



UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MÉXICO
CENTRO UNIVERSITARIO UAEM NEZAHUALCÓYOTL
LICENCIATURA EN INGENIERÍA EN SISTEMAS INTELIGENTES

“Sistema de Semáforos Inteligentes
para el Control de Tráfico Vehicular”

TESIS

QUE PARA OBTENER EL TÍTULO DE

Ingeniero en Sistemas Inteligentes

PRESENTAN

FRANCISCO MANZO CRUZ

LUIS ARZATE HERNÁNDEZ

DIRECTOR DE TESIS

M. EN C. JOSÉ ANTONIO CUEVAS BARRÓN

Cd. Nezahualcóyotl, México, enero de 2019

Resumen

Actualmente, el congestionamiento de tráfico vehicular es un problema que afecta directamente a los diferentes sectores de la población específicamente con pérdida de tiempo, obstrucción de avenidas por largas filas, acumulación de vehículos e incapacidad para definir la duración de viajes. En los últimos años los sistemas inteligentes han adquirido una gran trascendencia en el uso de tecnologías para la solución de problemas en zonas urbanas. Con ayuda de un sistema de semáforos inteligentes para el control de tráfico se puede disminuir considerablemente el congestionamiento en las intersecciones de avenidas que presentan una mayor afluencia vehicular.

El desarrollo de un sistema de semáforos inteligentes para el control de tráfico vehicular es posible mediante la implementación de tecnologías que involucran técnicas de procesamiento de imágenes y visión artificial. Tecnologías de uso libre que permiten un desarrollo a bajo costo como “Python” (Lenguaje de programación) y librerías como “OPEN CV”. Las técnicas de visión artificial simulan la observación de los seres humanos.

El ser humano recibe la información a través de los ojos, la información es enviada al cerebro a través del nervio óptico para ser procesada con el objetivo de influir en la toma de decisiones. Análogamente, mediante cámaras digitales es posible captar imágenes del tráfico vehicular en intersecciones de avenidas, procesar dichas imágenes en busca de objetos, análisis de texturas y movimiento con el fin de ajustar los intervalos de tiempo en cada semáforo de la intersección.

Abstract

Nowadays, traffic congestion is a problem that directly affects all sectors of the population, specifically by wasted time, increased vehicular queueing and inability to calculate the duration of trips. In recent years, intelligent systems have acquired a great importance in the use of technologies for the solution of urban problems. With the help of an intelligent control system for traffic lights, vehicular traffic congestion can be considerably reduced at the intersections of avenues that have a greater traffic flow.

The development of an intelligent control system for traffic lights is possible through the implementation of technologies that involve image processing and artificial vision techniques. Free-use technologies that allow low-cost development such as "Python" (programming language) and libraries such as "OPEN CV". Artificial vision techniques simulate human behavior.

Humans receive the information through their eyes, the information is sent to the brain through the optic nerve to be processed with the aim of influencing the decision making. Similarly, using digital cameras it is possible to capture images of vehicular traffic at intersections of avenues, process these images to find objects, analyze movement and textures in order to adjust the traffic lights time intervals for the intersection.

Índice general

RESUMEN	II
ABSTRACT	III
ÍNDICE GENERAL.....	IV
ÍNDICE DE FIGURAS	IX
ÍNDICE DE TABLAS.....	XII
CAPÍTULO 1 INTRODUCCIÓN	1
1.1 PLANTEAMIENTO DEL PROBLEMA.....	1
1.2 DELIMITACIONES	2
1.2.1 <i>Delimitación temática</i>	3
1.2.2 <i>Delimitación espacial</i>	3
1.2.3 <i>Delimitación temporal</i>	4
1.3 JUSTIFICACIÓN	4
1.4 HIPÓTESIS	5
1.5 OBJETIVOS	6
1.5.1 <i>Objetivo general</i>	6
1.5.2 <i>Objetivos específicos</i>	6
1.6 RESULTADOS ESPERADOS.....	7
1.7 ORGANIZACIÓN DEL DOCUMENTO.....	7
CAPÍTULO 2 ESTADO DEL ARTE	9
2.1 EVOLUCIÓN DE LOS SEMÁFOROS.....	9
2.2 CLASIFICACIÓN DE LOS SEMÁFOROS.....	11
2.2.1 <i>Semáforos no accionados por el tránsito</i>	12
2.2.1.1 Control sin mecanismo de sincronización para intersecciones aisladas.....	13
2.2.1.2 Control con mecanismo de sincronización para intersecciones aisladas	13
2.2.1.3 Control que permite coordinación para intersecciones sucesivas	13
2.2.2 <i>Semáforos accionados por el tránsito</i>	14
2.2.2.1 Control parcialmente accionado por el tránsito	15

2.2.2.2 Control totalmente accionado por el tránsito	16
2.2.2.3 Control adaptable al tránsito	16
2.2.2.4 Otros controles coordinados	17
2.2.2.5 Ciclos supervisor general.....	18
2.2.2.6 Casos especiales de sistemas progresivos de tiempo.	18
2.2.2.7 Coordinación mutua.....	19
2.3 SEMÁFOROS EN CIUDAD NEZAHUALCÓYOTL	19
2.4 SISTEMAS INTELIGENTES DE TRANSPORTE	20
2.4.1 Sistema de control de tránsito SCT (Argentina)	20
2.4.2 ARMM Car Count (Italia).....	22
2.4.3 SIEMENS Intelligent traffic management (Alemania).....	23
2.4.4 SCATS Sydney Coordinated Adaptive Traffic System (Australia).....	24
2.4.5 Traffic Control and Surveillance Systems (Hong Kong)	25
2.4.6 Surtrac Intelligent Traffic Signals (EUA).....	27
2.4.7 Sumitomo Intelligent Transport Systems (Japón).....	28
CAPÍTULO 3 MARCO TEÓRICO	30
3.1 PROCESAMIENTO DE IMÁGENES	30
3.1.1 Algoritmo para cambiar de color a escala de grises.....	32
3.1.2 Binarización.....	33
3.1.3 Umbralización	34
3.1.3.1 Algoritmo Otsu.....	36
3.1.4 Operaciones binarias con imágenes	39
3.2 VISIÓN ARTIFICIAL.....	42
3.2.1 Matriz de co-ocurrencia de niveles de gris.....	43
3.2.2 Algoritmo de Canny.....	45
3.2.2.1 Obtención del gradiente.....	45
3.2.2.2 Supresión no máxima al resultado del gradiente	46
3.2.2.3 Histéresis de umbral a la supresión no máxima.....	47
3.2.3 Algoritmos para substracción de fondo	49
3.2.3.1 Substracción con imagen de referencia.....	49
3.2.3.2 Substracción del fotograma anterior	49
3.2.3.3 Substracción del acumulado de fotogramas anteriores.....	50

3.2.3.4 Substracción KNN	50
3.2.4 Clasificador Haar cascade	51
3.2.4.1 Fase de entrenamiento	51
3.2.4.2 Fase de clasificación	53
3.2.5 Colisiones en un plano	54
3.2.5.1 Magnitud de un vector	55
3.2.5.2 Colisión entre un punto y un segmento de recta	56
CAPÍTULO 4 METODOLOGÍA	57
4.1 SCRUM	57
4.1.1 Principios de Scrum	58
4.1.1.1 Individuos e interacciones por sobre procesos y herramientas	59
4.1.1.2 Software funcionando por sobre documentación exhaustiva	59
4.1.1.3 Colaboración con el cliente por sobre la negociación de contratos	59
4.1.1.4 Respuesta al cambio por sobre el seguimiento de un plan	60
4.1.2 Elementos y roles de Scrum	60
4.1.2.1 Product Owner	62
4.1.2.2 Scrum Master	63
4.1.2.3 Equipo de Desarrollo	63
4.1.2.4 Product Backlog	63
4.1.2.5 Sprint Backlog	64
4.1.2.6 Incremento funcional potencialmente entregable	65
4.1.3 Aplicación	65
CAPÍTULO 5 DISEÑO E IMPLEMENTACIÓN	67
5.1 DESCRIPCIÓN GENERAL	67
5.2 MODELO VISTA CONTROLADOR	68
5.2.1 Modelo	69
5.2.2 Vista	69
5.2.3 Controlador	70
5.3 REQUERIMIENTOS FUNCIONALES	71
5.4 REQUERIMIENTOS NO FUNCIONALES	71
5.5 TOPOLOGÍA DE RED	71

5.6 ESTRUCTURA FÍSICA DEL SISTEMA.....	72
5.7 ESTRUCTURA LÓGICA DEL SISTEMA	74
5.8 DISEÑO DE INTERFACES GRÁFICAS.....	78
5.9 DESCRIPCIÓN DETALLADA DEL SISTEMA.....	79
5.9.1 Inicialización.....	79
5.9.2 Condición para modo inteligente.....	79
5.9.3 Ejecución de ciclo	79
5.9.4 Captura de Imágenes.....	80
5.9.5 Procesamiento de Imágenes.....	81
5.9.6 Clasificación.....	81
5.9.7 Análisis de texturas.....	82
5.9.8 Comparación de resultados	82
5.9.9 Ajuste de ciclos	83
5.9.9.1 Diseño para control de respuesta	83
5.9.10 Máscara de segmentación	88
5.9.11 Recolección de imágenes para entrenamiento	89
CAPÍTULO 6 PRUEBAS Y RESULTADOS	91
6.1 CONFIGURACIÓN DE LOS EXPERIMENTOS.....	91
6.1.1 Aplicación de máscaras de segmentación	94
6.1.2 Detección de movimiento.....	96
6.1.3 Recolección de imágenes para entrenamiento	97
6.1.4 Clasificación.....	98
6.1.5 Análisis de texturas.....	99
6.1.6 Ajuste de ciclos	99
6.2 DISCUSIÓN DE RESULTADOS.....	101
CAPÍTULO 7 CONCLUSIONES Y FUTURAS LÍNEAS DE DESARROLLO.....	102
7.1 CONCLUSIONES.....	102
7.2 FUTURAS LÍNEAS DE DESARROLLO	103
7.2.1 Implantación del sistema propuesto.....	104
7.2.2 Escalabilidad del sistema propuesto	104

CAPÍTULO 8 REFERENCIAS	105
APÉNDICE A: GLOSARIO DE TÉRMINOS	109

Índice de figuras

Figura 1.1 Vehículos registrados por año en ciudad Nezahualcóyotl.....	2
Figura 1.2 Municipio de Nezahualcóyotl.....	3
Figura 2.1 Diseño del primer semáforo instalado en Londres.....	10
Figura 2.2 Semáforo de LEDs.....	11
Figura 2.3 Semáforos en ciudad Nezahualcóyotl.....	19
Figura 2.4 Diagrama general del sistema SCT.....	21
Figura 2.5 Sensores del sistema ARMM.....	22
Figura 2.6 Consumo de servicios en la nube SIEMENS.....	23
Figura 2.7 Diagrama arquitectura de red sistema SCATS.....	25
Figura 2.8 Sistema para control de tráfico en Hong Kong.....	26
Figura 2.9 Intersecciones de Pittsburgh con sistema Surtrac.....	28
Figura 2.10 Diagrama de operación del sistema Sumitomo en Japón.....	29
Figura 3.1 Modelo de tres colores primarios RGB.....	31
Figura 3.2 Aplicación de algoritmo escala de grises.....	32
Figura 3.3 Aplicación de algoritmo de binarización.....	33
Figura 3.4 Aplicación de algoritmos de umbralización.....	35
Figura 3.5 Aplicación de algoritmo Otsu.....	39
Figura 3.6 Aplicación de operaciones lógicas en imágenes binarias.....	41

Figura 3.7 Aplicación de algoritmo Canny.....	48
Figura 3.8 Aplicación de algoritmo Substracción de fondo KNN.	50
Figura 3.9 Característica de borde para entrenamiento de haar cascade.....	51
Figura 3.10 Característica lineal para entrenamiento de haar cascade.	52
Figura 3.11 Característica de cuatro rectángulos para entrenamiento de haar cascade.	52
Figura 3.12 Ejemplo de características relevantes para un clasificador de rostros.....	53
Figura 3.13 Reconocimiento facial con haar cascade OPEN CV.....	54
Figura 3.14 Línea recta formada por los puntos P y Q.....	55
Figura 3.15 colisión entre un punto y un segmento de recta.....	56
Figura 4.1 Iteraciones que definen el ciclo de vida de un proyecto.....	61
Figura 4.2 Ventaja de Scrum contra metodologías de cascada.	62
Figura 5.1 Descripción general del sistema.	68
Figura 5.2 Modelo Vista Controlador.....	70
Figura 5.3 Diagrama de red en estrella.....	72
Figura 5.4 Estructura física del sistema.	73
Figura 5.5 Entradas y Salidas Raspberry Pi.....	74
Figura 5.6 Diagrama de flujo de algoritmo principal.	75
Figura 5.7 Diagrama de flujo de algoritmo para recolección de imágenes.....	77
Figura 5.8 Diseño de interfaces gráficas de usuario.	78

Figura 5.9 Ejecución de ciclo para dos semáforos A y B.....	80
Figura 5.10 Diseño de imagen resultado de una clasificación.	81
Figura 5.11 Ejemplo de texturas generadas a partir del tráfico.	82
Figura 5.12 Combinaciones en un cruce simple de dos sentidos.	84
Figura 5.13 Combinaciones de respuesta para un cruce simple.....	86
Figura 5.14 Afluencia vehicular equivalente en dos sentidos.	88
Figura 5.15 diseño de fases para la aplicación de máscaras de segmentación.....	89
Figura 5.16 Diseño de proceso para recolección de imágenes positivas.....	90
Figura 6.1 Ubicación 1 de la captura de video para pruebas.	91
Figura 6.2 Ángulos de cámara: primera configuración.	92
Figura 6.3 Ubicaciones 2, 3, 4 y 5 de la transmisión de video para pruebas.	93
Figura 6.4 Ángulos de cámara: segunda configuración.	94
Figura 6.5 Conjunto de máscaras de segmentación.	95
Figura 6.6 Detección de movimiento.	96
Figura 6.7 Recolección de imágenes para entrenamiento de Haar Cascade.	97
Figura 6.8 Aplicación de clasificador Haar Cascade Open CV.	98
Figura 6.9 Maqueta para simulación de ciclos.	100

Índice de tablas

Tabla 1.1 Delimitación temporal.....	4
Tabla 3.1 Compuerta lógica AND.....	40
Tabla 3.2 Compuerta lógica OR.....	40
Tabla 3.3 Compuerta lógica NOT.....	40
Tabla 3.4 Compuerta lógica XOR.....	41
Tabla 3.5 Representación matricial de una imagen en escala de grises.....	43
Tabla 3.6 Cálculo de la matriz de co-ocurrencia.....	44
Tabla 3.7 Matriz simétrica de co-ocurrencia.....	44
Tabla 3.8 Matriz normalizada de co-ocurrencia.....	44
Tabla 3.9 Máscara de convolución para obtener el filtro Gaussiano.....	46
Tabla 4.1 Product Backlog del proyecto.....	64
Tabla 5.1 Posibles combinaciones en un cruce simple de dos sentidos.....	84
Tabla 5.2 Intervalos de tiempo mínimo y tiempo máximo.....	85
Tabla 5.3 Eventos para el control de respuesta del sistema.....	85
Tabla 5.4 Posibles combinaciones en un cruce de cuatro sentidos.....	87
Tabla 6.1 Configuración de conectores GPIO.....	99

Capítulo 1

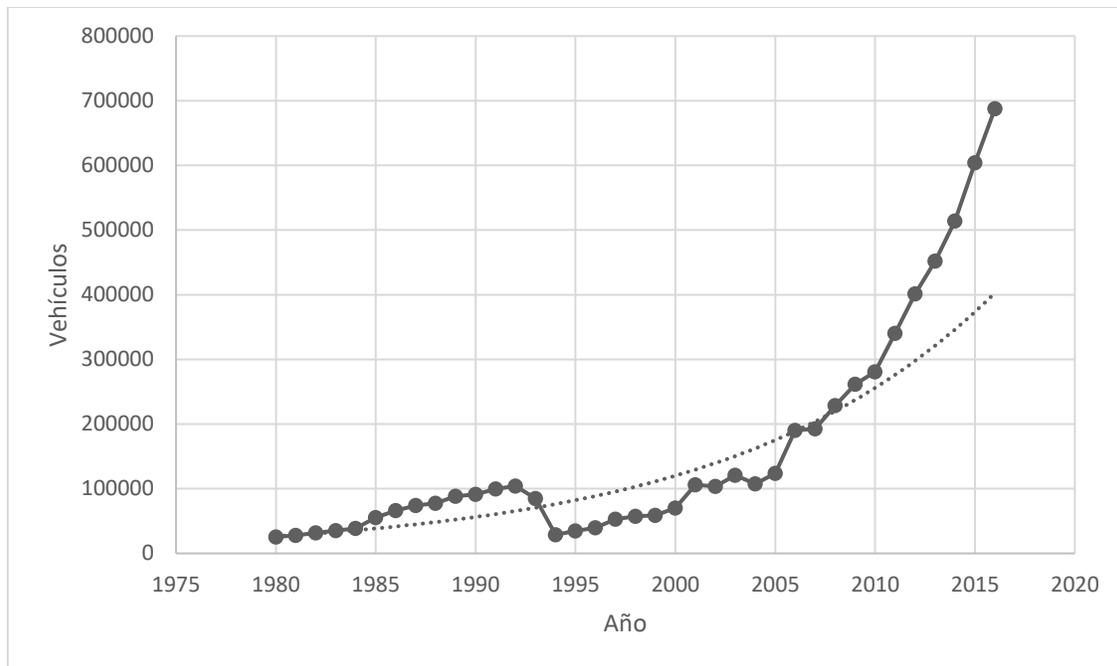
Introducción

En este capítulo se describe el congestionamiento de tráfico vehicular como un problema que se presenta en las principales zonas urbanas y que afecta directa e indirectamente a toda la población. Además, se plantea la necesidad de proponer e implementar soluciones que mejoren la movilidad mediante el uso de tecnologías de la información. Por último, se definen los objetivos de esta tesis.

1.1 Planteamiento del problema

Actualmente el número de habitantes en zonas urbanas crece de manera considerable, al mismo tiempo la capacidad de compra y el uso de automóvil particular por parte de los habitantes también aumenta. El congestionamiento de tráfico vehicular es un problema que afecta directamente a los diferentes sectores de la población específicamente con pérdida de tiempo, obstrucción de avenidas por largas filas, acumulación de vehículos, incapacidad para definir la duración de viajes, además, el congestionamiento de tráfico vehicular potencializa otros problemas como son: desperdicio de combustible, deterioro del medio ambiente y deterioro de la salud de la población.

De acuerdo con cifras del INEGI (Instituto Nacional de Estadística y Geografía), durante el año 2016 había 687 mil 837 vehículos en el municipio de Nezahualcóyotl (INEGI, 2016), hoy en día dicha cifra sigue en aumento, como se muestra en la siguiente gráfica (Figura 1.1) el aumento en el número de vehículos registrados en el municipio obedece una tendencia exponencial. Esto representa un problema considerando que la capacidad de las avenidas es limitada, esto aunado a otros factores como una inadecuada cultura vial fomenta el congestionamiento de tráfico vehicular en avenidas principales.



