 (sin ceja)	25 x 61	40 x 61	10	10
71 x 71 (con ceja)	30 x 71	50 x 71	12.5	12.5
86 x 86 (con ceja)	35 x 86	61 x 86	15	15
117 x 117 (con ceja)	35 x 117	61 x 117	15	15

Tabla 6. Dimensiones del tablero adicional de las señales restrictivas

Las señales restrictivas se colocarán en el punto mismo donde existe la restricción o de prohibición.

Las señales se fijarán en uno o dos postes colocados a un lado de la carretera sobre la banqueta.

En carreteras, la señal se colocará en todos los casos, de modo que su orilla interior quede a una distancia no menor de 50cm de la proyección del hombro del camino.

En carreteras, la parte inferior de la señal quedará a 1.50m sobre el hombro del camino y en las zonas urbanas a 2.00m sobre el nivel de la banqueta.

El tablero de las señales deberá quedar siempre en posición vertical a 90° con respecto al eje del camino.

El color del fondo de las señales restrictivas será blanco en acabado reflejante, excepto en las correspondientes a los caminos con corona menor de 6.00m que será en acabado mate. El anillo y la franja diametral serán en rojo según el patrón aprobado en ese manual, y el símbolo, letras y filete serán en negro, excepto las señales "ALTO" y "CEDA EL PASO".

El color del tablero adicional será de fondo blanco reflejante, con letras y filete en negro.

La señal de "ALTO" llevará fondo rojo con letras y filete en blanco. En acabado reflejante.

La señal "CEDA EL PASO" llevará fondo blanco en acabado reflejante, franja perimetral roja y leyenda en negro.



Figura 7. Señales Restrictivas

- | | |
|---------------------------------|---|
| SR-06 Alto | SR-21 Estacionamiento |
| SR-07 Ceda el Paso | SR-22 Prohibido Estacionarse |
| SR-08 Inspección | SR-23 Prohibida la Vuelta a la Derecha |
| SR-09 Velocidad | SR-24 Prohibida la Vuelta a la Izquierda |
| SR-10 Vuelta Continua | SR-25 Prohibido el Retorno |
| SR-11 Circulación | SR-26 Prohibido Seguir de Frente |
| SR-11 B Circulación | SR-27 Prohibido el Paso a Bicicletas, Vehículos y Motos |
| SR-12 Sólo Vuelta Izquierda | SR-28 Prohibido el Paso de Vehículos de Tracción Animal |
| SR-13 Conserve su Derecha | SR-29 Prohibido el Paso a Maquinaria Agrícola |
| SR-14 Doble Circulación | SR-30 Prohibido el Paso a Bicicletas |
| SR-15 Altura Libre Restringida | SR-31 Prohibido el Paso a Peatones |
| SR-16 Anchura Libre Restringida | SR-32 Prohibido el Paso de Vehículos Pesados |
| SR-17 Peso Restringido | SR-33 Prohibido el Uso de Señales Acústicas |
| SR-18 Prohibido Rebasar | SR-34 Uso Obligatorio del Cinturón de Seguridad |
| SR-19 Parada Prohibida | |
| SR-20 No Parar | |

De este tipo de señalamientos las que más nos interesa son las restrictivas de velocidad.

2.4.3 Señales restrictivas de velocidad.

Se utilizarán para indicar el límite máximo de la velocidad que se fije en el tramo de la carretera vía urbana, el cual será expresado en múltiplos de 10 con la abreviatura km/h, con excepción de la velocidad límite de 95km/h que se usan para autobuses.

Como 95 y 80km/h son los límites máximos de velocidad permisibles para autobuses y camiones respectivamente, en los casos en que la velocidad de proyecto sea mayor a estas, será necesario colocar una señal de velocidad por cada tipo de vehículos, según el caso, para fijar el límite máximo a cada uno de ellos. Deberá emplearse un tablero adicional que indique el tipo de vehículo.

El límite máximo de velocidad para vehículos que no sean camiones y autobuses es de 110km/h.

El límite máximo de velocidad en el D.F y Estado de México en zonas donde se ubiquen iglesias, escuelas, oficinas públicas, centros deportivos, hospitales y lugares de reunión es de 20Km/h; por ello, se toma un rango de velocidades que va de 20Km/ hasta 110Km/h en múltiplos de 10.

CAPÍTULO III:

DESARROLLO DE LA SIMULACIÓN

3.1 Manipulación de imágenes por software

Las señales viales a utilizar en este software son antes que nada imágenes digitales, como tales, para poder aplicarles cualquier algoritmo con base matemática se deben manejar como matrices.

En la actualidad el modelo de color más utilizado en imágenes digitales es el RGB (Red Blue Green), el cual desde el punto de vista matemático consta de 3 matrices diferentes que representan los valores píxel a píxel de cada uno de los colores primarios de dicho sistema (rojo, verde, azul).

Para reducir la complejidad y el tiempo de proceso de las operaciones que se efectuaran sobre las imágenes estas se homogeneizaran en tamaño adaptándolas a una resolución mínima que conserve sus rasgos esenciales pero que a la vez permita su manipulación rápida, y por lo tanto consuma una cantidad mínima de recursos del sistema. Posteriormente se convertirán a escala de grises para reducir el número de matrices por imagen de tres a una.

3.2 Implementación y pruebas de las memorias asociativas morfológicas en MATLAB

En primera instancia se consideró el entorno de desarrollo MATLAB y su lenguaje M propiedad de Mathworks para el desarrollo del software. En él se implementaron los algoritmos respectivos tanto de la fase de aprendizaje como de la de recuperación.