*Figura 1.1 Vehículos registrados por año en ciudad Nezahualcóyotl.
(Elaboración propia)*

Los principales medios que existen para regular el tráfico en el municipio son: implementación de reglamento de tránsito, programas de verificación de automóviles, programas de “hoy no circula”, sistemas de control de tránsito vehicular mediante semáforos, los cuales se colocan en los cruces de las avenidas más transitadas y en ocasiones asignación de oficiales de tránsito para descongestionar dichas avenidas; sin embargo, estos medios que operan actualmente no son suficientes para disminuir el congestionamiento del tráfico vehicular en horarios donde coinciden actividades comunes de la población, por ejemplo actividades escolares y/o laborales.

1.2 Delimitaciones

El trabajo de tesis se realizó con base en las siguientes delimitaciones:

1.2.1 Delimitación temática

La temática consiste en el diseño y desarrollo de un sistema de semáforos inteligentes para el control de tráfico vehicular que ayude a reducir el congestionamiento de tráfico vehicular en intersecciones mediante el uso de tecnologías de procesamiento de imágenes y visión artificial.

1.2.2 Delimitación espacial

El estudio y el desarrollo de este trabajo fueron realizados en México, ciudad Nezahualcóyotl, Estado de México.



*Figura 1.2 Municipio de Nezahualcóyotl.
(Google Maps. INEGI, 2018)*

1.2.3 Delimitación temporal

En la siguiente tabla se muestran los periodos comprendidos para cada una de las actividades que fueron realizadas.

	nov-17	dic-17	ene-17	feb-17	mar-18	abr-18	may-18	jun-18
Antecedentes	■							
Planteamiento del problema	■							
Justificación	■							
Objetivos	■							
Hipótesis	■							
Metodología		■	■					
Diseño		■	■					
Desarrollo y programación			■	■	■	■		
Pruebas				■	■	■		
Resultados					■	■	■	■
Conclusiones			■	■	■	■	■	■

*Tabla 1.1 Delimitación temporal.
(Elaboración propia)*

1.3 Justificación

En los últimos años los sistemas inteligentes han adquirido una gran trascendencia en el uso de tecnologías para la solución de problemas en zonas urbanas. El desarrollo de un sistema de semáforos inteligentes para el control de tráfico vehicular es posible mediante la implementación de tecnologías que involucran procesamiento de imágenes y visión artificial. Con ayuda de un sistema de semáforos inteligentes para el control de tráfico se puede disminuir considerablemente el congestionamiento en las intersecciones de avenidas que presentan una mayor afluencia vehicular.

Las ventajas de desarrollar un sistema inteligente para los semáforos que operan actualmente en la ciudad son:

- Utilización de infraestructura existente.
- Bajo costo en el desarrollo e implementación
- Disminución de congestionamiento vehicular en intersecciones de avenidas con mayor afluencia vehicular.

Los beneficios más importantes que trae consigo la implementación de un sistema de semáforos inteligentes para el control de tráfico vehicular son:

- Ahorro de tiempo por parte de los automovilistas ya que manejar vehículos no se considera como una actividad productiva para la mayoría de las personas.
- Ahorro de combustible debido a que al existir una reducción en el tiempo de manejo también se reduce de manera directa el consumo de combustible.
- Reducción en las emisiones de CO₂ lo cual representa una disminución en el deterioro del medio ambiente y al mismo tiempo en el deterioro de salud de la población.

1.4 Hipótesis

El uso de tecnología de captura de imágenes conjuntamente con la implementación de algoritmos de visión artificial permitirá el desarrollo de un sistema con semáforos inteligentes que ayuden a disminuir problemas de congestionamiento vehicular en la ciudad de Nezahualcóyotl.

1.5 Objetivos

A continuación, se describen las metas que fueron establecidas para la sustentación de esta tesis.

1.5.1 Objetivo general

Desarrollar un sistema de semáforos inteligentes que controle el cambio de luces en los semáforos utilizando algoritmos de visión artificial y procesamiento de imágenes para los puntos críticos de tráfico en la ciudad Nezahualcóyotl.

1.5.2 Objetivos específicos

- Analizar el estado del arte de los sistemas de control de tráfico vehicular ya existentes.
- Identificar una de las intersecciones con mayor congestionamiento de tráfico vehicular en la ciudad Nezahualcóyotl mediante medición, observación y análisis del tráfico vehicular.
- Adquirir el conocimiento sobre el funcionamiento, estructura y operación de los semáforos que son utilizados actualmente.
- Analizar diferentes algoritmos para el análisis de imágenes con el fin de adaptarlos al sistema inteligente.
- Desarrollar un sistema inteligente compatible con los semáforos existentes que controle el cambio de luz de los mismos con base en la cantidad de tráfico vehicular.
- Probar el sistema inteligente con secuencias de imágenes de alguno de los cruces críticos de la ciudad para verificar que el sistema inteligente genera un reconocimiento óptimo en un ambiente real y no controlado.

1.6 Resultados esperados.

- Un sistema de semáforos inteligentes que controle el cambio de luces en los semáforos utilizando algoritmos de visión artificial y procesamiento de imágenes.
- Reconocimiento óptimo de las imágenes capturadas.
- Una metodología implementada que brinde escalabilidad para futuros proyectos de desarrollo e implementación.
- Una propuesta para reducir el tráfico en algunos puntos de la ciudad en horas pico.
- Un sistema con mayor autonomía que no requiera de algún operador para controlar el tráfico y que mejore la logística de operación actual.
- Un sistema adaptable a los semáforos utilizados hoy en día en las avenidas principales de la ciudad.

1.7 Organización del documento

Este trabajo se encuentra organizado de la siguiente manera:

- En el **capítulo 1** se plantea el congestionamiento de tráfico vehicular como un problema, se propone el desarrollo de un sistema como solución y se definen los objetivos de este trabajo.
- En el **capítulo 2** se describen las etapas de evolución de las señales de control de tráfico, se resume la reglamentación vigente correspondiente al control de semáforos en México y se analizan las estrategias implementadas en otras ciudades del mundo para reducir el congestionamiento de tráfico vehicular.
- En el **capítulo 3** se introducen los elementos teóricos referentes a las disciplinas “Visión Artificial” y “Procesamiento de Imágenes” que han sido estudiados para el desarrollo del sistema.

- En el **capítulo 4** se presenta la metodología de desarrollo empleada en el diseño y la programación del sistema.
- En el **capítulo 5** se describen el diseño y arquitectura propuestos para el sistema.
- En el **capítulo 6** se muestran las pruebas realizadas al sistema y se presentan los resultados obtenidos.
- En el **capítulo 7** se exponen las conclusiones obtenidas a partir del desarrollo y las pruebas realizadas al sistema, se proponen otras líneas de desarrollo relacionadas con la escalabilidad del sistema y trabajos a futuro.
- Por último, en el **capítulo 8** se encuentran las referencias que fueron consultadas para la sustentación de esta tesis.

Capítulo 2

Estado del arte

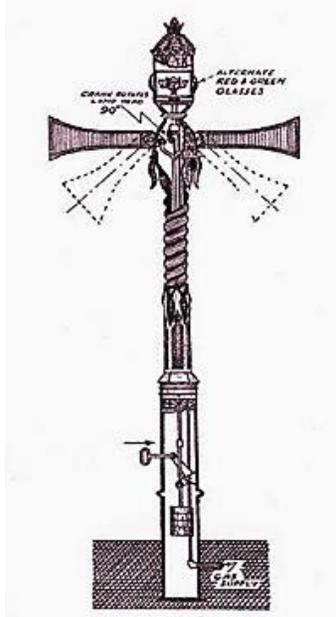
En este capítulo se mencionan las etapas de evolución de las señales de tránsito para regular el tráfico vehicular, se describe el estado actual de los semáforos que operan en las intersecciones de avenidas principales en la ciudad Nezahualcóyotl, se resumen aspectos importantes de la reglamentación vigente en México para el control de semáforos y se mencionan las soluciones que se han implementado en otros países para disminuir el congestionamiento de tráfico vehicular mediante “Sistemas de tráfico inteligentes” o “ITS” por sus siglas en inglés.

2.1 Evolución de los semáforos

“Antes de los semáforos para el control de tráfico vehicular, las luces rojas y verdes ya se empleaban en las señales de ferrocarril, que a su vez se heredaron de las señales marítimas, en donde se usaban banderas (ahora luces) de estos colores para evitar que los barcos colisionaran” (Bautista Rojas, 2016).

El origen de los semáforos para el control de tráfico vehicular se remonta al año 1868 en la ciudad de Londres, Inglaterra, donde se instaló el primer semáforo. El creador John Peake Knight (ingeniero ferroviario) se inspiró en los semáforos que controlaban los ferrocarriles de la época. El control dependía de un oficial de policía quién manualmente realizaba los cambios. Durante el día la señalización se basaba en dos brazos mecánicos y durante la noche la señalización incluía dos faroles: uno rojo y uno verde. Se utilizó gas como fuente de energía para este primer semáforo, por lo que un mes después se descartó el diseño debido a una explosión que mató al operador.

En el año 1914 se instaló en Cleveland, Estados Unidos, el modelo diseñado por el ingeniero Earnest Serrine quién reemplazó la instalación de gas por una instalación eléctrica.



*Figura 2.1 Diseño del primer semáforo instalado en Londres.
(Algarabía, 2013)*

En el año 1920, el oficial de policía de Detroit William Potts agregó la luz ámbar intermedia para advertir del cambio de luces y así evitar muchos accidentes de tránsito que ocurrían justo en el cambio inminente de luces. Sin embargo, debido a la II Guerra Mundial la producción de los semáforos se retrasó hasta 1940, igualmente en ese año la legislación de Estados Unidos aprobó el modelo de tres luces (verde, ámbar y rojo) que se sigue utilizando actualmente.

En México, el presidente Porfirio Díaz estableció un cuerpo de policías de tránsito para regular el flujo vehicular en las principales avenidas, fue hasta después del año 1930 que se instaló el primer semáforo automático en el cruce de Avenida Juárez y San Juan de Letrán (esquina donde ahora confluyen la Torre Latinoamericana, la explanada del Palacio de Bellas Artes y el anexo Guardiola).

A partir de la invención de los semáforos electrónicos de tres luces, se presentó la necesidad de automatizar los mecanismos de control, como respuesta a esta necesidad surgieron los semáforos cronometrados o de temporizador con intervalos fijos. Actualmente, los semáforos funcionan mediante diodos LED, se han añadido extensiones para peatones, emisión de señales auditivas, sin embargo, la clasificación de los semáforos depende del tipo de unidad de control que gestiona el funcionamiento.

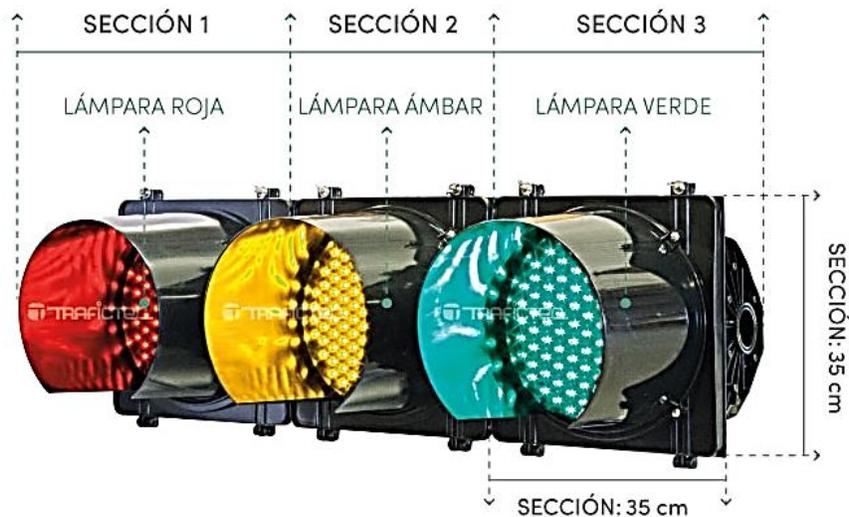


Figura 2.2 Semáforo de LEDs.

(Trafictec, 2018)

2.2 Clasificación de los semáforos

En México el Manual de Señalización Vial y Dispositivos de seguridad (Secretaría de Comunicaciones y Transportes, 2014) establece los siguientes criterios para la clasificación de semáforos para el control del tránsito de vehículos:

2.2.1 Semáforos no accionados por el tránsito

El uso primordial es en intersecciones viales, en donde los patrones de movilidad del tránsito son relativamente estables y constantes, o donde las variaciones del tránsito que se registran pueden adaptarse fácilmente a una programación coordinada sin causar demoras o congestión importantes.

La finalidad de un sistema de semáforos se cumple si este se apega a las necesidades del tránsito. Los ciclos excesivamente largos y la división inapropiada de los mismos ocasionan faltas de respeto y desobediencia. Los semáforos se manejarán manualmente cuando sea estrictamente necesario (durante eventos especiales como accidentes, manifestaciones o congestionamiento de una vía primaria), ya que este tipo de funcionamiento es menos eficaz que el control automático, con tiempos previamente fijados, especialmente en sistemas coordinados.

Cualquier plan de tiempos que se programe se confrontará con la información de aforos de tránsito para tener la seguridad de que los cambios de intensidad de tránsito en las vías se regulen lo mejor posible. Los factores que se tomarán en cuenta para programar el tiempo de los semáforos de una intersección son:

- Número de carriles y demás condiciones geométricas.
- Variaciones del flujo del tránsito para cada movimiento direccional.
- Necesidades de los vehículos comerciales y de transporte público.
- Lapso en segundos, entre el paso de dos vehículos consecutivos que salen de la intersección.
- Necesidades de los peatones.
- Necesidad de desalojar de la intersección a los vehículos y los peatones, al cambiar las indicaciones.

Existen tres tipos de mecanismos de control para los semáforos no accionados por el tránsito:

2.2.1.1 Control sin mecanismo de sincronización para intersecciones aisladas

El uso de este tipo de control, se recomienda únicamente en aquellas intersecciones aisladas en donde no es posible que se presente la necesidad de sincronizarse con el de otra intersección.

2.2.1.2 Control con mecanismo de sincronización para intersecciones aisladas

Este tipo de control tiene un sistema de sincronización y se usará en intersecciones aisladas cuando:

- En el futuro sea probable que se necesite la coordinación del semáforo con otros o que éste vaya a ser supervisado por un control maestro.
- Sean aceptables las duraciones fijas de ciclos y de intervalos, todo el tiempo que dure la operación de control de tránsito.

2.2.1.3 Control que permite coordinación para intersecciones sucesivas

Los semáforos no accionados por el tránsito funcionarán coordinadamente dentro de un radio de 400 m. A distancias mayores aún puede resultar conveniente la coordinación de semáforos sin rebasar los 800 m. Hay varios tipos de controles para coordinación, uno de los sistemas implica la supervisión de los controles locales por medio de un control maestro, mediante interconexiones de cables, por radio o por medio de un GPS conectado vía satélite con el centro de control.

En los controles locales de estos sistemas pueden emplearse mecanismos de inducción o dispositivos electrónicos de tiempo.

2.2.2 Semáforos accionados por el tránsito

Se usan en las intersecciones donde los volúmenes de tránsito fluctúan considerablemente en forma irregular, y en donde las interrupciones de la circulación deben ser mínimas en la dirección principal. Los semáforos accionados por el tránsito se clasifican en dos categorías generales.

- Totalmente accionados: Disponen de medios para ser accionados por el tránsito en todos los accesos de la intersección.
- Parcialmente accionados: Disponen de medios para ser accionados por el tránsito en uno o más accesos de la intersección, pero no en todos.

Las ventajas que proporcionan los controles accionados por el tránsito son las siguientes:

- Pueden resultar más eficientes en intersecciones donde las fluctuaciones del tránsito no se pueden prever y programar en la forma requerida para los sistemas con controles no accionados.
- Pueden ser de mayor eficiencia en intersecciones donde una o más circulaciones son esporádicas y de intensidad variable.
- Generalmente son más eficientes en intersecciones de calles principales con calles secundarias, debido a que interrumpen la circulación en la calle principal únicamente cuando se requiere dar paso a vehículos y a peatones en la calle secundaria y restringen esas interrupciones al tiempo mínimo indispensable.
- Pueden dar la máxima eficiencia en las intersecciones desfavorablemente localizadas dentro del sistema progresivo, en los que las interrupciones del tránsito en la calle principal son inconvenientes y se mantendrán al mínimo en frecuencia y duración.

- Proporcionan una operación continua sin demoras innecesarias en intersecciones aisladas, donde los controles no accionados por el tránsito, en ocasiones funcionan en operaciones de destello durante lapsos de escaso movimiento.
- Tienen aplicación especialmente en intersecciones donde la operación de semáforos solo hace falta durante períodos cortos del día.

Los mecanismos de control que se emplearán incluyen las siguientes opciones:

2.2.2.1 Control parcialmente accionado por el tránsito

Los controles parcialmente accionados por el tránsito son aplicables principalmente en las intersecciones de arterias de alto volumen y altas velocidades, con calles secundarias de escasa circulación; razón por la que disponen de medios para ser accionados por el tránsito en uno o más accesos, pero no en todas las aproximaciones de la intersección. Los detectores se ubican solo en los accesos secundarios.

Estos controles permiten que en la calle principal normalmente se tenga indicación de luz verde, la cual cambia a la calle secundaria únicamente como resultado de la acción de los vehículos y de los peatones. La duración de la indicación de luz verde en la calle secundaria es proporcional a las demandas del tránsito de la misma, previéndose un límite máximo de tiempo, más allá del cual no se puede mantener la indicación de luz verde en la calle secundaria, aunque haya alta demanda de tránsito. Al término de la fase requerida, la indicación de luz verde vuelve a la calle principal y se mantiene como mínimo, durante un intervalo previamente fijado; al terminar el intervalo mínimo, el control vuelve a quedar libre para responder a la acción del tránsito en la calle secundaria.

El control parcialmente accionado por el tránsito no recibe ninguna acción para el tránsito que circula en la calle principal, y por consiguiente, pueden otorgar frecuentemente el derecho de paso a la calle secundaria, en los momentos más inoportunos para la circulación de la calle principal.

2.2.2.2 Control totalmente accionado por el tránsito

En los controles totalmente accionados por el tránsito, los detectores se instalan en todos los accesos de la intersección y el derecho de paso se le da a una calle como resultado de uno o más accionamientos en esa misma calle. Cuando no hay demandas del tránsito en ninguna de las calles, la indicación de luz verde normalmente permanecerá en aquella a la que se dio por último; pero cuando una de las calles tenga más tránsito que las demás podrán resultar de mayor eficacia revertir el derecho de paso a esa calle.

En el caso de accionamiento continuo en una misma calle, el derecho de paso cederá al tránsito que espera en la calle transversal; al terminar un lapso máximo predeterminado, automáticamente regresará a la primera calle en la primera oportunidad, misma que no se puede presentar sino hasta después de terminado un período mínimo con la indicación de luz verde en la calle transversal. La duración de la indicación de luz verde para cada calle, con condiciones normales de tránsito, fluctúan entre los valores máximo y mínimo prefijado, dependiendo de los lapsos entre accionamientos.

Con el tipo más común de control totalmente accionado por el tránsito, el derecho de paso, de acuerdo con los accionamientos será cedido inmediatamente a la calle transversal si el tiempo transcurrido entre accionamientos en la calle con indicación de luz verde excede cierto valor predeterminado y si el período mínimo para dicha calle ha expirado.