Además este entorno de desarrollo tiene funciones preestablecidas para el cambio de tamaño de la imagen y la transformación de RGB a escala de grises, se utilizaron las funciones **imresize()** y **rgb2gray()** respectivamente.

Se crearon las funciones respectivas para la creación de las matrices máxima y mínima correspondientes a la fase de recuperación así como la función de recuperación.

Esta primera fase demostró el correcto funcionamiento de las memorias asociativas sin embargo MATLAB se mostró como una opción muy limitada en cuanto a la compatibilidad y el consumo de recursos pero más importante aún en la creación de la interfaz gráfica que representa el tablero de control.

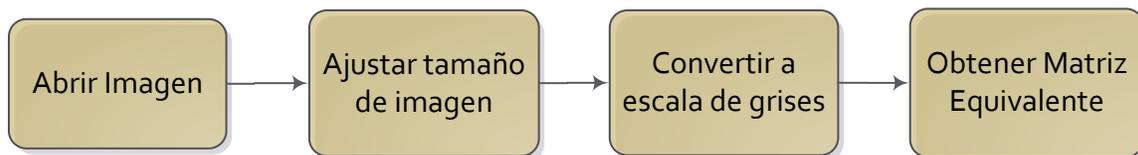
3.3 Implementación en lenguaje C# utilizando Visual Studio 2012

En este punto Visual Studio, en concreto su lenguaje C# se mostró como la opción más balanceada para desarrollar el simulador. Para esto se dividió el sistema en tres etapas:

- Ingreso y manipulación de imágenes
- Implementación y aplicación de los algoritmos de las memorias asociativas morfológicas
- Interfaz gráfica de usuario

3.4 Ingreso y manipulación de imágenes en C#

Esta etapa se subdividió de la siguiente forma:



Para la manipulación de imágenes en C# es imprescindible el *namespace* **System.Drawing** de forma especial la estructura **Color** y la clase **Bitmap**.

3.4.1 Abrir imagen

El proceso de abrir la imagen define el cargar la imagen por primera vez al programa. Consta de abrir la imagen desde el **Load** del Form principal de la interfaz, lo cual permite que las imágenes de las señales principales sean cargadas al ejecutar el programa, de una manera totalmente transparente al usuario. Como inconveniente para agregar más imágenes es que se realiza desde el código por lo cual solo los desarrolladores pueden modificarla.

Se utilizara un objeto del tipo bitmap que permitirá la obtención de sus propiedades.

3.4.2 Ajustar tamaño de imagen

Para el caso de ajustar el tamaño de la imagen se realizó desde una sobrecarga ya definida del constructor propio de la clase **bitmap** el cual permite que desde sus parámetros se le pasen los datos numéricos de la altura y el ancho deseado para el objeto a crearse.

3.4.3 Convertir a escala de grises

Color y **Bitmap** se utilizaron conjuntamente para la obtención de los valores RGB de la imagen y la posterior conversión a escala de grises.

Una vez obtenidos los valores RGB de cada píxel se convierten a gris mediante la **Formula de Conversión estándar NTSC** la cual se obtiene de la siguiente forma:

$$\text{Gris} = 0.2989 * \text{rojo} + 0.5870 * \text{verde} + 0.1140 * \text{azul}$$

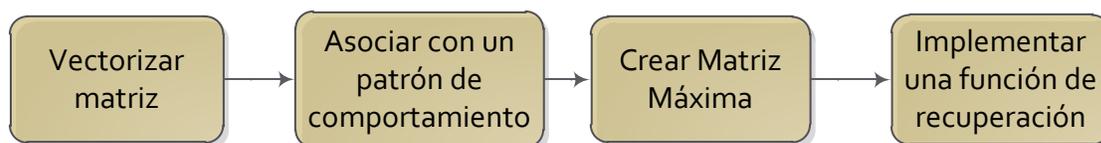
Una vez obtenidos estos valores se asignan nuevamente a un objeto bitmap que será almacenado para su posterior utilización.

3.4.4 Obtener matriz equivalente

Al mismo tiempo que se obtienen los valores NTSC estándar de gris y se crea el objeto **bitmap** cada valor se almacena en una matriz numérica equivalente.

3.5 Implementación y aplicación de los algoritmos de las memorias asociativas morfológicas

Esta etapa consiste en lo siguiente:



3.5.1 Vectorizar matriz

Como se ha mencionado las Memorias Asociativas Morfológicas toman como base dos vectores, creando la asociación mediante las matrices máxima y mínima, para el simulador uno de esos vectores serán los valores de la matriz equivalente dispuestos de tal forma que en vez de formar parte de una matriz lo hagan de un vector. Para ejemplificar esto tenemos el siguiente ejemplo:

$$\begin{bmatrix} 1 & 4 & 6 \\ 3 & 2 & 0 \\ 8 & 7 & 5 \end{bmatrix} \rightarrow [1 \ 4 \ 6 \ 3 \ 2 \ 0 \ 8 \ 7 \ 5]$$

3.5.2 Asociar con patrón de comportamiento

Ahora bien, dado que cada vector obtenido en la fase anterior representa una señal de vialidad el patrón asociado es el que determinara el comportamiento que deberá seguir el sistema.

Para determinar que comportamiento deberá seguir el sistema en cada señal estas se dividieron en dos grupos:

Las señales restrictivas, en especial aquellas que especifican explícitamente los límites de velocidad serán las que tendrán asociado un patrón de comportamiento que ajustara la velocidad. Esto significa que cada señal dentro de este grupo tendrá un patrón distinto.