2.2.2.3 Control adaptable al tránsito

Los controles adaptables a la densidad del tránsito, totalmente accionados por éste, permiten que el intervalo correspondiente a los lapsos entre el paso de vehículos sucesivos que exceden los tiempos prefijados para cada una de las calles, disminuya durante cada fase, de acuerdo con ciertos factores de las circulaciones; por ello, la probabilidad de que el intervalo de luz verde termine y se ceda el derecho de paso a la

calle transversal, aumenta proporcionalmente a la disminución del tránsito que circula con luz verde, al transcurso del tiempo en el cual se tienen vehículos detenidos con la indicación de luz roja, y al número de los mismos. Estos y otros factores hacen que el control totalmente accionado por el tránsito del tipo de densidad, sea más sensible a las demandas de circulación con amplias variaciones de intensidad.

Por medio de un sistema que usa un control maestro accionado por la circulación para supervisar los controles locales, se logra una combinación progresiva flexible y de los semáforos accionados por el tránsito; para ello, se instalan detectores en lugares estratégicos del sistema progresivo para suministrar al control maestro la información del tránsito en esos puntos. El control maestro selecciona el ciclo y los desfases predeterminados para lograr el mejor equilibrio, de acuerdo con las intensidades de circulación que presentan en ese tiempo.

Los controles locales estarán conectados al control maestro, que los manejará en un momento dado, conforme al ciclo y desfase seleccionados por el control maestro. En el sistema de control adaptable a la densidad del tránsito, el cambio de una combinación de tiempos a otra se efectúa rápidamente y con interferencia mínima para el tránsito que circula durante el cambio.

Si los controles locales son parcialmente accionados por el tránsito, la duración de fase correspondiente a la calle secundaria, dentro del ciclo seleccionado, se determina por medio de accionamientos en detectores instalados en la misma. Este sistema permite una gran flexibilidad para lograr la coordinación efectiva de circulaciones tanto en una vía como en una red de calles.

2.2.2.4 Otros controles coordinados

Además del control adaptable a la densidad del tránsito, existen otras aplicaciones de controles accionados por el tránsito para lograr la operación coordinada de una serie de intersecciones con semáforos, como las siguientes:

2.2.2.5 Ciclos supervisor general

Se puede imponer un ciclo supervisor general sobre una serie de controles parcialmente accionados por el tránsito, mediante un control maestro de tiempo que envía impulsos a cada uno de los controles, o por medio de un control local, o por sistemas de sincronización en cada intersección. El ciclo general y los desfases se determinan de la misma manera que para un control no accionado por el tránsito. La función del ciclo supervisor, es asegurar que los controles parcialmente accionados por el tránsito permitan, cuando menos, el intervalo mínimo de luz verde en la calle principal en la proporción de tiempo más conveniente para mantener la circulación progresiva en ésta.

Cada control parcialmente accionado por el tránsito, incrementa el intervalo de luz verde en la calle principal todo el tiempo que no lo requiera la demanda de la calle secundaria y, por lo tanto, permite la mayor fluidez posible. Una desventaja que se puede presentar es que un intervalo de luz verde tan largo, puede congestionar seriamente alguna intersección adyacente en la que las demandas del tránsito transversal sean mayores. El tipo de operación que se acaba de describir, obviamente no tiene las mismas características relativas a la regulación de velocidades que el de un sistema progresivo y, por ende, sólo se empleará cuando es más importante mover el mayor volumen de tránsito con el mínimo de demoras, para controlar las velocidades.

2.2.2.6 Casos especiales de sistemas progresivos de tiempo.

Una aplicación más del accionamiento del tránsito en circulaciones coordinadas se presenta en un sistema progresivo con semáforos accionados en el que el espaciamiento entre intersecciones en uno o más lugares del sistema es tal, que la mejor disposición de los tiempos progresivos implica una reducción en la eficiencia de sincronización. Las intersecciones complicadas en un sistema progresivo, también pueden ocasionar dificultades para fijar los tiempos. En dichos casos es posible lograr mejores resultados con controles accionados por el tránsito.

2.2.2.7 Coordinación mutua.

La coordinación mutua se realiza a través de dos controles parcialmente accionados por el tránsito, aislados, ubicados en intersecciones adyacentes o próximas entre sí, de tal manera que se reduzca al mínimo las paradas del tránsito en la arteria principal; esto se logra interconectando los controles de manera que haya un desfase fijo entre accionamientos en las calles transversales.

2.3 Semáforos en ciudad Nezahualcóyotl

El flujo de tráfico vehicular en la ciudad Nezahualcóyotl se regula mediante “semáforos de tiempo fijo” o “semáforos no accionados por el tráfico” (sección 2.2.1, página 12). El ciclo y los intervalos de este tipo de semáforos se controlan mediante temporizadores cronometrados.



*Figura 2.3 Semáforos en ciudad Nezahualcóyotl.
(Google Maps, 2017)*

2.4 Sistemas inteligentes de transporte

“Conjunto de soluciones tecnológicas de las telecomunicaciones y la informática (telemática) diseñadas para mejorar la operación y seguridad del transporte terrestre, tanto para carreteras como para vialidades urbanas” (Secretaría de Comunicaciones y Transportes, 2014). Debido a que el congestionamiento de tráfico vehicular es un problema que afecta a las principales ciudades de todo el mundo, son cada vez más los investigadores en conjunto con empresas del sector privado e instituciones públicas que se han sumado a la tarea de proponer soluciones a corto plazo, con el objetivo de resolver dicha problemática.

Es importante señalar que cada una de las propuestas de solución responde a una demanda local, con base en las condiciones y necesidades específicas de la ciudad. Algunos de los sistemas que se han implementado en otras ciudades del mundo para resolver el problema de congestionamiento vehicular son:

2.4.1 Sistema de control de tránsito SCT (Argentina)

Se trata de un sistema altamente funcional que opera actualmente en las ciudades de Tandil, Benito Juárez y Laprida geográficamente pertenecientes a Argentina. La principal característica de este sistema es que opera de manera centralizada, esto quiere decir que todos los semáforos se encuentran en comunicación directa con una central de control. Su funcionabilidad depende de un controlador satelital para GPS y de una red amplia con servicios de internet.

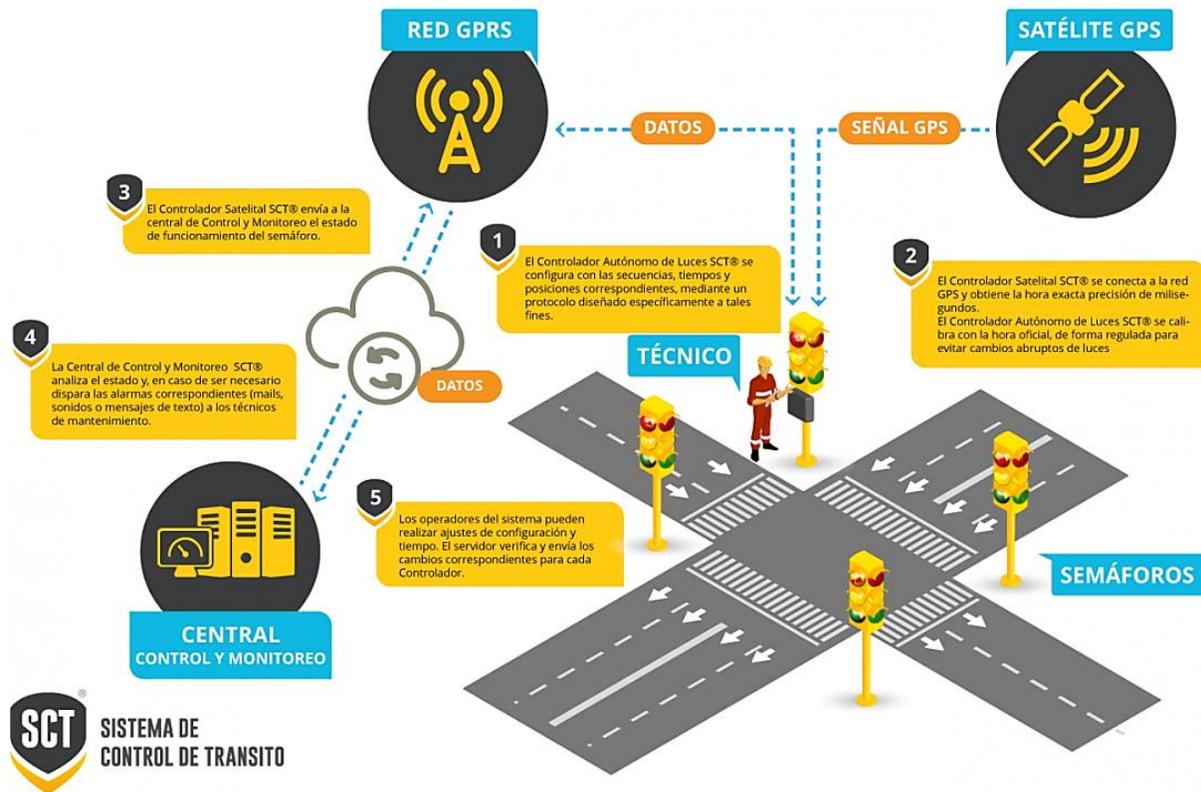


Figura 2.4 Diagrama general del sistema SCT.
(SCT vial, 2012)

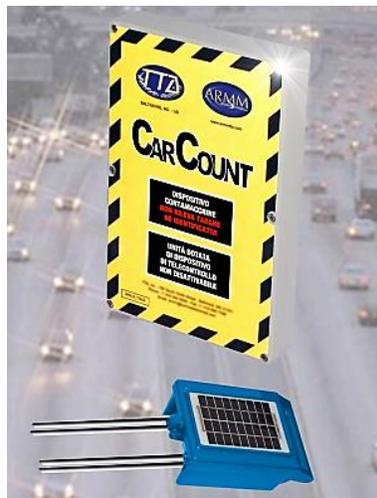
- El Controlador Autónomo de Luces se configura con las secuencias, tiempos y posiciones correspondientes, mediante un protocolo diseñado específicamente a tales fines.
- El Controlador Satelital SCT® se conecta a la red GPS y obtiene la hora exacta con precisión de milisegundos, para informarla al Controlador Autónomo de Luces a intervalos regulares.
- El Controlador Autónomo de Luces SCT® se calibra con la hora oficial, de forma regulada para evitar cambios abruptos de luces, e informa al Controlador Satelital su estado de funcionamiento.
- El Controlador Satelital SCT® envía a la central de Control y Monitoreo el estado de funcionamiento del semáforo.

- La Central de Control y Monitoreo SCT® analiza el estado y, en caso de ser necesario dispara las alarmas correspondientes (mails, sonidos o mensajes de texto) a los técnicos de mantenimiento; también se verifican y envían los cambios de configuración correspondientes para cada Controlador, en caso de ser necesario (SCT vial, 2012).

Al tratarse de un sistema centralizado y que además depende de servicios externos como internet y transmisión de señales vía GPS, muestra una desventaja ya que requeriría una mayor inversión e infraestructura en la implementación del sistema en ciudades como la ciudad de Nezahualcóyotl.

2.4.2 ARMM Car Count (Italia)

Sistema centralizado que funciona mediante la instalación de sensores neumáticos en el suelo para el conteo de vehículos, almacena en tiempo real la información referente al flujo de tráfico vehicular, además gracias a las herramientas que provee para el análisis de datos estadísticos es un sistema que se ha implementado en aeropuertos de distintos países (MeteoPoint, 2016).

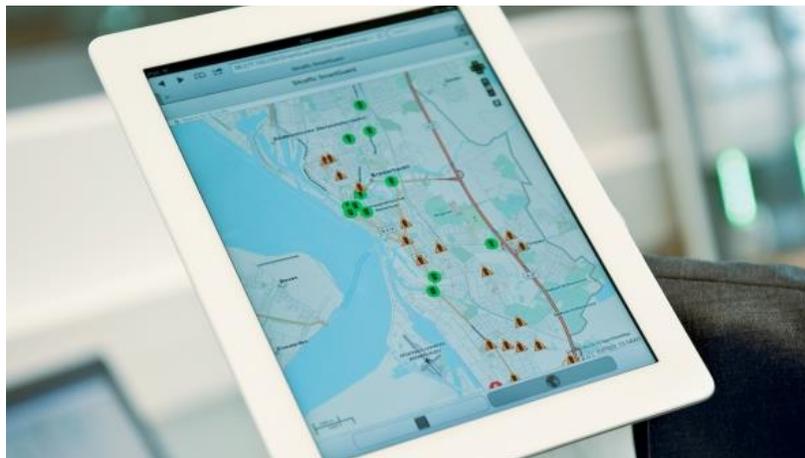


*Figura 2.5 Sensores del sistema ARMM.
(MeteoPoint, 2016)*

La colocación de sensores sobre el asfalto de las avenidas implica una mayor inversión en infraestructura y mantenimiento por lo que actualmente la implementación de este sistema en la ciudad Nezahualcóyotl no sería óptima si se requiere una solución de bajo costo.

2.4.3 SIEMENS Intelligent traffic management (Alemania)

El Sistema creado por la compañía SIEMENS ha sido implementado en 50 ciudades de 18 países diferentes incluyendo las principales ciudades de Alemania, opera mediante el consumo de servicios web y servicios de nube exclusivos de la compañía. Actualmente es el sistema de gestión de tráfico vehicular en la nube más grande del mundo.



*Figura 2.6 Consumo de servicios en la nube SIEMENS.
(Siemens, 2016)*

La computadora de tráfico smartGuard de Siemens es un sistema de control de tráfico que opera en una nube segura. El servicio permite a las ciudades pequeñas y municipios construir sistemas virtuales de control de tráfico basados en la web sin tener que invertir en hardware costoso.

En cambio, se conectan a una computadora de tráfico central totalmente operada por Siemens. Para integrar las infraestructuras de las ciudades en smartGuard, sistemas de semáforos, detectores de tráfico y aparcamientos, se desarrolló un controlador especial que se puede alojar en un armario de control junto a cada instalación. El personal autorizado puede conectarse desde cualquier dispositivo con capacidad de Internet a través de una nube privada, y administrar los sistemas (Siemens, 2016).

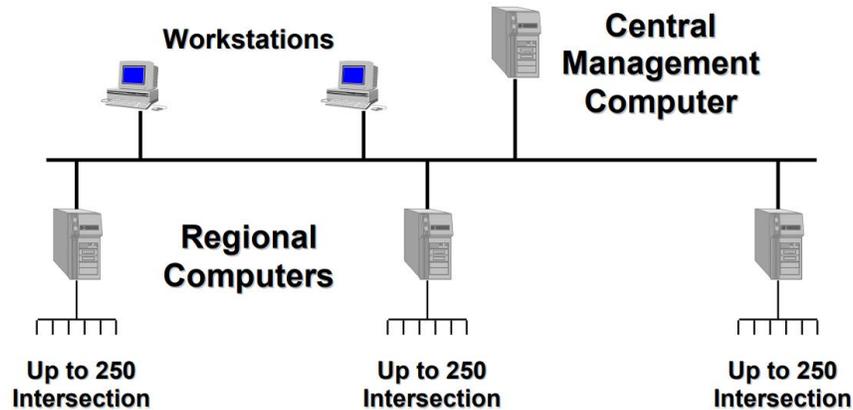
La centralización y la contratación de personal para el análisis del tráfico en tiempo real, junto con la contratación de servicios web y servicios de nube, representan limitantes en la implementación de este sistema en la ciudad Nezahualcóyotl considerando soluciones a bajo costo y a corto o mediano plazo.

2.4.4 SCATS Sydney Coordinated Adaptive Traffic System (Australia)

El Sistema “SCATS” opera principalmente en la ciudad de Sídney, Australia y ha sido vendido a otros 27 países (SCATS, 2018). Las principales características de este sistema son:

- Información en tiempo real para operaciones avanzadas y monitoreo de mantenimiento, que incluye información de detección, sincronización, saturación, estado, comunicación y coordinación.
- Monitoreo con alarmas en tiempo real que. El monitoreo con alarmas de detección altamente intuitivo especifica el carril exacto, el problema de detectado y determina si ese problema es un detector de corto circuito o circuito abierto, todo desde la GUI central (Interfaz gráfica de usuario).
- Diagramas de tiempo y espacio en tiempo real: diagramas que ayudan a mostrar el estado en tiempo real de los corredores viales para monitorear las llegadas de congestiones, así como información teórica sobre tiempos de viaje.

- Informes históricos: SCATS presenta una gama de informes en pantalla fáciles de entender, para ayudar con la planificación futura y la revisión de operaciones anteriores. Esto significa que no hay necesidad de realizar análisis del tráfico de forma manual o mediante encuestas costosas. Los informes incluyen sincronización, detección, saturación de carril, volumen de carril, coordinación e información sobre consultas.



*Figura 2.7 Diagrama arquitectura de red sistema SCATS.
(SCATS, 2018)*

Su funcionalidad se sustenta en la colocación de sensores en el pavimento. Similar al sistema “ARMM Car Count” (sección 2.4.2, página 22). Por lo tanto, presenta desventajas similares para una posible implementación a bajo costo y a corto plazo en intersecciones de la ciudad Nezahualcóyotl.

2.4.5 Traffic Control and Surveillance Systems (Hong Kong)

Para satisfacer la necesidad siempre cambiante de la comunidad y afilar la ventaja de Hong Kong como un centro internacional de transporte, la política del Gobierno de este país es alentar el desarrollo de nuevas iniciativas de transporte.

Con este fin, el Gobierno está tomando la delantera para atraer a compañías extranjeras, inversores y expertos internacionales a Hong Kong para ayudar a desarrollar un sistema de transporte eficiente, seguro y respetuoso del medio ambiente. Desde principios de 1980, todos los túneles y el área de control Tsing Ma han sido equipados con instalaciones integradas de control y vigilancia del tráfico, incluyendo cámaras CCTV, detectores automáticos de incidentes, señales de control de carril, señales de límite de velocidad variable y señales de mensajes variables, etc. para una movilidad eficiente y efectiva.

Los sistemas de control y vigilancia del tráfico se han ampliado en los últimos años para incluir el corredor occidental de Shenzhen, el área de control de Tsing Sha, la carretera Tuen Mun y la autopista Tolo. Se están implementando y se implementarán nuevos sistemas de control y vigilancia de tráfico como parte de los proyectos de autopistas, incluida la ampliación de Fanling Highway, Central Wanchai Bypass, proyectos relacionados con Hong Kong-Zhuhai-Macao Bridge, Tuen Mun-Chek Lap Kok Link, Liantang / Punto de control fronterizo Heung Yuen Wai, obras asociadas y túnel de estaño Tseung Kwan O-Lam (Hong Kong Transport Department, 2018).