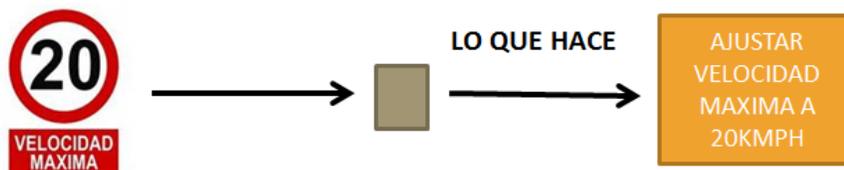


Figura 8. Patrón de comportamiento (Señales Restrictivas)

Las señales preventivas, por otra parte, tendrán un comportamiento especial, el cual consistirá en mostrar una advertencia en un lugar destinado para ello. El patrón para cada una de estas imágenes será el mismo.

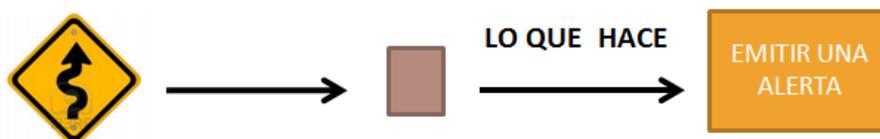


Figura 9. Patrón de Comportamiento (Señales Preventivas)

3.5.2.1 Patrones de comportamiento

Los patrones de comportamiento son vectores de la forma $[x,0,0,0,0,0,\dots,0]$ conteniendo un valor mayor a cero (x) y todos los demás elementos nulos; el valor de la x debe ser mayor a 255, la razón de esto se encuentra en el capítulo de pruebas. En cuanto el algoritmo clasifica una imagen en un determinado vector, la posición de la x indica, con base en un vector de referencia, que debe hacer el sistema. Dicho vector contiene los comportamientos disponibles para las diferentes acciones que realiza el simulador. Por ejemplo:

Se tiene el patrón de comportamiento $[0,0,0,x,0,0,0,\dots,0]$ y el vector de referencia es $[a,b,c,d,e,\dots]$; como se puede observar el valor “ x ” está en la cuarta posición, y en el vector de acciones la cuarta posición la ocupa la acción “ d ”, esto quiere decir que la acción a seguir por el simulador respecto a la señal reconocida es “ d ”.

Acciones que realiza el sistema

Las acciones que puede llevar a cabo el simulador se pueden clasificar en tres tipos principales:

- **Reducir la velocidad:** Esta es la acción más directa de todas ya que consiste en actuar directamente sobre la velocidad, esta acción está asociada a las señales restrictivas que indican explícitamente el límite de velocidad, cada límite de velocidad tiene un elemento en el vector de referencia, lo que aunado al hecho de que los límites de velocidad van desde 20Km/h hasta 110km/h, hace que sean 10 valores diferentes, cada uno en una posición diferente del vector de referencia, las señales con este comportamiento están en la figura .



Figura 10. Señales reductoras de velocidad

- **Emitir una Alerta Preventiva:** La alerta preventiva consiste en una alerta que se despliega y muestra el letrero de prevención, está asociada a las señales tanto preventivas y restrictivas que no requieren necesariamente de la variación de la velocidad pero si de la atención del conductor. Al ser el mismo comportamiento se tiene un único valor en el vector de acciones, y todas las señales que se engloban en esta categoría tienen el mismo patrón de comportamiento asociado.



Figura 11. Señales con Alerta de Precaución

- **Emitir una Alerta de Reducción de Velocidad:** Esta acción está asociada a señales preventivas y restrictivas que si bien requieren una reducción de velocidad no existe forma de determinar cuál es la velocidad ideal a la que debería ir el vehículo, por lo que simplemente se emitirá una alerta advirtiendo al conductor que deberá reducir su velocidad próximamente. Las señales que provocan este comportamiento son las que se muestran en la figura.



Figura 12. Señales con Alerta de Reducción de velocidad

3.5.3 Vector de referencia

El vector de referencia contiene los diferentes comportamientos correspondientes a cada patrón, como se mencionó en el punto anterior, el simulador utiliza 12 patrones de comportamiento, lo que da como resultado que el vector de referencia debe tener cuando menos 12 elementos. Los valores de los elementos del vector de referencia se muestran en las tablas.

Acción a realizar	Reducir a 20 km/h	Reducir a 30 km/h	Reducir a 40 km/h	Reducir a 50 km/h	Reducir a 60 km/h	Reducir a 70 km/h
Valore en el vector de Referencia	20	30	40	50	60	70

Tabla 7. Valores de los elementos del vector de referencia

Acción a realizar	Reducir a 80 km/h	Reducir a 90 km/h	Reducir a 100 km/h	Reducir a 110 km/h	Mostrar alerta de precaución	Mostrar alerta de reducir velocidad
Valore en el vector de Referencia	80	90	100	110	200	230

Tabla 8. Valores de los elementos del vector de referencia

Los valores para las alertas se seleccionaron que fueran por encima de 110 para que no se confundieran con los valores posibles de velocidad.

3.5.4 Creación de la matriz máxima

Una vez teniendo el vector de la imagen y el vector del patrón, se asocian de la forma mencionada en el Capítulo 2, lo cual crea una matriz de asociación. Cada imagen tiene una matriz de asociación y todas las matrices de asociación son del mismo tamaño. De estas matrices se obtiene la matriz Máxima. En la figura siguiente se observa el diagrama de flujo de la matriz máxima.

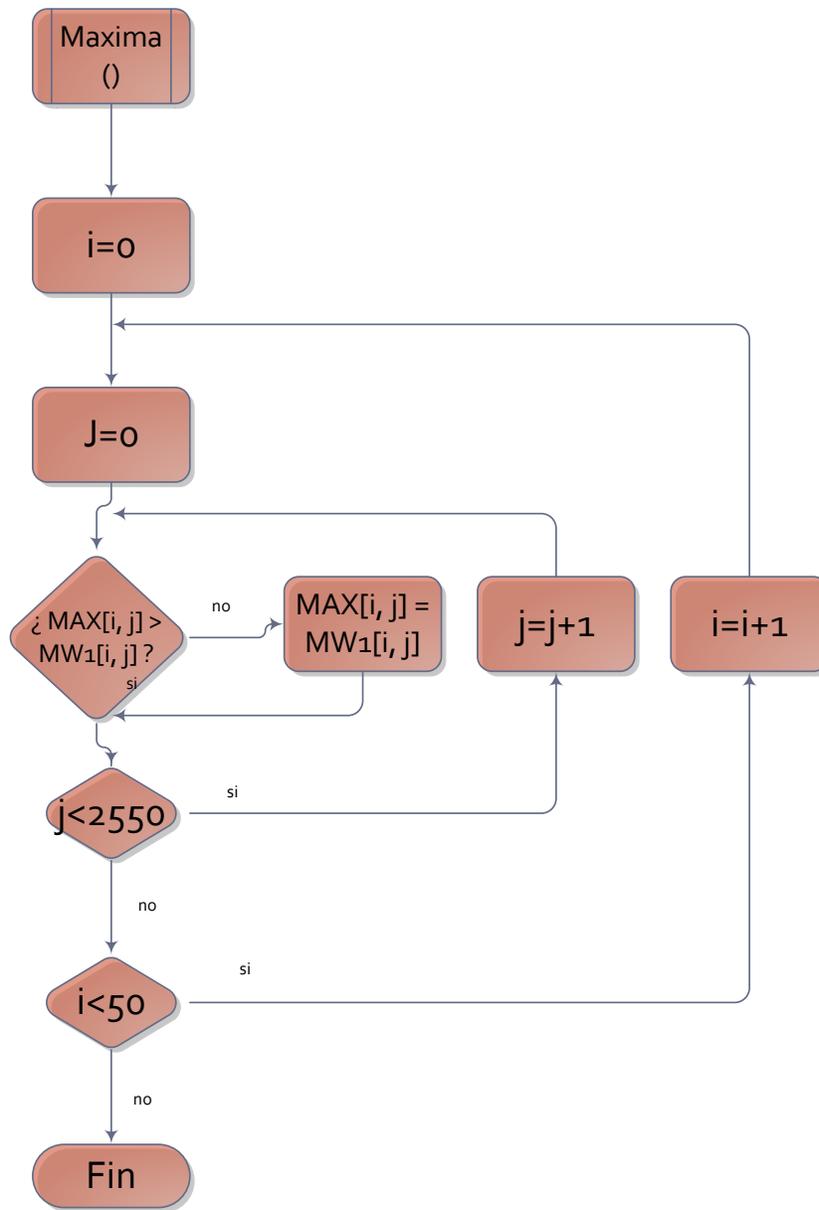


Figura 13. Diagrama de flujo de la Matriz Máxima

3.5.5 Implementar una función de recuperación

Ya que se han obtenido las matrices Máxima y Mínima se tienen que recuperar los patrones de comportamiento dependiendo la imagen que se le pase a la función de recuperación. Para esto dicha función aplicara los algoritmos de recuperación tanto a la matriz máxima como a la mínima para minimizar la probabilidad de error.