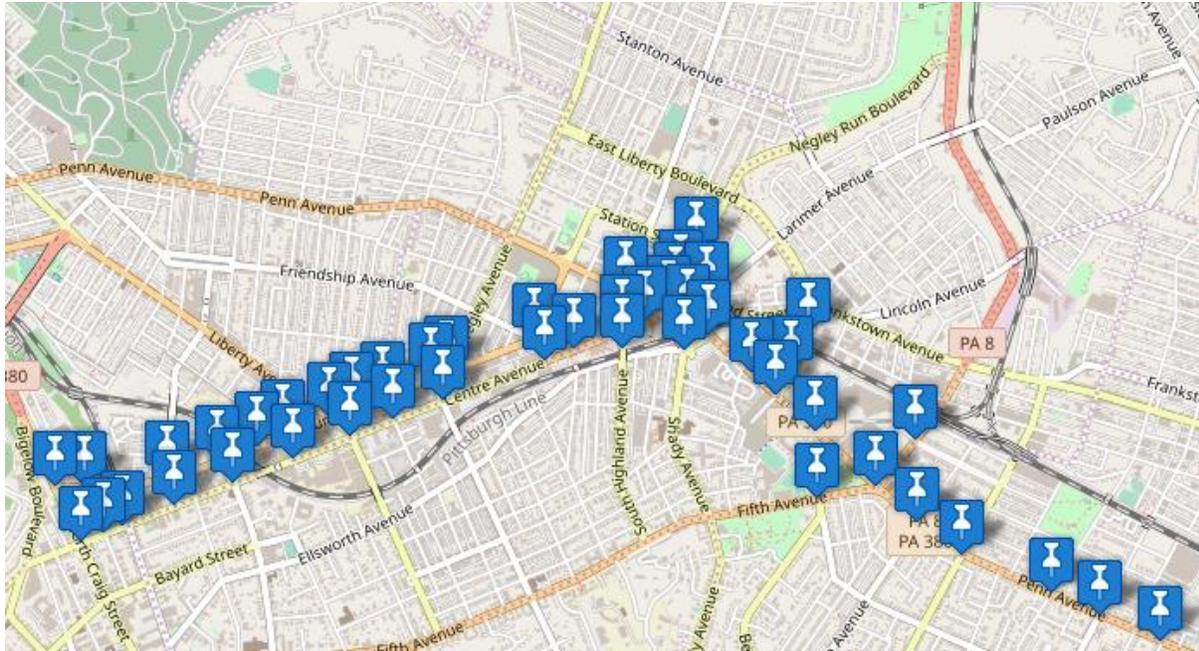
*Figura 2.8 Sistema para control de tráfico en Hong Kong.
(Hong Kong Transport Department, 2018)*

La implementación del sistema para control de tráfico en Hong Kong ha tomado varios años ya que es un sistema integral enfocado también a otros problemas como la seguridad, el desarrollo constante y la colaboración de varias empresas privadas con el sector gubernamental son ventajas que conllevan al aumento en los costos de desarrollo, operación y mantenimiento por lo que tampoco es un sistema considerable a corto plazo.

2.4.6 Surtrac Intelligent Traffic Signals (EUA)

Surtrac es un sistema para el control de la señal de tráfico en tiempo real, que combina la investigación de la inteligencia artificial y la teoría del tráfico. El sistema optimiza el rendimiento de las señales para el tráfico que está realmente en la carretera, lo que mejora el flujo de tráfico tanto en redes urbanas como en corredores y permite menos espera, congestión reducida, viajes más cortos y menos contaminación. En junio de 2012, la compañía puso en funcionamiento una versión piloto del sistema en nueve intersecciones del vecindario East Liberty de Pittsburgh.

El cruce de tres carreteras principales (Penn Circle, Penn Avenue y Highland Avenue) conduce a patrones de tráfico cambiantes a lo largo del día, lo que dificulta la gestión de la congestión con los métodos convencionales de control de la señal de tráfico. Surtrac redujo los tiempos de viaje en más del 25% en promedio, y los tiempos de espera se redujeron en promedio un 40%. Desde el éxito inicial del piloto, han expandido el sistema varias veces. A partir de 2016, Surtrac se instaló en 50 intersecciones en Pittsburgh, con expansiones adicionales actualmente en planificación (Surtrac, 2018).



*Figura 2.9 Intersecciones de Pittsburgh con sistema Surtrac.
(Surtrac, 2018)*

2.4.7 Sumitomo Intelligent Transport Systems (Japón)

ITCS es un sistema de gestión del tráfico a gran escala que logra un tráfico vial seguro y fluido. Este sistema realiza una gestión de tráfico óptima para las ciudades y el entorno al proporcionar información de tráfico a través de diversos medios y controlar señales basadas en el resultado del análisis de información de tráfico recopilada por los detectores en las carreteras.

Sumitomo Electric ofrece una amplia gama de productos, desde detectores, controladores de señal, dispositivos centrales hasta dispositivos de transmisión de datos. Estos productos se están utilizando en muchas instalaciones de Japón, incluido el Metropolitan Tokyo Traffic Control Center, el centro de control de tráfico más grande de Japón (Sumitomo Electric, 2018).



Figura 2.10 Diagrama de operación del sistema Sumitomo en Japón.
(Sumitomo Electric, 2018)

Capítulo 3

Marco teórico

En este capítulo se describen las técnicas de procesamiento de imágenes y visión artificial para el reconocimiento de objetos dentro de una imagen. Se especifican las herramientas y los elementos teóricos necesarios para el diseño propuesto de un sistema de semáforos inteligentes para el control de tráfico vehicular.

3.1 Procesamiento de imágenes

“Una imagen puede definirse como una función bidimensional $f(x, y)$, donde “ x ” y “ y ” son coordenadas espaciales (planas) y la amplitud de f en cualquier par de coordenadas (x, y) se llama intensidad o nivel de gris de la imagen en ese punto. Cuando x, y y los valores de intensidad de f son todas cantidades finitas y discretas, llamamos a la imagen una “imagen digital”. El campo del procesamiento de imágenes digitales se refiere al procesamiento imágenes digitales por medio de una computadora digital. Tenga en cuenta que una imagen digital está compuesta de un número finito de elementos, cada uno de los cuales tiene una ubicación particular y un valor. Estos elementos se denominan píxeles. Pixel es el término más utilizado para designar los elementos de una imagen digital” (Gonzalez & Woods, 2008).

Las técnicas utilizadas para el procesamiento de imágenes se basan en la estructuración de una imagen a través de una matriz de píxeles, análogamente podemos representar a cada uno de los píxeles mediante un vector de 3 dimensiones con la siguiente información:

$$A = [x, y, z] \tag{3.1}$$

Donde A representa al pixel, x representa la ubicación del pixel en el eje x , y representa la posición del pixel en el eje y de un plano, por último, z representa la intensidad del pixel.

Existen varios formatos de imagen y varias formas de modelar el color y la intensidad de los píxeles, entre los principales modelos se encuentran RGB, CMYK, HSV y HSL. El diseño del sistema propuesto se basa en el modelo RGB o BGR el cual define como colores primarios al rojo, verde y azul. La combinación de los tres colores genera blanco mientras que la ausencia de los tres genera negro por lo tanto las combinaciones restantes generan toda la gama de colores.

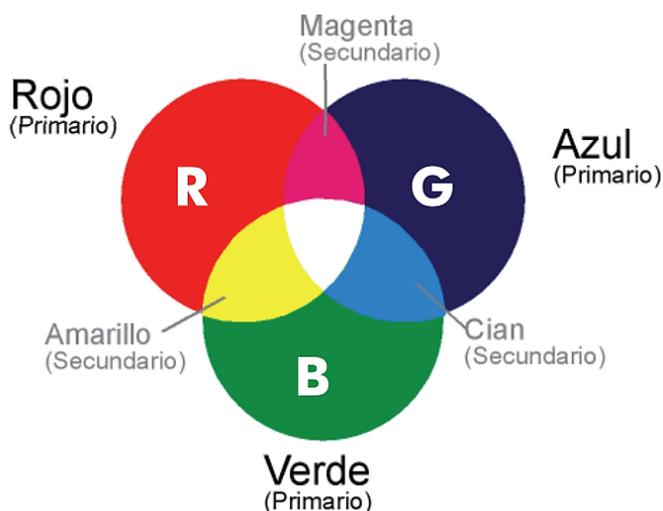


Figura 3.1 Modelo de tres colores primarios RGB.

(De Los Santos A. , 2011)

El procesamiento de imágenes se conforma de muchas familias de algoritmos entre las cuales destacan: cambio entre espacios de color, transformaciones geométricas, suavizado y umbralización, transformaciones morfológicas, detección de contornos, filtros, detección de líneas rectas etc. Asimismo, existe software con bibliotecas especializadas como Matlab, Open CV, con los algoritmos ya codificados, solo basta entender el funcionamiento del algoritmo con relación a los parámetros de entrada.

Para efectos de abstracción en este trabajo solo se documentan los algoritmos de procesamiento de imágenes que en conjunto con otras técnicas contribuyen con un aporte significativo al reconocimiento de objetos (en este caso automóviles) dentro de una imagen.

3.1.1 Algoritmo para cambiar de color a escala de grises

Para que el valor de un pixel en una imagen digital sea color gris tomando en cuenta las componentes R, G, y B se debe cumplir la siguiente condición:

$$R = G = B \quad (3.2)$$

El método más sencillo para que se cumpla la condición 3.2 es promediar los tres valores del modelo RGB del pixel original y colocar el valor resultante en las 3 componentes.

$$XG_{ij} = \frac{X_{i,j,1} + X_{i,j,2} + X_{i,j,3}}{3} \quad (3.3)$$

Donde XG es la matriz de grises, de dimensiones M x N y $X_{i,j,1}$ $X_{i,j,2}$ $X_{i,j,3}$ las componentes correspondientes a los canales R, G y B respectivamente (De Los Santos P. J., 2016).



*Figura 3.2 Aplicación de algoritmo escala de grises.
(OpenCV.org, 2018)*

3.1.2 Binarización

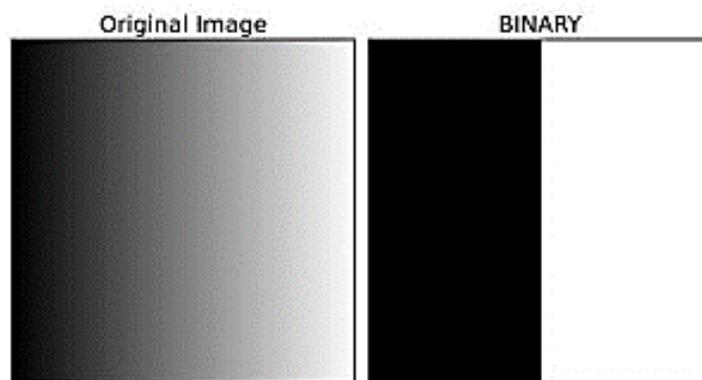
La complejidad de un algoritmo depende directamente de la cantidad de iteraciones y la cantidad de datos involucrados en el proceso de la información, por lo tanto, entre menos información se trabaje, los procesos finalizan más rápido y ocupan menos espacio en memoria. Binarizar es una de las formas de reducir el volumen de información cuando se trabaja con procesamiento de imágenes. Los algoritmos de Binarización consisten en asignar a cada pixel solamente dos posibles valores: 0 que representa el negro y 1 o 255 que representan blanco.

“Para obtener una imagen binaria se hace una transformación no lineal de la imagen de entrada, obteniéndose una imagen de salida en la cual cada pixel puede tomar alguno de dos valores: negro o blanco” (Sucar & Gómez, 2011). Del tal forma que:

$$S[x, y] = 1, E[x, y] > T \quad (3.4)$$

$$S[x, y] = 0, E[x, y] \leq T \quad (3.5)$$

Para este procedimiento es necesario transformar la imagen original a escala de grises, posteriormente se asigna un valor de umbral T , el cual es el valor que le indica al algoritmo el límite intermedio para convertir a blanco los pixeles que se encuentran por encima del umbral y en caso contrario convertir a negro.



*Figura 3.3 Aplicación de algoritmo de binarización.
(OpenCV.org, 2018)*

3.1.3 Umbralización

Durante el proceso de binarizar una imagen es necesario definir un valor para el umbral, si bien es un valor que se puede definir aleatoriamente, no en todas las posibles combinaciones se obtienen resultados óptimos y más aún cuando se trata de encontrar o analizar diferentes regiones en la imagen, por ejemplo cuando el objetivo es identificar como primer plano (color blanco) las regiones que pertenezcan a determinados objetos y como segundo plano (color negro) las regiones que pertenezcan al fondo de la imagen.

Por lo anterior surge el concepto de umbralización en el cual se utilizan algoritmos que permiten encontrar un valor óptimo para el umbral. Es extremadamente extensa la cantidad de algoritmos que existen para la umbralización de imágenes, si bien aún no existe el algoritmo perfecto que arroje resultados óptimos en cualquier imagen, existen categorías que nos permiten elegir el tipo de algoritmo dependiendo de la tarea que se desea realizar.

Separar caracteres que forman una palabra dentro de un texto, detectar ciertos tipos de células en imágenes médicas, en el caso de este proyecto separar los vehículos que aparecen en la imagen de una vialidad, son ejemplos de segmentación en imágenes. Mehmet Sezgin y Bülent Sankud establecieron las siguientes categorías para los algoritmos de umbralización en el año 2004:

- Algoritmos basados en la forma del histograma.
- Algoritmos basados en agrupamiento o clusterización.
- Algoritmos basados en la entropía de la imagen.
- Algoritmos basados en atributos de la imagen (bordes, momentos, texturas).
- Métodos basados en la información espacial de los píxeles.
- Algoritmos basados en características locales como rango y varianza.

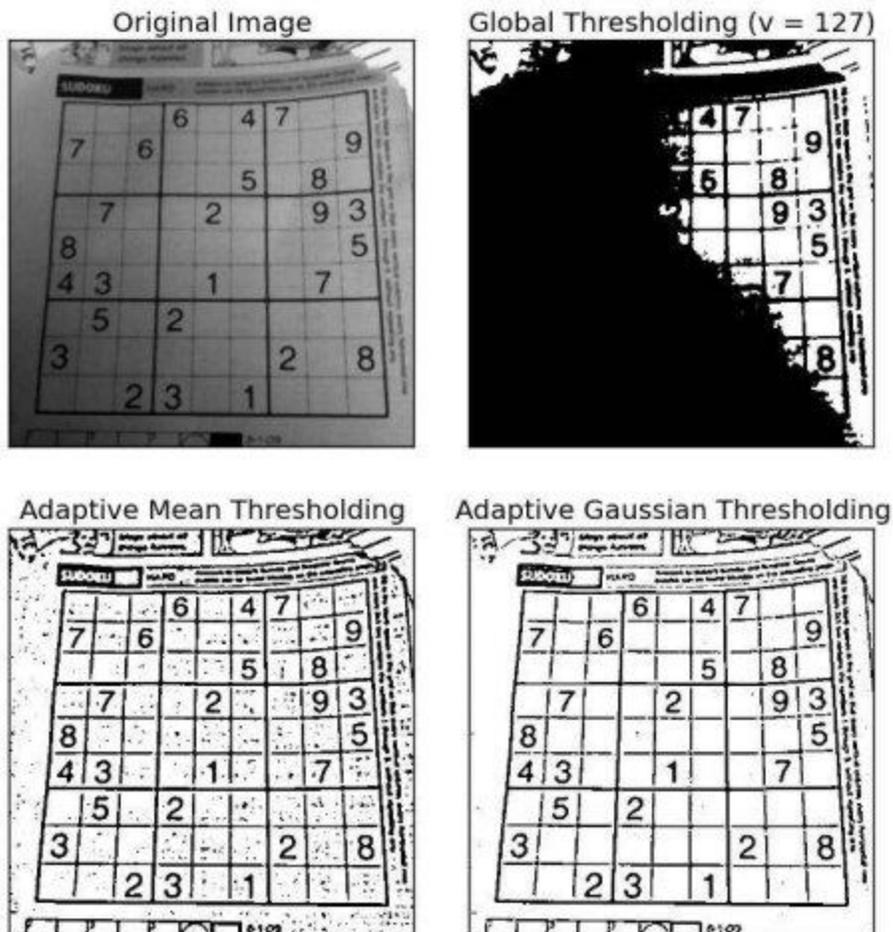


Figura 3.4 Aplicación de algoritmos de umbralización.

(OpenCV.org, 2018)

Para el diseño del sistema de semáforos inteligentes para el control de tráfico vehicular, se utilizará el algoritmo OTSU para los procedimientos que requieran la umbralización de imágenes ya que de acuerdo a las conclusiones de Azhar y Widjanarko, OTSU se comporta mejor en imágenes con histogramas no bimodales, imágenes que no muestran una diferencia contrastante entre el fondo y los objetos del primer plano (Azhar & Widjanarko, 2002).

3.1.3.1 Algoritmo Otsu

El método de Otsu fue inventado por Nobuyuki Otsu en 1979 (OTSU, 1979). La metodología computacional del algoritmo descrito en la revista Cubana de Ciencias Informáticas (López-Portilla Vigil, Menéndez Alonso, & Iglesias Martínez, 2016) establece que:

El algoritmo Otsu se basa en maximizar una medida estadística que se denomina varianza entre clases. Un umbral que produzca la mejor separación entre clases, en términos de los valores de intensidad de estas, será el umbral óptimo. Si se tiene una imagen en escala de grises con L niveles de intensidad de gris ($0, 1, 2, \dots, L-1$), el número de píxeles con nivel i se denota por n_i y el total del número de píxeles es $N = n_0 + n_1 + n_2 + \dots + n_{L-1}$; el histograma normalizado de la imagen se obtiene de la siguiente manera:

$$p_i = \frac{n_i}{N}, p_i \geq 0, \sum_{i=0}^{L-1} p_i = 1 \quad (3.6)$$

Si se supone que los píxeles de la imagen se agrupan en dos clases C_0 y C_1 (objeto y fondo de la imagen, o viceversa) mediante un umbral de nivel t , se tiene que la clase C_0 se corresponde con los píxeles con niveles $[0 \dots t]$ y la clase C_1 se corresponde con los píxeles con niveles $[t+1, \dots, L-1]$. Por tanto, la probabilidad de que un píxel sea asignado a cada una de estas clases viene dada por:

$$w_0 = P_r(C_0) = \sum_{i=0}^t p_i = w(t) \quad (3.7)$$

$$w_1 = P_r(C_1) = \sum_{i=t}^{L-1} p_i = 1 - w(t) \quad (3.8)$$

Los valores medidos de las intensidades correspondientes a las clases C_0 y C_1 respectivamente son:

$$\mu_0 = \sum_{i=0}^t i P_r(i|C_0) = \sum_{i=0}^t \frac{ip_i}{w_0} = \frac{\mu(t)}{w(t)} \quad (3.9)$$

$$\mu_1 = \sum_{i=t+1}^{L-1} i P_r(i|C_1) = \sum_{i=t+1}^{L-1} \frac{ip_i}{w_1} = \frac{\mu_T - \mu(t)}{1 - w(t)} \quad (3.10)$$

Donde: $w(t) = \sum_{i=0}^t p_i$ y $\mu(t) = \sum_{i=0}^t ip_i$ son la suma de las intensidades e intensidad promedio hasta el nivel t respectivamente para la imagen. Mientras que $\mu(t) = \mu(L-1) = \sum_{i=0}^{L-1} ip_i$ es la intensidad media global para toda la imagen. Con las definiciones antes realizadas se cumplen las siguientes relaciones:

$$w_0 \mu_0 + w_1 \mu_1 = \mu_T w_0 + w_1 = 1 \quad (3.11)$$

La varianza de las clases C_0 y C_1 viene dada por:

$$\sigma_0^2 = \sum_{i=0}^t (i - \mu_0)^2 P_r(i|C_0) = \sum_{i=0}^t \frac{(i - \mu_0)^2 P_i}{w_0} \quad (3.12)$$

$$\sigma_1^2 = \sum_{i=t+1}^{L-1} (i - \mu_1)^2 P_r(i|C_1) = \sum_{i=t+1}^{L-1} \frac{(i - \mu_1)^2 P_i}{w_1} \quad (3.13)$$

Con el fin de evaluar la factibilidad del umbral (en el nivel t), se toman las siguientes medidas de criterio discriminante (o medidas de la separabilidad entre clases) utilizadas en el análisis discriminante como se muestra en la ecuación 3.14.

$$\sigma_B^2 = w_0 (\mu_0 + \mu_t)^2 + w_1 (\mu_1 - \mu_T)^2 \quad (3.14)$$

Entonces el problema se reduce a un problema de optimización, el cual consiste en buscar un umbral t que maximice las funciones de objeto (las medidas de criterio) en la ecuación 3.14. Este punto de vista está motivado por el hecho de que la varianza entre clases es una medida de la diferencia entre dos partes. Cuanto mayor sea la varianza entre clases, mayor es la diferencia entre las dos partes. Si la varianza entre clases disminuye, esto implica cometer un error a la hora de extraer un objeto del fondo de la imagen. Como resultado, se hace que la varianza entre clases sea máxima, lo que implica que la probabilidad de error sea mínima. El umbral que contribuya a la mejor separación de las clases en niveles de gris, sería el mejor umbral. Esta es precisamente la esencia del algoritmo de Otsu.

El umbral óptimo t^* que maximiza σ_B^2 se selecciona en la siguiente búsqueda secuencial para los valores de t de 0 a $L-1$ mediante el uso de las cantidades acumuladas simple $w(t) = \sum_{i=0}^t P_i$ y $\mu(t) = \sum_{i=0}^t ip_i$ o explícitamente usando las ecuaciones 3.7, 3.8, 3.9 y 3.10:

$$\sigma_B^2(t) = \frac{[\mu_T w(t) - \mu(t)]^2}{w(t)[1-w(t)]} \quad (3.15)$$

Por lo tanto, el umbral óptimo t^* es:

$$\sigma_B^2(t^*) = \max_{0 \leq t \leq L-1} |\sigma_B^2(t)| \quad (3.16)$$

En la ecuación 3.15, se observa que el cálculo de la varianza entre clases requiere un gran número de operaciones matemáticas de suma, multiplicación y división. En el caso especial de la división, su implementación en hardware es difícil y ocupa gran cantidad de recursos. Una posible solución es el empleo del logaritmo, de esta forma las operaciones de multiplicación y división se convierten en sumas y restas respectivamente. Por lo tanto, la ecuación 3.15 puede ser transformada de la siguiente manera:

$$\log_2 \sigma_B^2(t) = 2 \log_2 [\mu_T w(t) - \mu(t)] - \log_2 w(t) - \log_2 [1 - w(t)] \quad (3.17)$$

La función antilogaritmo no es requerida en este caso, ya que $\sigma_B^2(t)$ y $\log_2 \sigma_B^2(t)$ alcanzan su valor máximo para el mismo umbral. En este caso, el umbral óptimo puede ser obtenido como:

$$\sigma_B^2(t^*) = \max_{0 \leq t \leq L-1} |\log_2 \sigma_B^2(t)| \quad (3.18)$$

En el algoritmo de Otsu, el cálculo de la varianza entre clases tiene que ser iterado para los niveles de L grises, por lo tanto, el punto crítico a la hora de implementar una arquitectura hardware correspondiente a dicho algoritmo se encuentra en el cálculo de la varianza entre clases. De ahí la importancia de que se desarrolle una buena arquitectura para mejorar la eficiencia de este algoritmo.

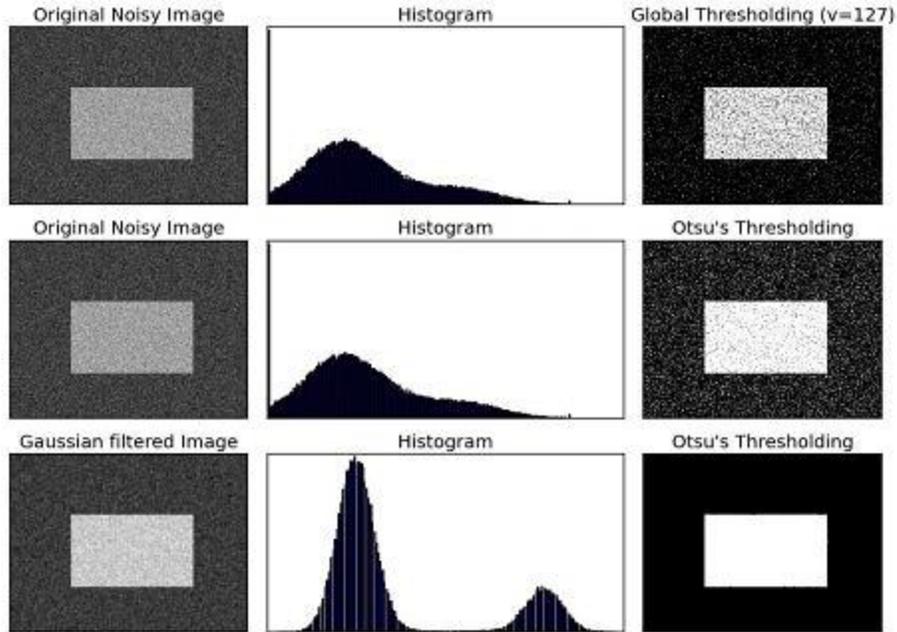


Figura 3.5 Aplicación de algoritmo Otsu.
(OpenCV.org, 2018)

3.1.4 Operaciones binarias con imágenes

Generalmente, este tipo de operaciones son utilizadas después de emplear algoritmos de segmentación, para aplicar filtros u otro procesamiento sobre zonas específicas de la imagen original conservando intactas las regiones que se desea ignorar. Se requieren dos imágenes binarias del mismo tamaño ya que para obtener el resultado se aplican operaciones lógicas bit a bit entre las dos imágenes (excepto para la compuerta lógica NOT donde sólo se requiere una entrada). Dichas operaciones lógicas son: AND, OR, NOT y XOR.

A continuación, se describen cada una de las compuertas antes mencionadas, sean A y B dos pixeles pertenecientes a dos imágenes diferentes, con las mismas coordenadas de posición (x, y) , y Q el valor del pixel resultante para la posición (x, y) .

- AND: Si el valor de los dos pixeles de entrada es 1, entonces, el pixel resultante obtiene el valor de 1 (color blanco).

A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1



*Tabla 3.1 Compuerta lógica AND
(Elaboración propia)*

- OR: El pixel resultante es blanco cuando al menos uno de los dos pixeles de entrada es blanco.

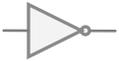
A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1



*Tabla 3.2 Compuerta lógica OR
(Elaboración propia)*

- NOT: El valor del pixel resultante es inverso al valor del pixel de entrada, en otras palabras, si el valor de entrada es negro el resultado es blanco y viceversa.

A	Q
0	1
1	0



*Tabla 3.3 Compuerta lógica NOT
(Elaboración propia)*

- XOR: Si los pixeles de entrada son iguales el resultado es un pixel negro, si los pixeles de entrada son diferentes entre sí, el pixel resultante es color blanco.

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0



Tabla 3.4 Compuerta lógica XOR
(Elaboración propia)

A continuación, se ilustra la aplicación de las compuertas lógicas, el color negro representa las regiones con pixeles de valor 0, el color blanco pixeles con valor 1, las líneas punteadas se muestran solamente para referencia y no forman parte del resultado.

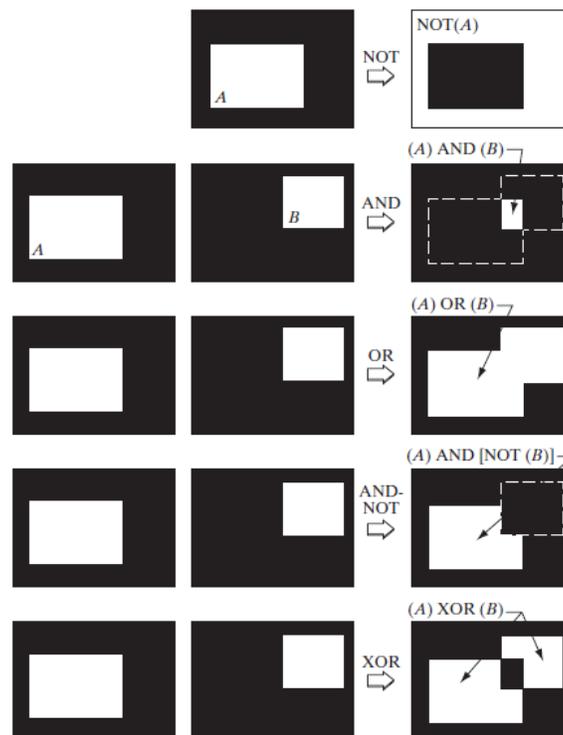


Figura 3.6 Aplicación de operaciones lógicas en imágenes binarias.
(Gonzalez & Woods, 2008)

3.2 Visión artificial

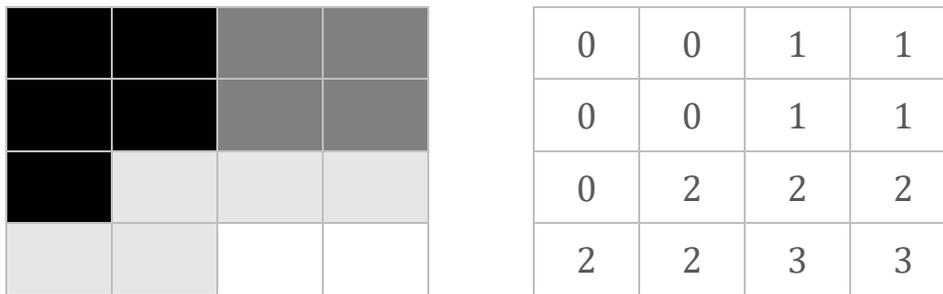
También conocida como visión por computador, tiene como objetivo extraer imágenes del mundo físico, procesarlas a través de un computador y obtener información semejante a la interpretación humana especializada. Es una disciplina científica que incluye métodos para adquirir, procesar, analizar y comprender las imágenes del mundo real con el fin de producir información numérica o simbólica para que puedan ser tratados por un computador (Vélez Serrano, Moreno Díaz, Sánchez Calle, & Esteban Sánchez-Marín, 2003).

Las técnicas de visión artificial simulan a la interpretación humana, el ser humano recibe la información a través de los ojos, la información es enviada al cerebro a través del nervio óptico para ser procesada. Análogamente la primera fase de la visión artificial consiste en capturar imágenes mediante sensores (cámaras digitales). La segunda fase consiste en eliminar información no deseada de la imagen mediante filtros, transformaciones geométricas, etc. La tercera fase consiste en aislar las regiones de interés de la imagen (secciones 3.1.2, 3.1.3, y 3.1.4). La última etapa consiste en la clasificación o reconocimiento de patrones, en esta etapa se pueden distinguir objetos mediante el análisis de atributos que describen a los objetos de estudio.

Los diferentes objetos que aparecen en una imagen se pueden estudiar analizando propiedades como bordes y texturas. Es importante considerar que las técnicas de visión artificial aplicadas a imágenes suelen aplicarse también a videos (secuencia de cierto número de imágenes por segundo). Un ejemplo muy común al trabajar con secuencias de imágenes es identificar las diferencias entre cada una de las imágenes para identificar el movimiento de objetos o para aislar los objetos del fondo de la imagen. Por lo que se requiere un procesamiento de datos veloz que permita clasificar objetos en tiempo real, actualmente existen lenguajes de programación que facilitan estas tareas como C++ o Python (independiente al sistema operativo), también existen bibliotecas como YOLO y OPEN CV con los algoritmos más utilizados para procesamiento de imágenes y clasificación de objetos.

3.2.1 Matriz de co-ocurrencia de niveles de gris.

También denominada GLCM por sus siglas en inglés (grey level co-occurrence matrix), es el histograma que se obtiene a partir de los valores de la escala de grises concurrentes en un desplazamiento dado sobre una imagen, la matriz GLCM es indispensable para el análisis de texturas (Haralick, Shanmugan, & Dinstein, 1973). En la siguiente figura se representa una imagen de tamaño 4x4 donde los niveles de grises corresponden a los valores (0,1,2,3).



*Tabla 3.5 Representación matricial de una imagen en escala de grises.
(Jiménez Guerrero, 2015)*

Para construir la matriz GLCM requiere una orientación (0° , 90° , 45° , 135°) y una distancia de referencia al pixel vecino. La estructura de la matriz resultante deberá considerar todas las posibles combinaciones entre los valores de pixel de referencia y pixel vecino. EL procedimiento para generar la matriz GLCM consiste en iterar sobre cada pixel de la imagen, contabilizar y registrar cada una de las diferentes tuplas que se forman mediante (valor del pixel de referencia, valor del pixel vecino). En cada iteración el pixel vecino se convierte en el pixel de referencia y se obtiene un nuevo pixel vecino desplazándonos el valor de la distancia hacia la orientación definida previamente. La matriz GLCM de la tabla anterior (Tabla 3.5) para los valores de distancia=1 y orientación= 0° queda de la siguiente manera:

0	0	1	1
0	0	1	1
0	2	2	2
2	2	3	3

→

2	2	1	0
0	2	0	0
0	0	3	1
0	0	0	1

*Tabla 3.6 Cálculo de la matriz de co-ocurrencia.
(Jiménez Guerrero, 2015)*

Previo a realizar los cálculos para obtener las características de una textura, la matriz GLCM debe ser simétrica y debe también estar normalizada en términos de probabilidad. Para lograr que sea simétrica se realiza la suma de la matriz transpuesta. Para la tabla anterior (Tabla 3.6) se obtiene como resultado:

4	2	0	0
2	4	0	0
0	0	6	1
0	0	1	2

*Tabla 3.7 Matriz simétrica de co-ocurrencia.
(Jiménez Guerrero, 2015)*

Para normalizar la matriz de co-ocurrencia en términos de probabilidad, se divide el número de veces que se repite un evento entre el total de eventos posibles. Para el caso de la tabla anterior (Tabla 3.7) se divide cada valor entre 24.

0.166	0.083	0.042	0
0.083	0.166	0	0
0.042	0	0.250	0.042
0	0	0.042	0.083

*Tabla 3.8 Matriz normalizada de co-ocurrencia.
(Jiménez Guerrero, 2015)*

A los parámetros que se derivan de la matriz GLCM se les conoce como descriptores de textura, los principales son:

- **Contraste:** este parámetro obtiene un valor alto cuando el contraste es alto, y bajo cuando los valores altos de la matriz GLCM se encuentran cerca de la diagonal.

$$C_1 = \sum_{i,j} (i - j)^2 \cdot C_{i,j} \quad (3.19)$$

- **Energía:** La energía tiende a ser mínima cuando los píxeles de la imagen tienden a ser iguales.

$$C_2 = \sum_{i,j} C_{i,j}^2 \quad (3.20)$$

- **Entropía:** Contrario a la energía, tiende a un valor máximo cuando todos los elementos de la matriz son iguales.

$$C_3 = \sum_{i,j} -C_{i,j} \cdot \log_2[C_{i,j}] \quad (3.21)$$

3.2.2 Algoritmo de Canny

Es un algoritmo desarrollado por John F. Canny en 1986. Se utiliza para detectar contornos en una imagen. Los bordes o contornos son las zonas de píxeles donde ocurre un cambio brusco en los niveles de gris. EL algoritmo consta de 3 etapas: Obtención del gradiente, Supresión no máxima e histéresis de umbral (Canny, 1986).

3.2.2.1 Obtención del gradiente

En este paso se calcula la magnitud y orientación del vector gradiente en cada píxel. Lo primero que se realiza es la aplicación de un filtro gaussiano a la imagen original con el objetivo de suavizar la imagen y tratar de eliminar el posible ruido existente.

El proceso de suavizado se obtiene promediando los valores de intensidad de los píxeles en el entorno de vecindad con una máscara de convolución (Valverde Rebaza, 2007).

$$\frac{1}{273}$$

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

Tabla 3.9 Máscara de convolución para obtener el filtro Gaussiano (Valverde Rebaza, 2007)

3.2.2.2 Supresión no máxima al resultado del gradiente

Este paso consiste en adelgazar el ancho de los bordes obtenidos con el gradiente hasta lograr que los bordes midan un píxel de ancho. Se consideran cuatro direcciones identificadas por las orientaciones de 0° , 45° , 90° y 135° con respecto al eje horizontal. Para cada píxel se encuentra la dirección que mejor se aproxime a la dirección del ángulo de gradiente. El gradiente de una imagen $f(x, y)$ en un punto (x, y) se define como un vector bidimensional dado por la siguiente ecuación (Valverde Rebaza, 2007).

$$G[f(x, y)] = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x} f(x, y) \\ \frac{\partial}{\partial y} f(x, y) \end{bmatrix} \quad (3.22)$$

Siendo un vector perpendicular al borde, donde el vector G apunta en la dirección de variación máxima de f en el punto (x, y) por unidad de distancia, con la magnitud y dirección dadas por:

$$|G| = \sqrt{G_x^2 + G_y^2} = |G_x| + |G_y|, \theta(x, y) = \tan^{-1} \frac{G_y}{G_x} \quad (3.23)$$

Se obtiene E_m a partir de la magnitud de gradiente y E_o a partir de la orientación, de acuerdo a las expresiones anteriores. Posteriormente se observa si el valor de la magnitud de gradiente es más pequeño que al menos uno de sus dos vecinos en la dirección del ángulo obtenida en el paso anterior. De ser así se asigna el valor 0 a dicho píxel, en caso contrario se asigna el valor que tenga la magnitud del gradiente. La salida de este segundo paso es la imagen I_n con los bordes adelgazados, es decir, $E_m(i, j)$, después de la supresión no máxima de puntos de borde (Valverde Rebaza, 2007).

3.2.2.3 Histéresis de umbral a la supresión no máxima

En este paso se aplica una función de histéresis basada en dos umbrales; con este proceso se pretende reducir la posibilidad de aparición de contornos falsos. El proceso consiste en tomar la orientación de los puntos de borde de la imagen y tomar dos umbrales, el primero más pequeño que el segundo. Para cada punto de la imagen se debe localizar el siguiente punto de borde no explorado que sea mayor al segundo umbral. A partir de dicho punto seguir las cadenas de máximos locales conectados en ambas direcciones perpendiculares a la normal del borde siempre que sean mayores al primer umbral. Así se marcan todos los puntos explorados y se almacena la lista de todos los puntos en el contorno conectado. Es así como en este paso se logra eliminar las uniones en forma de Y de los segmentos que confluyan en un punto. Para cada píxel (i, j) se encuentra la dirección d_k que mejor se aproxima a la dirección $E_o(i, j)$, que viene a ser la perpendicular al borde. Si $E_m(i, j)$ es más pequeño que al menos uno de sus dos vecinos en la dirección d_k , al píxel (i, j) de I_n se le asigna el valor 0, $I_n(i, j) = 0$ (supresión), de otro modo $I_n(i, j) = E_m(i, j)$.

Frecuentemente, es común que un cuarto y último paso se realice en el algoritmo de Canny, este paso consiste en cerrar los contornos que pudiesen haber quedado abiertos por problemas de ruido. Un método muy utilizado es el algoritmo de Deriche y Cocquerez. Este algoritmo utiliza como entrada una imagen binarizada de contornos de un píxel de ancho. El algoritmo busca los extremos de los contornos abiertos y sigue la dirección del máximo gradiente hasta cerrarlos con otro extremo abierto.

El procedimiento consiste en buscar para cada píxel uno de los ocho patrones posibles que delimitan la continuación del contorno en tres direcciones posibles. Esto se logra con la convolución de cada píxel con una máscara específica. Cuando alguno de los tres puntos es ya un píxel de borde se entiende que el borde se ha cerrado, de lo contrario se elige el píxel con el valor máximo de gradiente y se marca como nuevo píxel de borde y se aplica nuevamente la convolución. Estos pasos se repiten para todo extremo abierto hasta encontrar su cierre o hasta llegar a cierto número de iteraciones determinado (Valverde Rebaza, 2007).