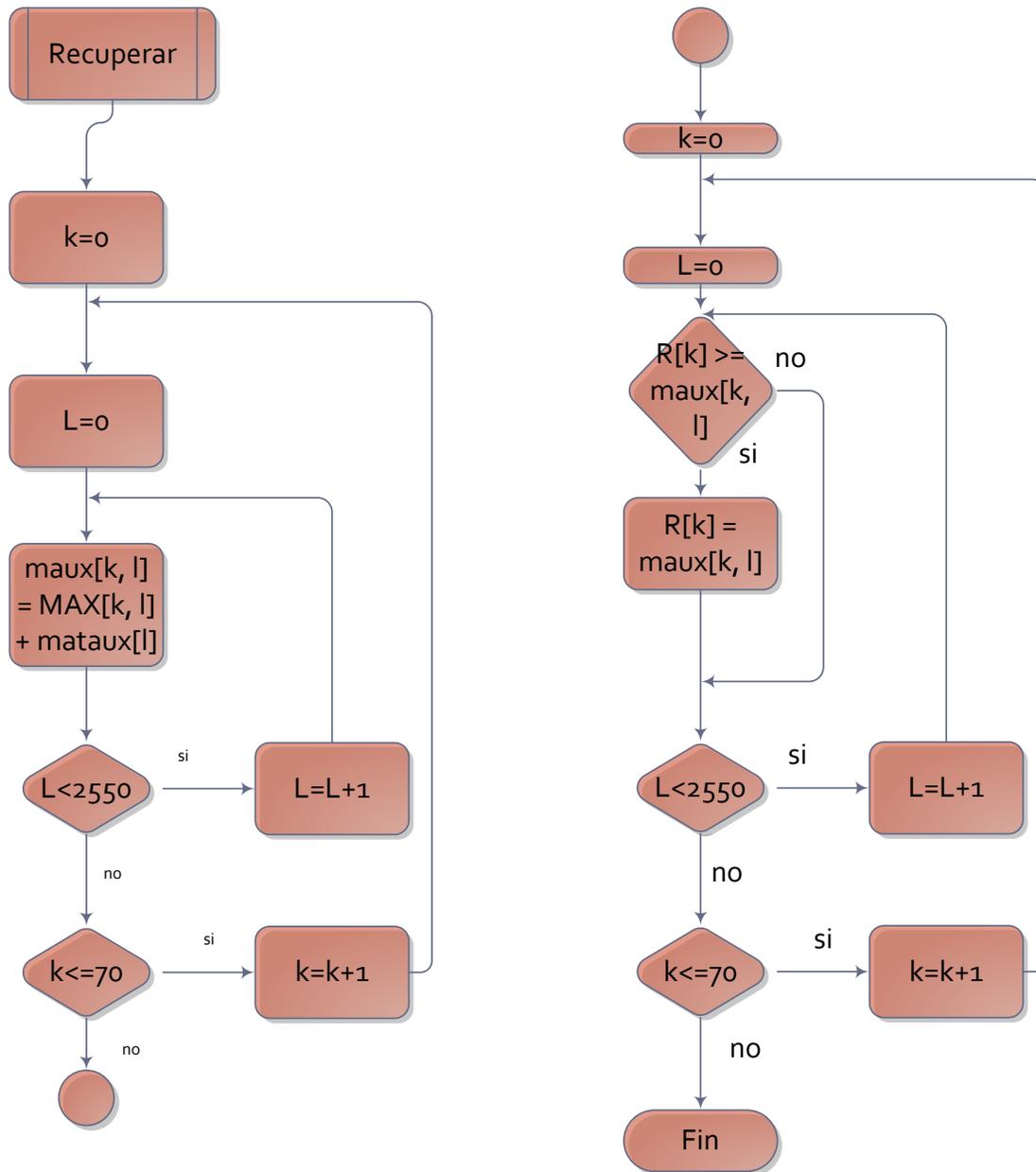


Figura 14. Diagrama de Flujo de la función de Recuperación

3.6 Interfaz Gráfica de usuario

La interfaz gráfica en Visual C# se implementó en tres etapas.

3.6.1 Primera etapa

La primera etapa consistió en una interfaz bastante básica la cual solamente servía para mostrar los resultados *en bruto* del algoritmo de asociación y de recuperación esta interfaz se creó teniendo en mente el objetivo de comprobar la eficacia de la implementación del algoritmo y solo mostraba una imagen y su patrón recuperado.



Figura 15. Interfaz primera etapa

3.6.2 Segunda etapa

Esta segunda etapa consistió en añadir más imágenes y ponerlas disponibles para observar el comportamiento que se obtenía con cada una. Desde el punto de vista de controles se observa en la figura que cuenta con 20 pictureboxes que se encuentran del lado izquierdo de la ventana (cuadro amarillo), así como el velocímetro del lado derecho (círculo verde), además del cuadro de alerta (cuadro morado) y los botones de aceleración y freno (cuadro azul) y los correspondientes al algoritmo de memorias morfológicas (círculo rojo).

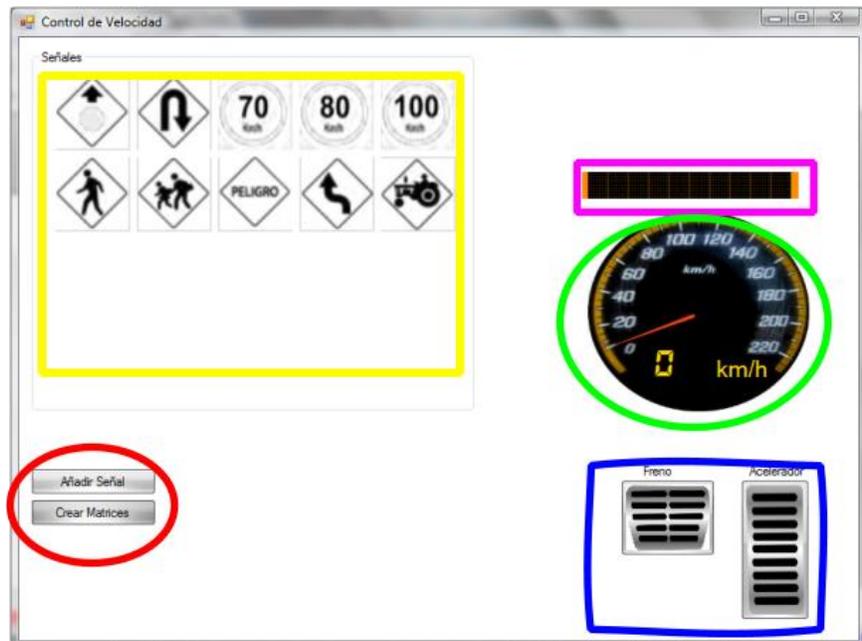


Figura 16. Interfaz Segunda Etapa

3.6.3 Tercera etapa

La tercera etapa de la interfaz es la definitiva. Al ser una interfaz interactiva de tipo gráfico, no sigue un comportamiento lineal, sino que su comportamiento está enfocado a los eventos que ocurren en esta. Los controles generadores de eventos son los temporizadores y los botones.

3.6.3.1 Eventos condicionados por los temporizadores o timers

El simulador cuenta con cinco temporizadores principales, estos son:

Temporizador de Imagen Aleatoria: Como su nombre lo indica este temporizador es el responsable de mostrar las imágenes de una señal aleatoria en la interfaz, y al mismo tiempo pasara los datos de la matriz equivalente al algoritmo de recuperación para su correcta detección. Se utilizan señales aleatorias para que el comportamiento del simulador sea como si de un camino desconocido se tratase.

Temporizador de Precaución: Es el temporizador encargado de mostrar la alerta en caso de que se haya encontrado con una señal que así lo requiera, alternará entre el letrero encendido y apagado para dar un efecto de parpadeo.

Temporizadores de Aceleración y Frenado: Están encargados de dar un comportamiento realista a la visualización del velocímetro para que los cambios de velocidad se observen de forma gradual y no de una forma abrupta, totalmente irreal.

Temporizador de Control: Es la parte central del proceso de control, este se activa una vez que el algoritmo le ha pasado el patrón de velocidad a seguir y es el encargado de activar y desactivar el temporizador de freno o el de precaución según corresponda.

3.6.3.2 Eventos condicionados por botones y la barra deslizante

Hay tres botones interactivos principales, estos son:

Botón de Inicio/Alto: Es el encargado de iniciar o detener la secuencia de señales que se mandaran, activa o desactiva el temporizador de imagen aleatoria.

Botón de Aceleración: Al mantenerse oprimido activa el temporizador de aceleración lo que se traduce en un incremento de velocidad, al mismo tiempo desactiva el temporizador de control y los temporizadores que este pueda estar utilizando, ya sea el de precaución o el de frenado. En cuanto se deja de presionar desactiva el temporizador de aceleración.

Botón de Frenado: Al mantenerse presionado activa el temporizador de frenado y desactiva los temporizadores de control y precaución. Al dejar de presionarse desactiva el temporizador de frenado.

Cabe añadirse que la acción de desactivar los temporizadores de control al oprimirse tanto el botón de aceleración como el de frenado responde a que la responsabilidad del vehículo es por encima de todo del conductor así que esta acción se interpreta como el deseo del conductor por obtener el control del vehículo.

Barra deslizante: permite controlar el tiempo en que aparece un señalamiento vial; por lo que condiciona al temporizador de imagen aleatoria según la posición en que se encuentre. Contiene 10 posiciones, cada una representa el número de segundos que transcurrirán para mantener visible una señal vial.

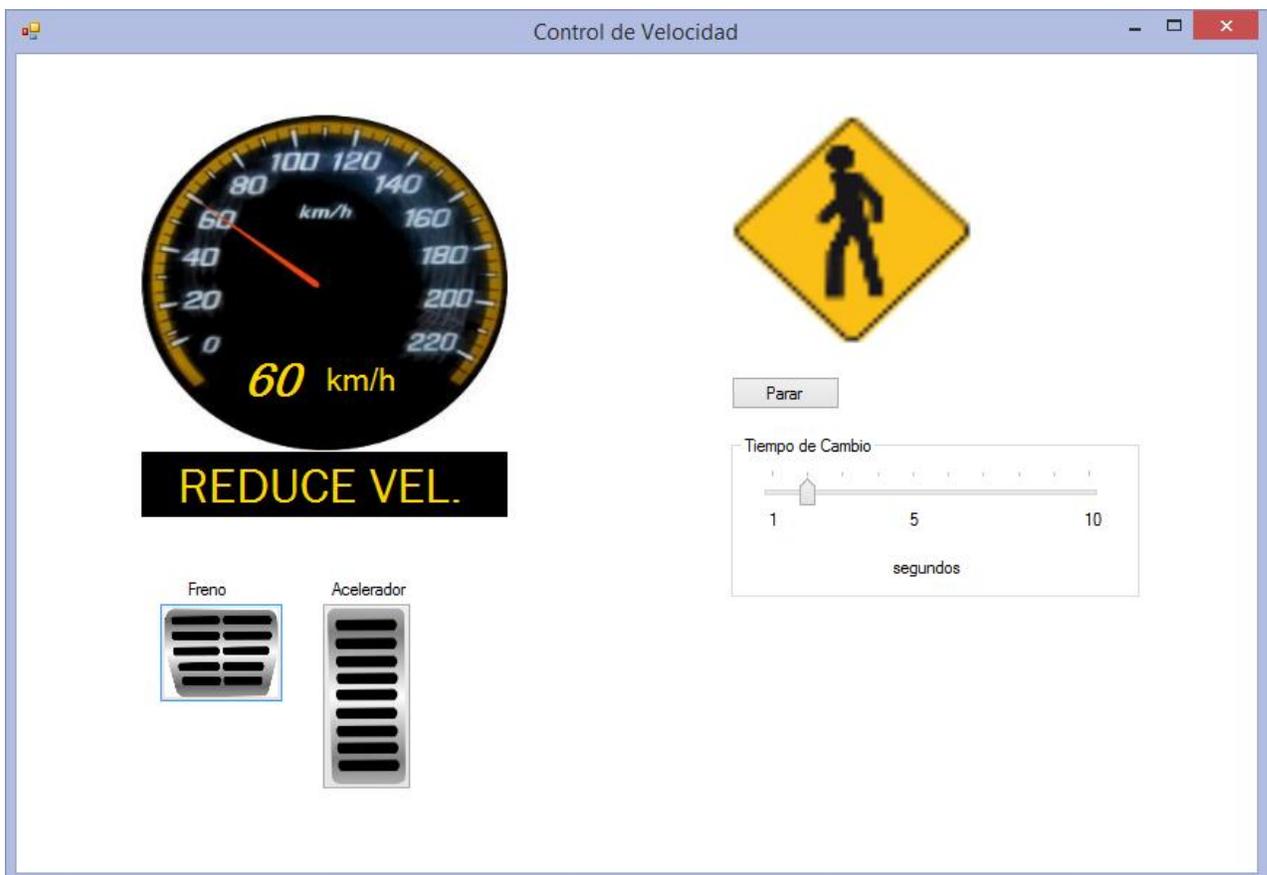


Figura 17. Interfaz Tercera Etapa

En la figura se observa que la interfaz se simplificó y ahora solo cuenta con un picturebox que es el que muestra la imagen seleccionada por el temporizador de imagen aleatoria y que es el cual está tomando el sistema como base para actuar.