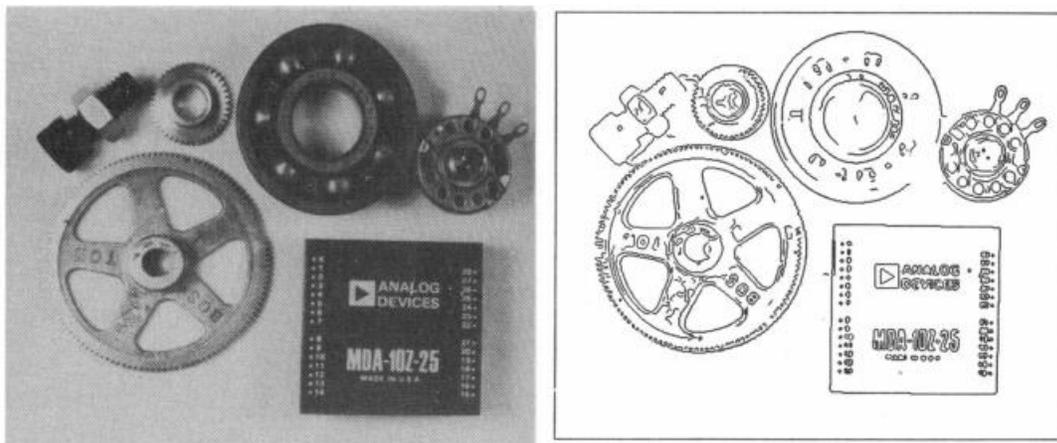


Figura 3.7 Aplicación de algoritmo Canny.

(Canny, 1986)

3.2.3 Algoritmos para substracción de fondo

Los algoritmos para la resta de fondo consisten en separar los elementos de un conjunto de imágenes secuenciales en dos planos: el primero que representa todos los objetos en movimiento y el segundo plano que representa a los objetos estáticos en conjunto con los elementos que conforman el fondo de la imagen. Entre los principales algoritmos se encuentran: Substracción con imagen de referencia, substracción del fotograma anterior, substracción del acumulado de fotogramas anteriores y la substracción KNN (Miguel Lozano, 2016).

3.2.3.1 Substracción con imagen de referencia

Esta técnica es la más simple, consiste en tomar una imagen de fondo fija, es importante que la imagen no contenga objetos ajenos al fondo, a partir de esta se obtienen los objetos en movimiento restando a cada una de las imágenes la imagen del fondo que se tiene como referencia. La desventaja de esta técnica es que es vulnerable a cambios de iluminación por lo que los algoritmos con esta técnica requieren imágenes de escenas perfectamente controladas (Miguel Lozano, 2016).

3.2.3.2 Substracción del fotograma anterior

Semejante al algoritmo anterior, sin embargo, en lugar de restar una imagen fija, a cada imagen nueva se le resta la imagen anterior. Con este algoritmo se solucionan los problemas por cambios de iluminación, pero la desventaja es que si un objeto deja de moverse en la siguiente iteración para a ser parte del fondo (Miguel Lozano, 2016).

3.2.3.3 Substracción del acumulado de fotogramas anteriores

Esta técnica corrige las vulnerabilidades del algoritmo anterior ya que acumula un buffer o un historial de las imágenes de fondo de acuerdo a la siguiente fórmula:

$$u_t = (1 - \alpha)u_{t-1} + \alpha p_t \quad (3.24)$$

Donde p_t es el nuevo valor del píxel, u_{t-1} es la media del fondo en el instante $t-1$, u_t es la nueva media del fondo y α es la ratio de aprendizaje (qué tan rápido olvida las imágenes anteriores) (Miguel Lozano, 2016).

3.2.3.4 Substracción KNN

La mejora que aporta esta técnica es que se basa en el algoritmo de clasificación supervisada “K vecinos más cercanos”, es muy eficiente cuando el número de píxeles que se corresponden con el primer plano es bajo. Se comenzó a utilizar en el 2006 y consiste en clasificar los píxeles para saber si pertenecen al fondo o a un objeto en movimiento, la técnica arroja resultados muy precisos sin embargo su ejecución requiere más tiempo de procesamiento comparado con otras técnicas (Miguel Lozano, 2016).



Figura 3.8 Aplicación de algoritmo Substracción de fondo KNN.

(OpenCV.org, 2018)

3.2.4 Clasificador Haar cascade

La detección de objetos mediante clasificadores de cascada es un método efectivo propuesto por Paul Viola y Michael Jones en su artículo "Rapid Object Detection using a Boosted Cascade of Simple Features" (Viola & Jones, 2001). La clasificación se basa en un algoritmo de aprendizaje automático donde una función de cascada es entrenada mediante muchas imágenes positivas y negativas. Después del entrenamiento el clasificador puede detectar objetos en cualquier otra imagen.

3.2.4.1 Fase de entrenamiento

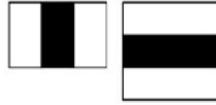
Para poder entrenar un clasificador de cascada se necesita un gran número de imágenes positivas (aquellas imágenes que contienen el objeto que queremos reconocer) igualmente se requiere un conjunto de imágenes negativas (aquellas que no contienen el objeto). El clasificador comienza a extraer características o patrones de contrastes de brillo que se generan en diferentes regiones de cada una de las imágenes mediante funciones de características haar. Los principales patrones a los cuales se les denomina "kernel" se muestran a continuación (OpenCV.org, 2018):

- Característica de borde:



*Figura 3.9 Característica de borde para entrenamiento de haar cascade.
(OpenCV.org, 2018)*

- Característica lineal:



*Figura 3.10 Característica lineal para entrenamiento de haar cascade.
(OpenCV.org, 2018)*

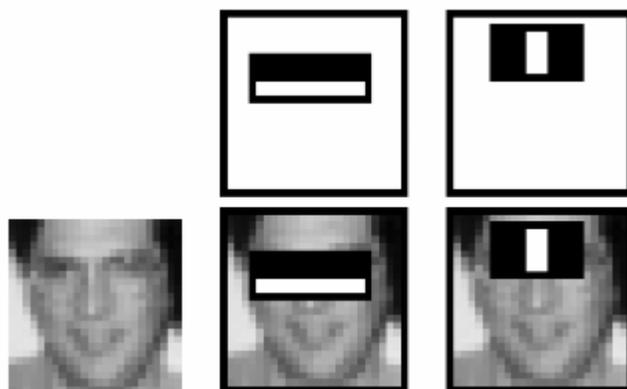
- Característica de cuatro rectángulos:



*Figura 3.11 Característica de cuatro rectángulos para entrenamiento de haar cascade.
(OpenCV.org, 2018)*

Posteriormente todos los tamaños y ubicaciones posibles de cada kernel se utilizan para calcular muchas características. Con una ventana de 24x24 da como resultado más de 160000 características. Para cada cálculo de características se necesita calcular la suma de los píxeles debajo de los rectángulos blanco y negro. Para optimizar el algoritmo se introdujo la imagen integral. Por grande que sea su imagen, reduce los cálculos para un píxel dado a una operación que involucra solo cuatro píxeles.

Entre todas las características que se calculan, la mayoría son irrelevantes. Para el ejemplo de clasificación de rostros la fila superior muestra dos características relevantes. La primera se enfoca en la propiedad de que la región de los ojos a menudo es más oscura que la región de la nariz y las mejillas. La segunda característica se basa en que los ojos son más oscuros que el puente de la nariz. Sin embargo, las mismas ventanas aplicadas a las mejillas o cualquier otro lugar son irrelevantes. Para seleccionar solamente las mejores características se utiliza el algoritmo Adaboost.



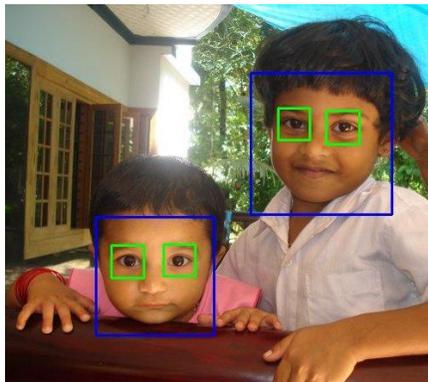
*Figura 3.12 Ejemplo de características relevantes para un clasificador de rostros.
(OpenCV.org, 2018)*

Se aplican todas las características sobre todas las imágenes de entrenamiento. Para cada característica, se busca el mejor umbral que clasificará las caras en positivas y negativas. Al mismo tiempo se descartan las clasificaciones erróneas ya que se les da un mayor peso a las características con un índice de error mínimo. Durante el entrenamiento cada imagen recibe el mismo peso al principio. Después de cada clasificación, se aumentan los pesos de las imágenes mal clasificadas. Luego se realiza el mismo proceso. Se calculan nuevas tasas de error y también nuevos pesos. el proceso continúa hasta que se alcanza la precisión requerida, la tasa de error o se encuentra el número requerido de características.

3.2.4.2 Fase de clasificación

El clasificador resultante es una suma ponderada de los clasificadores débiles obtenidos del proceso de entrenamiento. Se llama débil porque por sí solo no puede clasificar la imagen, pero junto con otros clasificadores débiles se forma un clasificador fuerte. Incluso 200 características proporcionan detección con 95% de precisión (OpenCV.org, 2018).

De esta forma para el ejemplo anterior se pueden reducir 160000 características a solo 6000 relevantes. El concepto de cascada surge cuando en lugar de aplicar todas esas 6000 funciones o características en una ventana, las características se agrupan en diferentes etapas de clasificadores y se aplican una a una. (Normalmente, las primeras etapas contendrán muchas menos características). Si una ventana falla la primera etapa, se desecha y no se consideran las características restantes. Si pasa, se aplica la segunda etapa de características y continua el proceso. La ventana que pasa todas las etapas es una región del objeto buscado (OpenCV.org, 2018).



*Figura 3.13 Reconocimiento facial con haar cascade OPEN CV.
(OpenCV.org, 2018)*

3.2.5 Colisiones en un plano

En el campo de visión artificial es común analizar el comportamiento de los objetos una vez que han sido reconocidos, como se describe en la sección 3.1 de este documento, una imagen puede representarse como un plano bidimensional, por lo tanto, se puede conocer si un objeto dentro de la imagen colisiona con otro o con ciertas zonas de la imagen mediante las coordenadas (x, y) de los píxeles del objeto.

3.2.5.1 Magnitud de un vector

La magnitud de un vector \overrightarrow{PQ} es la distancia entre el punto inicial P y el punto final Q. En símbolos la magnitud de \overrightarrow{PQ} se define como $|\overrightarrow{PQ}|$. Si se conocen las coordenadas del punto P y del punto Q, se puede usar la fórmula 3.25 (distancia entre dos puntos) para encontrar su magnitud (LEHMANN, 1989).

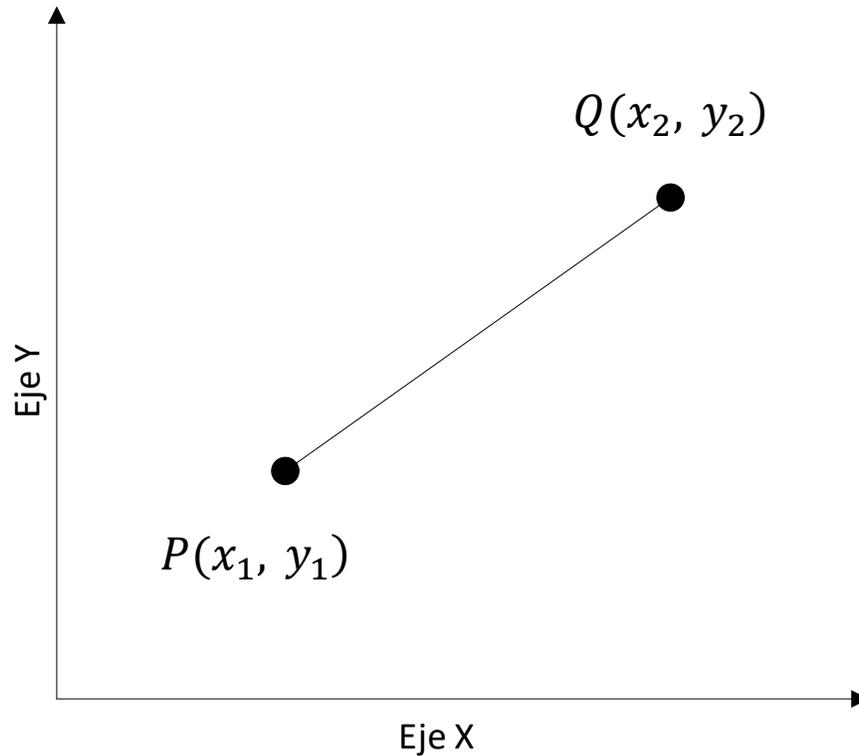


Figura 3.14 Línea recta formada por los puntos P y Q.
(Elaboración propia)

$$|\overrightarrow{PQ}| = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (3.25)$$

3.2.5.2 Colisión entre un punto y un segmento de recta

La colisión más común es cuando un punto P_x en movimiento colisiona con un segmento de línea recta definido por dos puntos fijos P_1 y P_2 . Si la suma de las distancias $|\overrightarrow{P_x P_1}|$ y $|\overrightarrow{P_x P_2}|$ es igual a la magnitud del vector formado por los puntos P_1 y P_2 , entonces el punto P_x colisiona con el segmento, de lo contrario el punto P_x y el segmento se encuentran separados entre sí. En resumen, el punto P_x colisiona con el segmento formado por los puntos P_1 y P_2 si se cumple que:

$$|\overrightarrow{P_x P_1}| + |\overrightarrow{P_x P_2}| = |\overrightarrow{P_1 P_2}| \quad (3.26)$$

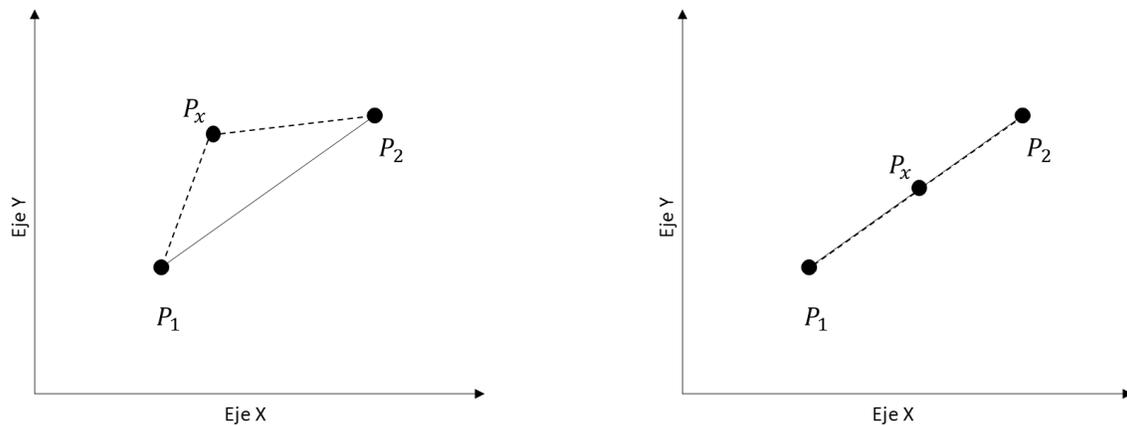


Figura 3.15 colisión entre un punto y un segmento de recta.

(Elaboración propia)

Capítulo 4

Metodología

Existen muchas metodologías para el desarrollo de sistemas, sin embargo, la sustentación de esta tesis se debe al diseño de una propuesta y al desarrollo continuo de un sistema inteligente que permita obtener resultados eficientes en cuanto a objetivos. En este capítulo se da a conocer la metodología “SCRUM”, por ser una metodología modular que permite realizar mejoras continuas a procesos complejos sin comprometer el flujo o afectar el control del proyecto.

4.1 Scrum

El modelo Scrum fue identificado y definido por Ikujiro Nonaka e Hirotaka Takeuchi a principios de los 80, al analizar cómo desarrollaban los nuevos productos las principales empresas de manufactura tecnológica: Fuji-Xerox, Canon, Honda, NEC, Epson, Brother, 3M y Hewlett-Packard (Takeuchi & Nonaka, 1986). En su estudio, Nonaka y Takeuchi compararon la nueva forma de trabajo en equipo, con el avance en formación de melé (scrum en inglés) de los jugadores de Rugby, a raíz de lo cual quedó acuñado el término “scrum” para referirse a ella.

En el libro “Proyecto ágiles con Scrum: flexibilidad, aprendizaje, innovación y colaboración en contextos complejos” (Alaimo, 2013), se describe a scrum de la siguiente manera:

En lugar de proporcionar una descripción completa y detallada de cómo deben realizarse las tareas de un proyecto, Scrum genera un contexto relacional e iterativo, de inspección y adaptación constante para que los involucrados vayan creando su propio proceso.

Esto ocurre debido a que no existen ni mejores ni buenas prácticas en un contexto complejo. Es el equipo de involucrados quien encontrará la mejor manera de resolver sus problemáticas. Este tipo de soluciones serán emergentes.

El equipo de desarrollo se encuentra apoyado en dos roles: el ScrumMaster y el Product Owner. El ScrumMaster es quien vela por la utilización de Scrum, la remoción de impedimentos y asiste al equipo a que logre su mayor nivel de performance posible. Puede ser considerado un coach o facilitador encargado de acompañar al equipo de desarrollo. El Product Owner es quien representa al negocio, stakeholders, cliente y usuarios finales. Tiene la responsabilidad de conducir al equipo de desarrollo hacia el producto adecuado.

El progreso de los proyectos que utilizan Scrum se realiza y verifica en una serie de iteraciones llamadas Sprints. Estos Sprints tienen una duración fija, pre-establecida de no más de un mes. Al comienzo de cada Sprint el equipo de desarrollo realiza un compromiso de entrega de una serie de funcionalidades o características del producto en cuestión.

Al finalizar el Sprint se espera que estas características comprometidas estén terminadas, lo que implica su análisis, diseño, desarrollo, prueba e integración al producto. En este momento es cuando se realiza una reunión de revisión del producto construido durante el Sprint, donde el equipo de desarrollo muestra lo construido al Product Owner y a cualquier stakeholder interesado en participar. El feedback obtenido en esta reunión puede ser incluido entre las funcionalidades a construir en futuros Sprints.

4.1.1 Principios de Scrum

Scrum es el modelo más utilizado dentro de las Metodología Ágiles. Muchos de los valores y principios del Manifiesto Ágil tienen su origen en Scrum, a continuación, se revisan los valores del manifiesto ágil desde la perspectiva de Scrum (Alaimo, 2013):

4.1.1.1 Individuos e interacciones por sobre procesos y herramientas

Scrum se apoya en la confianza hacia las personas, sus interacciones y los equipos. Los equipos identifican lo que hay que hacer y toman la responsabilidad de hacerlo, removiendo todos los impedimentos que encuentren en su camino y estén a su alcance. Los equipos trabajan en conjunto con otras partes de la organización cuando los impedimentos están fuera de su ámbito de control.

4.1.1.2 Software funcionando por sobre documentación exhaustiva

Scrum requiere que al final de cada Sprint se entregue un producto funcionando. La documentación es entendida, en Scrum, como un producto intermedio sin valor de negocio. Los equipos pueden documentar tanto como crean necesario, pero ninguno de estos documentos puede ser considerados como el resultado de un Sprint. El resultado de un Sprint es, nuevamente, el producto funcionando. El progreso del proyecto se mide en base al producto funcionando que se entrega iterativamente.