3.7 Conjunción del sistema

Una vez que se tuvieron todas las partes que conforman al sistema, hubo que conjuntarlas para que funcionaran de forma correcta, para esto los tres algoritmos que se diseñaron y programaron en los puntos anteriores se conjuntaron.

Los primeros dos puntos que son la manipulación de imágenes y el algoritmo de memorias asociativas se incluyeron en una clase llamada “morfológica” y cuyo diagrama de clases vemos en la siguiente figura.

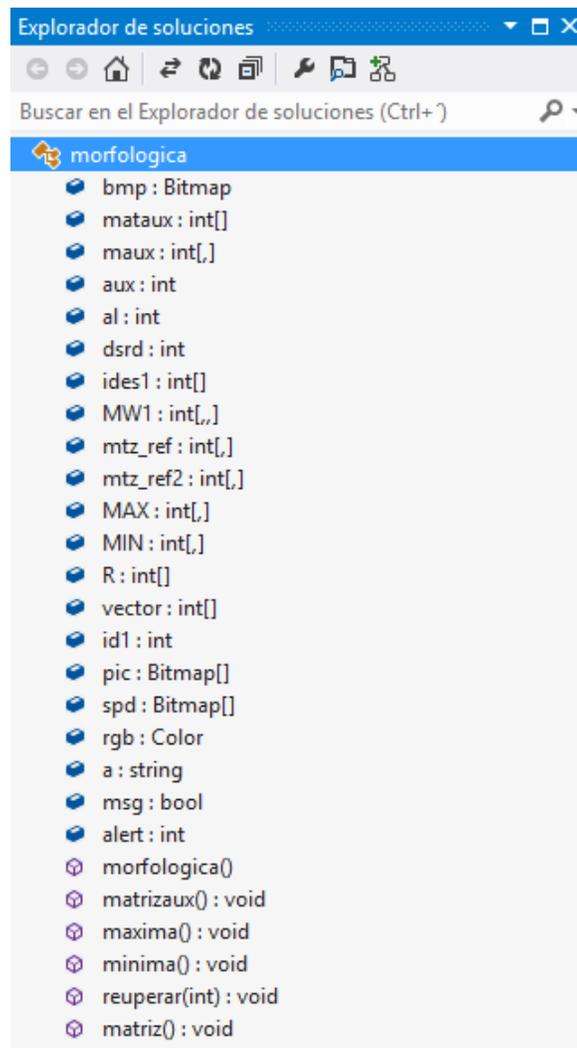


Figura 18. Clase Morfológica

CAPÍTULO IV:

PRUEBAS Y RESULTADOS

patrón de comportamiento recuperado. Estos valores son enteros positivos mayores que cero.

El primer valor propuesto fue el valor de 50 y se fue incrementando en 50 hasta 200, después se seleccionó 255 por ser el valor más alto que puede tener un píxel de una imagen en escala de grises, y finalmente se hicieron pruebas con valores mayores a 255 (300 y 500) los resultados pueden verse en la tabla 9.

Valor de X	Imágenes Totales	Imágenes Bien Clasificadas	Imágenes Mal Clasificadas
50	50	38~40	10~12
100	50	40~42	8~10
150	50	43~44	6~7
200	50	45~47	3~5
255	50	50	0
>255	50	50	0

Tabla 9. Pruebas de Recuperación para 50 imágenes

También se hicieron pruebas incrementando el número de imágenes hasta 75.

Valor de X	Imágenes Totales	Imágenes Bien Clasificadas	Imágenes Mal Clasificadas
50	75	62~65	10~13
100	75	64~66	9~11
150	75	65~68	7~10
200	75	68~69	6~7
255	75	75	0
>255	75	75	0

Tabla 10. Pruebas de Recuperación para 75 imágenes

Las pruebas se hicieron con 3 conjuntos de imágenes distintos, cada uno con diferentes imágenes.

Como puede observarse al incrementarse el número de imágenes la incidencia de errores se incrementa. Esto se debe a que al incrementar la cantidad de imágenes se incrementa el número de veces que debe calcular la matriz máxima, y esto a su vez ocasiona que los valores de la matriz se sobrescriban múltiples veces, y que probablemente se pierda la información referente a una imagen en particular.

Adicionalmente se hicieron pruebas variando el tamaño del vector de acción, en la tabla se observan las pruebas para un vector totalmente ocupado, es decir, con 11 elementos, en el que cada uno de sus valores representa un comportamiento y por lo tanto el vector no tiene ningún valor de cero.

Valor de X	Imágenes Totales	Imágenes Bien Clasificadas	Imágenes Mal Clasificadas
50	50	43~45	5~7
100	50	45~46	4~5
150	50	45~48	2~5
200	50	48~49	1~2
255	50	50	0
>255	50	50	0

Tabla 11. Pruebas para un patrón de comportamiento de 12 elementos

También se realizaron pruebas a un patrón con un número de elementos mucho mayor, en este caso con 100 elementos, todo esto con el fin de encontrar la opción que mejor se adaptara a las necesidades del programa.

Valor de X	Imágenes Totales	Imágenes Bien Clasificadas	Imágenes Mal Clasificadas
50	50	40~43	7~10
100	50	44~45	5~6
150	50	45~46	4~5
200	50	46~48	2~4
255	50	50	0
>255	50	50	0

Tabla 12. Pruebas para un patrón de 100 elementos

4.2 Resultados

4.2.1 resultados de las pruebas al algoritmo

Las pruebas del algoritmo demostraron que para el número de imágenes que se utilizaron (47), las mejores opciones para el valor diferente de cero de los vectores de los patrones de comportamiento deben de ser iguales o mayores a 255, por lo tanto cualquier valor por encima de este valor es funcional para el sistema.

Se busca que el número de imágenes mal clasificadas sea prácticamente nulo, puesto que, un mal reconocimiento puede llevar a comportamientos indebidos del vehículo.

CONCLUSIÓN

Conclusiones

En la realización de este proyecto tuvimos la oportunidad de estudiar e implementar algunos algoritmos de Inteligencia Artificial, en concreto las Memorias Asociativas Morfológicas, también nos permitió ampliar nuestros horizontes en cuanto a los alcances y las potencialidades que tiene la programación en dos de los más importantes entornos de desarrollo en la actualidad, como son Visual Studio y MATLAB.

La investigación de información nos permitió conocer un importante campo de acción de los sistemas de Inteligencia Artificial, como son los sistemas de prevención de accidentes que muchos fabricantes de automóviles están implementando entre otros avances tecnológicos presentes en esta área.

La realización de los métodos para la manipulación de imágenes que necesitábamos nos llevó a obtener conocimientos muy útiles de cara a la vida laboral que comenzaremos en breve, así mismo nos vimos forzados a utilizar nuestra creatividad para hacer que el programa funcionara de manera optimizada.

La programación de la interfaz nos permitió agrupar y aplicar nuestros conocimientos de las materias de programación que llevamos a lo largo de la carrera, gracias a esto hemos obtenido cierta experiencia en resolver problemas por medio de la programación.

En lo referente al objetivo general, diseñamos exitosamente el software de simulación y comprobamos que las memorias asociativas tienen un amplio alcance en aplicaciones con imágenes, en este caso concreto, con señalamientos viales.

Además encontramos que si bien las memorias morfológicas son un algoritmo complejo, su ejecución en los equipos de cómputo actuales no es para nada tardada, su tiempo de retardo o latencia es muy bajo, lo que permite que el sistema se ejecute de una manera bastante transparente al usuario.

ANEXOS

Anexo I: Códigos

1.1 Código de conversión de imagen a Vector

```
public void matriz()
{
    double auxx;
    int indice = 0;
    for (int x = 0; x < bmp.Width; x++)
    {
        for (int y = 0; y < bmp.Height; y++)
        {
            rgb = bmp.GetPixel(x, y);

            //pasando imagen de señal a escala de grises
            auxx = 0.299*Int32.Parse(rgb.R.ToString()) +
0.587*Int32.Parse(rgb.G.ToString()) + 0.114*Int32.Parse(rgb.B.ToString());
            aux = (int)auxx;

            //vectorizando datos de señal a escala de grises
            mataux[indice] = aux;
            indice++;

        }
    }
    mataux[2500 + id1] = 255;
}
```

1.2 Código de creación de la matriz máxima

```
public void maxima()
{
    for (int i = 0; i < 50; i++)
        for (int j = 0; j < 2550; j++)
            for (int k = 0; k <= 48 ; k++)
            {
                if (MW1[i, j, (k+1)] > mtz_ref[i,j])
                {
                    MAX[i, j] = MW1[i, j, (k+1)];
                    mtz_ref[i, j] = MW1[i, j, (k + 1)];
                }
            }
}
```

1.3 Código de Recuperación

```
public void reuperar(int ar)
{
    for (int k = 0; k < 50; k++)
    {
        R[k] = 255;
    }
    for (int k = 0; k < 50; k++)
        for (int l = 0; l < 2550; l++)
        {
            maux[k, l] = MAX[k, l] + mataux[l];
        }
    for (int k = 0; k < 50; k++)
        for (int l = 0; l < 2550; l++)
        {
            if (R[k] >= maux[k, l])
                R[k] = maux[k, l];
        }
}
```

Anexo II: Referencias de la investigación.

II.1 Control crucero

- <http://www.circulaseguro.com/como-funciona-el-control-de-crucero/>

II.2 Control de crucero adaptativo

- <http://www.supermotor.com/revista/actualidad/242893/control-inteligente-velocidad:-tecnologia-servicio-conductor.html>

II.3 Sistema de información de señales de tráfico

- <http://www.volvocars.com/es/allcars/volvov40/specifications/Pages/features.aspx>
- <https://www.youtube.com/watch?v=X15nPXvGviE>

II.4 Límites máximos de velocidad

- <http://mx.lexdir.com/articles/limites-de-velocidad-532>

II.5 Normatividad de señales viales

(Ver Manual de Dispositivos para el Control del Tránsito en Calles y carreteras)

- <http://dgst.sct.gob.mx/index.php?id=602>
- <http://www.setravi.df.gob.mx/work/sites/stv/resources/LocalContent/515/1/reglamentodetransito.pdf>
- http://www.edomex.gob.mx/generaldegobierno/doc/pdf/REGLAMENTOS_DE_TRANNSITO_imagenes.pdf