4.1.1.3 Colaboración con el cliente por sobre la negociación de contratos

El Scrum Product Owner es el responsable de la relación que existe con los usuarios finales, stakeholders y áreas de la organización que van a obtener el beneficio del producto. El Scrum Product Owner es parte del Equipo Scrum y trabaja colaborativamente con el resto de los individuos dentro del equipo para asegurarse que el producto construido tenga la mayor cantidad posible de valor al final de cada iteración.

4.1.1.4 Respuesta al cambio por sobre el seguimiento de un plan

Scrum, por diseño, se asegura que todo el mundo dentro de un equipo tenga toda la información necesaria para poder tomar decisiones informadas sobre el proyecto en cualquier momento. El progreso es medido al final de cada Sprint mediante software funcionando y la lista de características pendientes está visible continuamente y para todos los miembros. Esto permite que el alcance del proyecto cambie constantemente en función de la retroalimentación provista por los stakeholders. Fomentar el cambio es una ventaja competitiva.

4.1.2 Elementos y roles de Scrum

Scrum es una metodología empleada para trabajar en equipo con reuniones a lo largo de todo el proyecto, esta metodología está preparada para sufrir cambios inesperados durante la implementación del proyecto. Se plantea de inicio un objetivo el cual se quiere obtener como resultado, cabe mencionar que para realizar este tipo de metodología se debe ser autónomo y contar con autoorganización. Se establecen intervalos de tiempo que se denominan Sprints en los cuales se debe presentar el avance del proyecto. De ser necesario, los integrantes se transmiten el conocimiento entre ellos debido a algún atraso, por cuestiones de experiencia, las actividades a realizar se deben desglosar en tareas pequeñas, específicas y claras.

A cada una de las tareas se le asigna un número de horas dependiendo de la experiencia de la persona que toma dicha tarea, se realizan reuniones diarias donde los integrantes deben exponer básicamente tres puntos: ¿Qué se ha hecho?, ¿Qué se planea hacer y si existen ¿Cuáles son los impedimentos para lo que se planea hacer? (Trigas Gallego, 2012).



*Figura 4.1 Iteraciones que definen el ciclo de vida de un proyecto.
(Trigas Gallego, 2012)*

Existe una planificación inicial de un Back Log el cual es un documento donde se deben agregar uno a uno los requisitos del sistema definido por prioridades, el backlog se puede ir modificando dependiendo de las prioridades del Product Owner, además, se debe definir un Sprint inicial “sprint 0” en el cual se agregan los primeros objetivos del proyecto a realizar por parte de los integrantes.

Para proyectos grandes que involucran varios equipos Scrum, se realizan también en esta etapa, reuniones que permitan juntar a estos equipos y discutir para revisar avances, dependencias e impedimentos en el desarrollo del proyecto. Adicional a esta etapa en un Sprint Review Meeting, se lleva a cabo el proceso donde el Equipo Scrum demuestra el Sprint Deliverable al Propietario del producto y a los Socios relevantes. Igualmente, el Scrum Master y el Equipo Scrum se reúnen para discutir las lecciones aprendidas a lo largo del Sprint, información que se documenta en una sección de lecciones aprendidas que pueden aplicarse a los futuros Sprints.

Los elementos y los roles que componen la metodología Scrum marcan una gran ventaja sobre otras metodologías como “cascada” en proyecto escalables o de gran magnitud al permitir realizar cambios y adecuaciones sin afectar el costo del proyecto.



Figura 4.2 Ventaja de Scrum contra metodologías de cascada.

(Trigas Gallego, 2012)

4.1.2.1 Product Owner

El Product Owner es la persona responsable del éxito del producto desde el punto de vista de los stakeholders. El Product Owner se focaliza en maximizar la rentabilidad del producto. La principal herramienta con la que cuenta para poder realizar esta tarea es la priorización. De esta manera puede reordenar la cola de trabajo del equipo de desarrollo para que éste construya con mayor anticipación las características o funcionalidades más requeridas por el mercado o la competitividad comercial. Otra responsabilidad importante del Product Owner es la gestión de las expectativas de los stakeholders mediante la comprensión completa de la problemática de negocio y su descomposición hasta llegar al nivel de requerimientos funcionales.

4.1.2.2 Scrum Master

Es la persona responsable de que la metodología se cumpla y comprobar que su funcionamiento sea el esperado, se debe encargar de solucionar los inconvenientes que se vayan suscitando durante el desarrollo del proyecto. El ScrumMaster es el Coach del equipo y es quien lo ayuda a alcanzar su máximo nivel de productividad posible. Se espera, además, que el ScrumMaster acompañe al equipo de trabajo en su día a día y garantice que todos, incluyendo al Product Owner, comprendan y utilicen Scrum de forma correcta.

4.1.2.3 Equipo de Desarrollo

El equipo de desarrollo está formado por todos los individuos necesarios para la construcción del producto en cuestión, es el único responsable por la construcción y calidad del producto, es auto-organizado. Esto significa que no existe un líder externo que asigne las tareas ni que determine la forma en la que serán resueltos los problemas.

Es el mismo equipo quien determina la forma en que realizará el trabajo y cómo resolverá cada problemática que se presente.

4.1.2.4 Product Backlog

El Backlog del Producto es básicamente un listado de ítems (Product Backlog Ítems, PBIs) o características del producto a construir, mantenido y priorizado por el Product Owner. Es importante que exista una clara priorización, ya que es esta priorización la que determinará el orden en el que el equipo de desarrollo transformará las características (ítems) en un producto funcional acabado.

ID	Prioridad	Descripción	Hrs	Asignado a
1	alta	Definición de tema de tesis.	100	LA, FM
2	normal	Investigación de sistemas de semáforos inteligentes aplicados.	50	LA, FM
3	normal	Definición de lenguaje de programación para el proyecto.	8	LA, FM
4	alta	Aprendizaje del lenguaje.	50	LA, FM
5	alta	Instalación de Python, componentes y configuración.	100	LA
6	normal	Investigación de Open CV.	10	LA, FM
7	alta	Pruebas con los Algoritmos.	200	FM
12	normal	Escala de grises.	10	LA
13	normal	Binarización y segmentación.	1	FM
14	normal	Umbralización Otsu.	1	FM
15	normal	Operaciones binarias con Imágenes (AND).	1	LA
16	normal	Detector de bordes Canny.	1	FM
17	normal	Algoritmo para obtener matriz GLCM.	6	LA
18	normal	Análisis de textura (energía y entropía) mediante matriz GLCM.	6	LA
19	normal	Detección de colisiones entre un punto y una recta definida.	1	LA
20	normal	Substracción de fondo KNN.	1	LA
21	alta	Entrenamiento Haar Cascade	50	LA
22	alta	Clasificación haar cascade.	50	FM
23	normal	Transmitir video de un dispositivo a otro mediante streaming.	1	FM
24	normal	Comunicación mediante sockets.	1	FM
25	normal	Sincronización de semáforos.	20	FM,
26	normal	Control de luces con micro controlador.	100	LA

*Tabla 4.1 Product Backlog del proyecto.
(Elaboración propia)*

4.1.2.5 Sprint Backlog

El Sprint Backlog es el conjunto de PBIs que fueron seleccionados para trabajar en ellos durante un cierto Sprint, conjuntamente con las tareas que el equipo de desarrollo ha identificado que debe realizar para poder crear un incremento funcional potencialmente entregable al finalizar el Sprint.

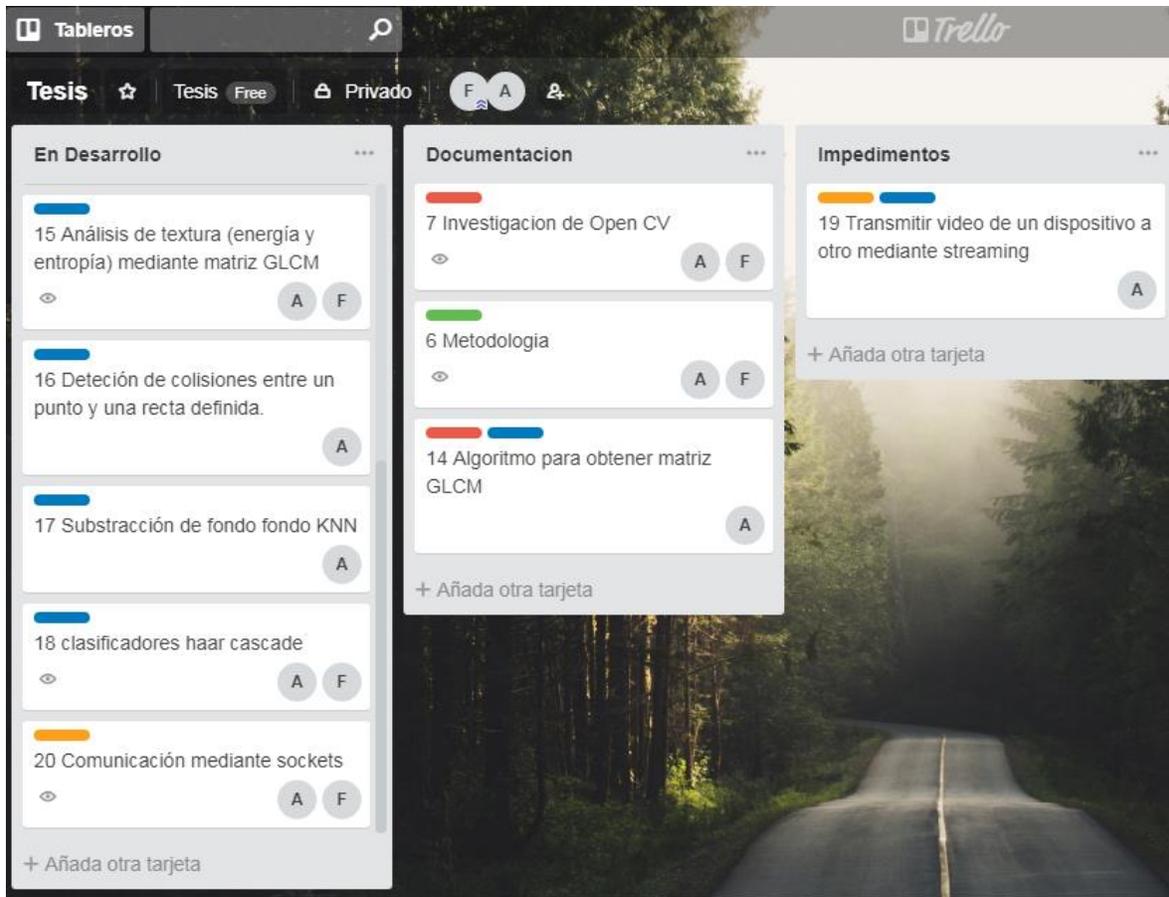
4.1.2.6 Incremento funcional potencialmente entregable

El resultado de cada Sprint debe ser un incremento funcional potencialmente entregable, un producto que está siendo construido de manera evolutiva y que crece con cada Sprint y que posee características suficientemente validadas como para ser desplegado en producción.

4.1.3 Aplicación

Para este proyecto, a partir de la Tabla 4.1 se comenzó a realizar un Sprint cada 2 semanas para analizar el avance de las actividades, en caso de que no se lograran cumplir en tiempo, se fue ajustando la prioridad y la asignación de cada actividad. El tablero Scrum muestra los procesos que se han seguido durante el desarrollo de todo el proyecto, cada integrante del equipo debe ir moviendo las fichas a los estatus marcados en la parte superior de cada columna según su avance. Cuando existe algún impedimento por motivos ajenos al proyecto donde se compromete el avance del proyecto, se genera una notificación dirigida a cada miembro para proponer una solución y así reducir riesgos en los tiempos de entrega.

A continuación, se muestra el tablero de Scrum que se creó con la tecnología Trello (software especializado para proyectos con metodologías Scrum) para la aplicación de la metodología.



En la actualidad el proyecto se encuentra funcional y en etapa de pruebas para poder ser presentado y evaluado por el jurado el cual es considerado bajo el nombre de cliente dentro de esta metodología, así mismo los autores de este proyecto como desarrolladores tenemos el rol de product owner, de la misma forma, entendemos las necesidades del proyecto y lo que debe finalizar como un entregable funcional para nuestro cliente. Los entregables que resultan de cada Sprint son probados por el director de Tesis. Finalmente, la etapa de Lanzamiento es el presente documento de Tesis donde el producto ya es funcional, puede ser testeado e incluso se pueden aportar mejoras como retrospectivas del mismo, preparar futuras implantaciones y promover nuevas líneas de investigación.

Capítulo 5

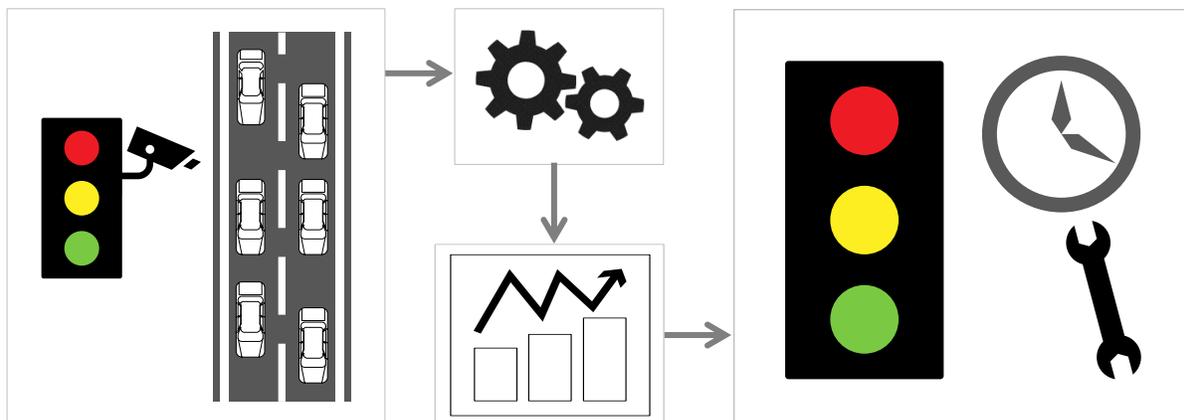
Diseño e implementación

En este capítulo se concentran las especificaciones previas a la construcción del código, se establece el modelo, se presenta la arquitectura y se describe la comunicación entre cada uno de los elementos que conforman el sistema de semáforos inteligentes para el control de tráfico vehicular.

5.1 Descripción general

El diseño del sistema “semáforos inteligentes para el control de tráfico vehicular” se realizó con base en la siguiente premisa: “Si el volumen del flujo vehicular que circula en las avenidas es variable, entonces el ciclo de los semáforos que controlan el tráfico vehicular también debe ser variable y debe poder adaptarse al nivel de demanda”. El sistema está diseñado para ser configurable, flexible y escalable. Esto significa que se reducen las limitaciones para que el sistema sea implementado en cualquier tipo de intersección.

Para lograr el objetivo se empleó una arquitectura donde a partir de la captura de imágenes se ejecutan procesos en tiempo real mediante algoritmos de visión artificial, los cuales arrojan interpretaciones que, a su vez, se comparan automáticamente para que finalmente, el sistema elija los intervalos de tiempo adecuados para la duración del ciclo en los semáforos.



*Figura 5.1 Descripción general del sistema.
(Elaboración propia)*

5.2 Modelo vista controlador

El modelo vista controlador es un patrón para desarrollo de aplicaciones que se encarga de mantener una división de 3 aspectos: los datos de la aplicación, las interfaces gráficas de usuario y la parte lógica o de control. Los patrones de Diseño son soluciones a problemas muy típicos y con mucha recurrencia durante el desarrollo de cualquier aplicación dentro del contexto de la programación, por buenas prácticas se debe considerar un término llamado “programación Solid” la cual consiste en 5 reglas:

- **Responsabilidad Simple:** Cada una de las clases que componen el sistema deben estar desarrolladas con un fin concreto y simple, no es recomendable codificar métodos reutilizables a lo largo del desarrollo ya que puede ocurrir que al final cada clase termine realizando tareas que no le corresponden.
- **Abierto o Cerrado:** el desarrollo de cualquier aplicación debería ser abierto para extenderse, pero cerrado para poder modificar lo ya existente es decir que si se construyen clases o métodos deben ser creados de forma que se puedan extender sin poder modificar el código de lo que ya existe.
- **Sustitución Liskov:** se deberían crear todas las clases derivadas para poder ser tratadas como la clase base de la que derivan.

- Segregación de Interfaces: Se mantiene la misma idea de la responsabilidad simple solo que hablando en esta vez de interfaces y estas a su vez sean utilizadas para la definición de métodos abstractos, busca tener tantas interfaces sean necesarias pero que definan los menos métodos posibles en vez de lo contrario.
- Inversión de Dependencias: busca que el sistema sea completamente desacoplado es decir que las clases interactúen entre sí sin conocer exactamente las clases de nivel interno un patrón que cumple directamente con este principio es el de Inyección de dependencias.

A continuación, se definen los elementos que constituyen el patrón modelo vista controlador:

5.2.1 Modelo

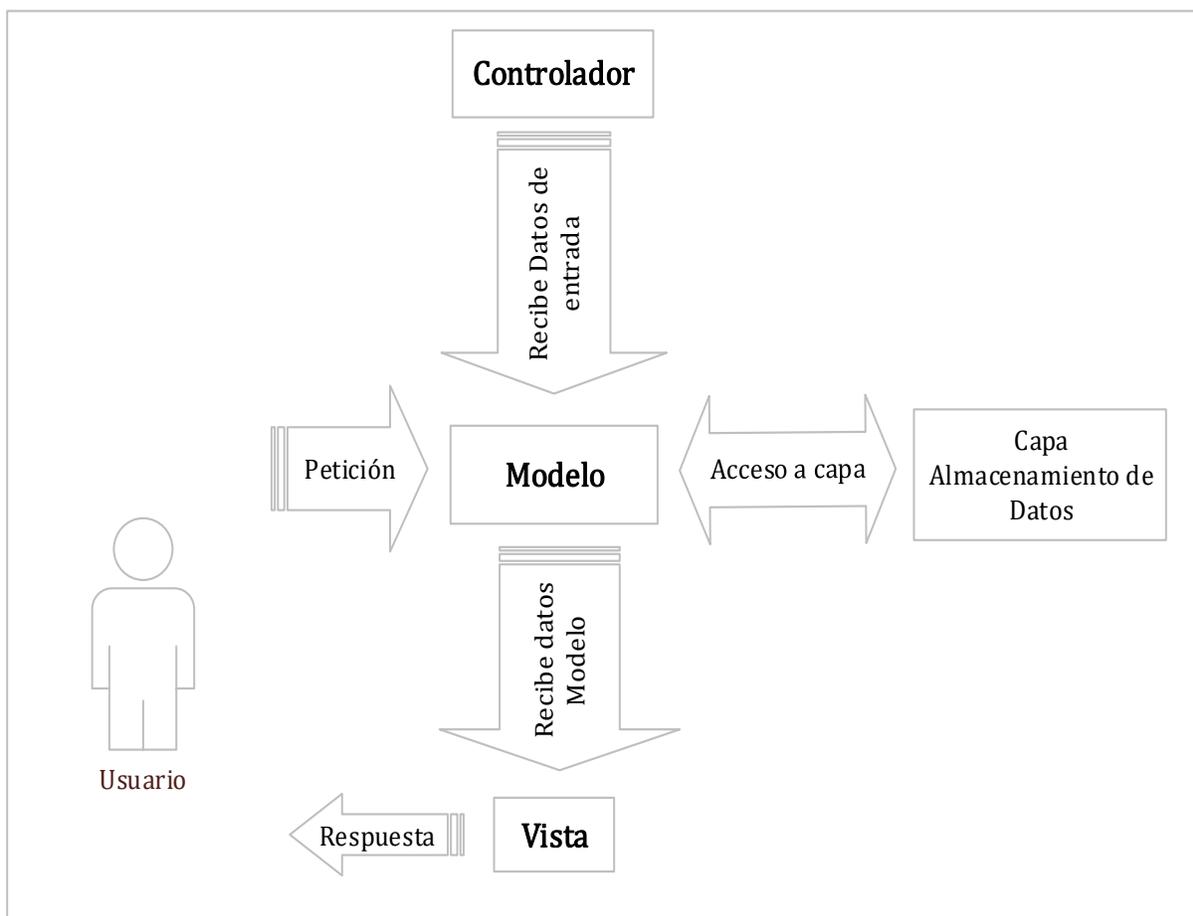
Es responsable de la recuperación de datos convirtiéndolos en conceptos significativos para la aplicación, así como su procesamiento, validación, asociación y cualquier otra tarea relativa a la manipulación de dichos datos. De una manera más clara es el mecanismo que accede al repositorio donde se almacenan los datos, normalmente por regla de buenas prácticas se recomienda sea independiente del sistema de almacenamiento (Base de Datos).

5.2.2 Vista

Es la forma en la que se visualiza la información, también son los componentes con los que se está operando el sistema. Cuando el usuario consulta datos al sistema la vista es la encargada de obtener los datos del modelo y mostrarlos según las necesidades del cliente. La vista hace una presentación de los datos del modelo estando separada de los objetos del modelo.

5.2.3 Controlador

La capa del controlador gestiona las peticiones de los usuarios. Es responsable de responder la información solicitada con la ayuda tanto del modelo como de la vista. Esta capa es la encargada de ser el intermediario entre el usuario y los datos del sistema, tomando el control de los datos para llevar acabo los flujos del software, así mismo se encarga de hacer trasformaciones dependiendo de las Vistas. Se encuentra orientado a los eventos que pueden producirse mientras se está operando el sistema por ejemplo clics, actualización de formularios, o actualización de la información del sistema desde el lado del usuario.



*Figura 5.2 Modelo Vista Controlador.
(Elaboración propia)*

5.3 Requerimientos funcionales

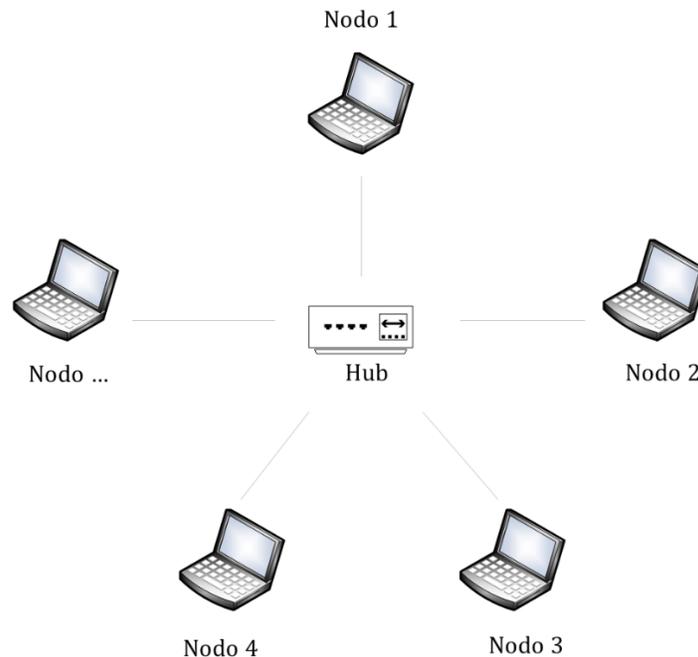
- Cámaras de captura digital para imágenes y video. (recomendables cámaras de alta sensibilidad con estabilizadores anti vibración).
- Placas Arduino o Raspberry Pi con relevadores (1 relé por cada luz del semáforo que se desee controlar).
- Equipo con sistema operativo Linux, procesador de 64 bits y librería OPENCV instalada.
- Infraestructura básica de red (inalámbrica o cableada).

5.4 Requerimientos no funcionales

- Desarrollo del sistema:
 - Distribución Anaconda Python.
 - Entorno de desarrollo con soporte para lenguaje Python.
PyCharm JetBrains (Recomendable).

5.5 Topología de red

Para la comunicación del sistema se utiliza una red en estrella ya que este tipo de red permite agregar equipos fácilmente, reconfigurar nodos rápidamente, además, facilita la prevención y corrección de fallas.



*Figura 5.3 Diagrama de red en estrella.
(Elaboración propia)*

5.6 Estructura física del sistema

La estructura general del sistema se conforma por un ordenador principal en el cual se ejecutan todos los procesos de clasificación, reconocimiento y toma de decisiones respecto a los intervalos de tiempo del semáforo. Segundo, se conforma por nodos esclavos o escuchadores los cuales se encuentran en espera de instrucciones, por ejemplo, instrucciones para activar o desactivar interruptores que controlan las luces del semáforo o instrucciones para transmitir una imagen procedente de la cámara digital. El número de nodos depende del número de arterias viales y sentidos que interceptan en el cruce que se desee controlar. Para controlar dos sentidos se proponen dos nodos esclavos (A y B) y un nodo central como se muestra en la siguiente figura.



*Figura 5.4 Estructura física del sistema.
(Elaboración propia)*

Cada nodo esclavo (A y B) contiene un micro-ordenador (raspberry pi) conectado a la cámara digital y a un circuito eléctrico con un relevador para cada luz del semáforo que se desea manipular. La Raspberry Pi es una pequeña computadora considerada también como un microcontrolador que tiene origen en Reino Unido, es de muy bajo costo comparada con otras en el mercado, tiene integrados los dispositivos de hardware necesarios para su funcionamiento y la opción de añadir otros componentes externos como sensores.

Usualmente se utiliza para objetivos académicos y de investigación, es capaz de soportar sistemas operativos optimizados para su hardware como Linux ARM, Debían, Fedora y ArchLinux. Cuenta con pines de entrada y salida digitales configurables para propósito general.

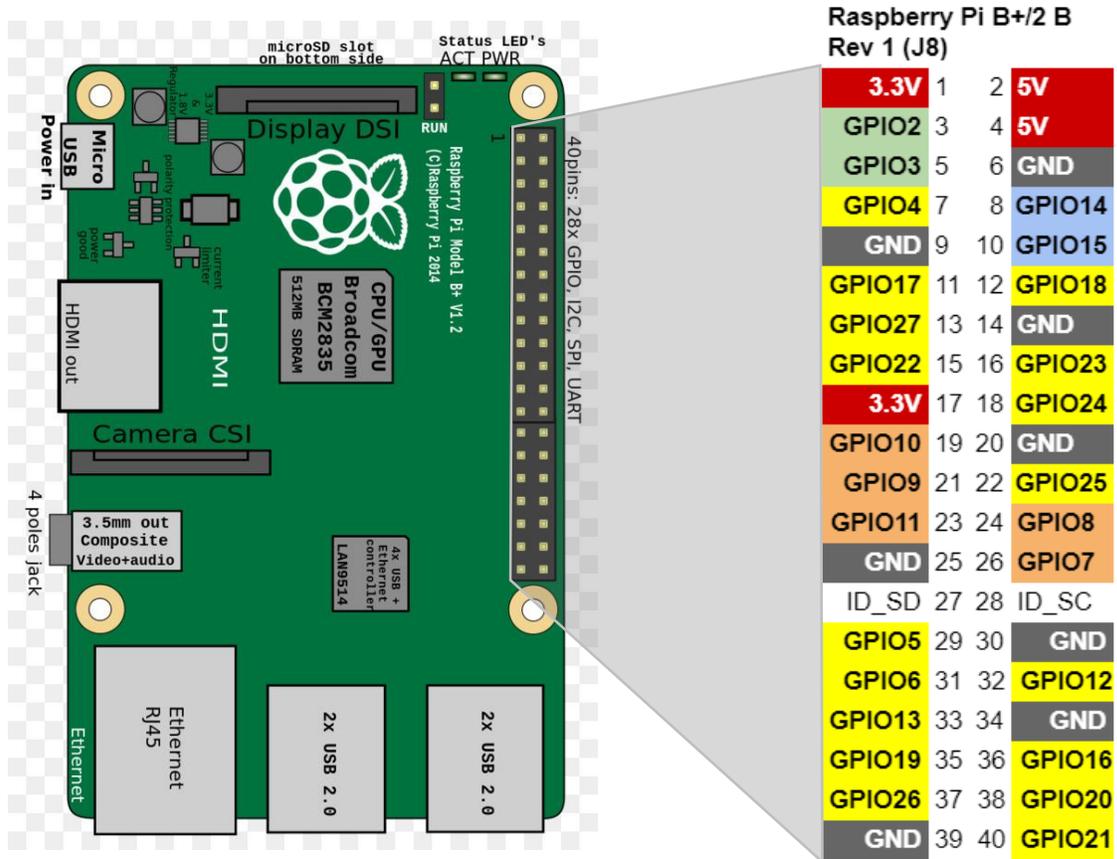


Figura 5.5 Entradas y Salidas Raspberry Pi.
(Raspberry Pi Foundation, 2018)

5.7 Estructura lógica del sistema

La arquitectura general del algoritmo principal obedece el siguiente flujo:

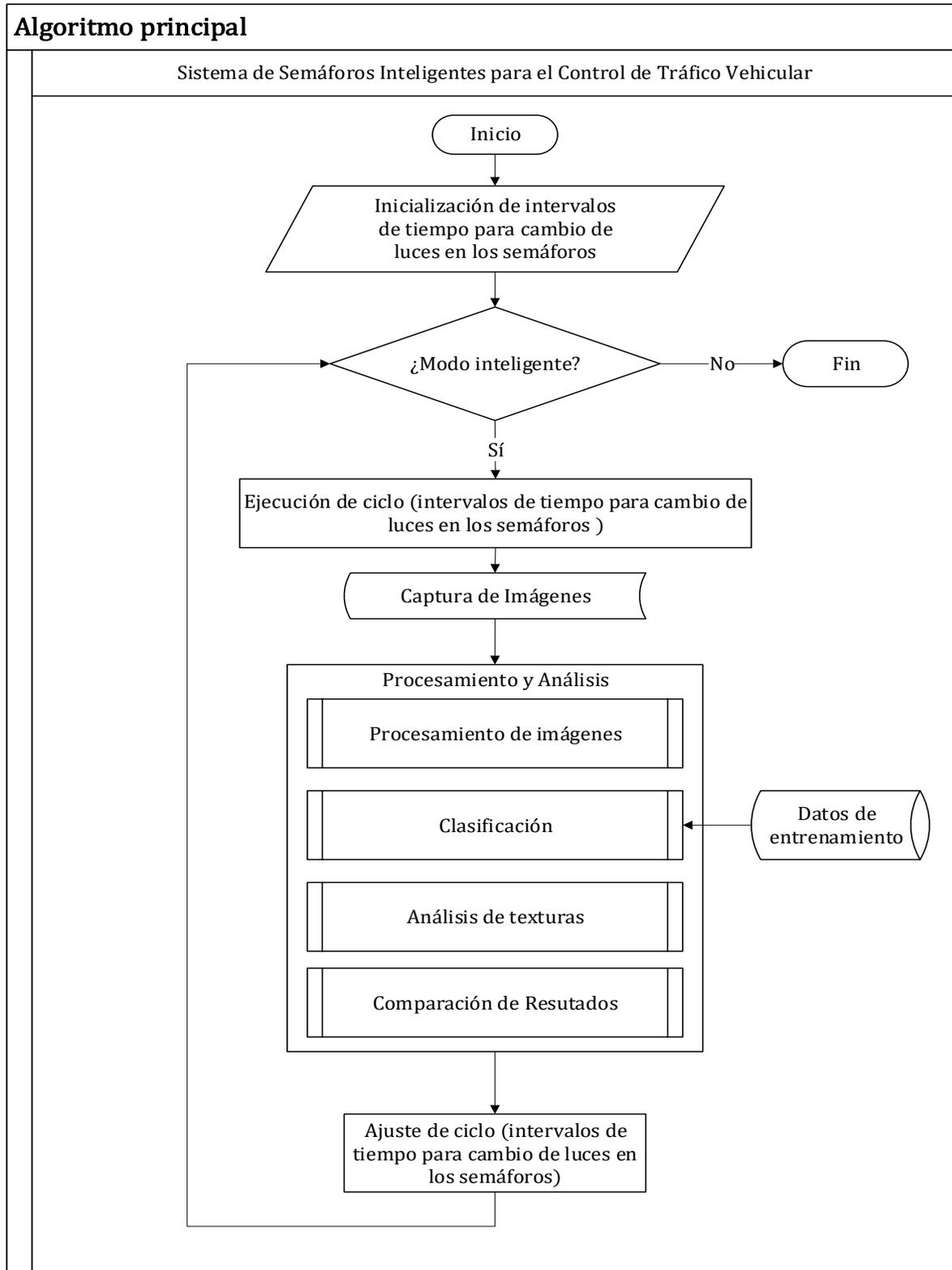


Figura 5.6 Diagrama de flujo de algoritmo principal.
(Elaboración propia)

El funcionamiento del sistema depende directamente del reconocimiento de vehículos por medio de un clasificador de cascada, por lo tanto, el entrenamiento del clasificador es la etapa más importante si se pretende obtener resultados óptimos. En el entrenamiento se deben contemplar las condiciones de un ambiente no controlado para que los resultados no se vean afectados por agentes externos, por ejemplo, si se desea que el sistema controle los semáforos de una intersección durante la noche, es necesario realizar un entrenamiento durante horarios nocturnos para que el clasificador aprenda a reconocer patrones significativos en condiciones de poca iluminación ambiental, patrones como por ejemplo la proyección de luces provenientes de los faros en los vehículos.

En la sección 3.2.4.1 de este documento se describe que para realizar el entrenamiento de un clasificador son necesarios dos conjuntos, uno de imágenes positivas y otro de imágenes negativas, por lo tanto, la propuesta de diseño para este sistema consiste también en automatizar la recolección de imágenes para facilitar el entrenamiento. La recolección automática de imágenes permite una etapa de aprendizaje constante para el sistema, de tal modo que mientras más se utilice, mayor será el volumen de información almacenada que permitirá generar nuevos patrones de reconocimiento, y así, en futuras implementaciones se obtendrá un reconocimiento más preciso en la etapa de clasificación.

El proceso general para la recolección automática de imágenes cumple con el siguiente procedimiento.

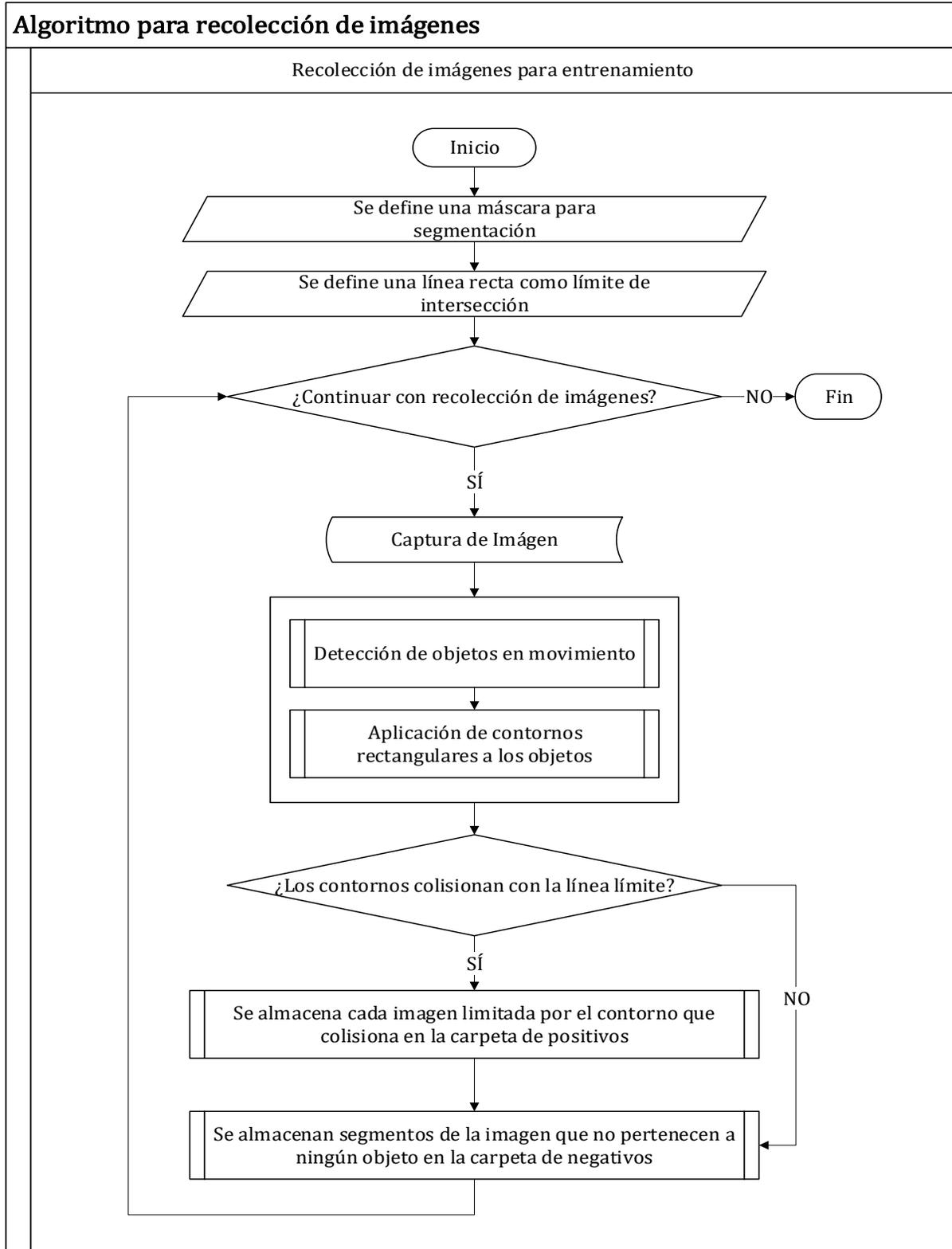


Figura 5.7 Diagrama de flujo de algoritmo para recolección de imágenes
(Elaboración propia)

5.8 Diseño de Interfaces gráficas

La propuesta de diseño del sistema de semáforos inteligentes contempla una automatización de procesos donde no interviene ningún tipo de control humano, por lo tanto se descarta el uso de interfaces gráficas durante el funcionamiento del sistema, sin embargo, las interfaces gráficas de usuario son necesarias en las etapas de configuración, implementación, pruebas e implantación. El diseño de las interfaces gráficas de usuario se basa en dos modelos de ventana, uno de ventana principal para las tareas propias a la gestión del sistema y otro de ventanas secundarias para visualización de información adicional. En la siguiente figura se muestran ambos modelos de ventana.



Figura 5.8 Diseño de interfaces gráficas de usuario.

(Elaboración propia)

5.9 Descripción detallada del sistema

A continuación, se describe el diseño de las etapas y los procesos que conforman el sistema de semáforos inteligentes para el control de tráfico vehicular:

5.9.1 Inicialización

En la etapa de inicialización se define la duración del ciclo igual a la duración con la que opera el semáforo convencionalmente, se cargan los archivos necesarios para la clasificación y se establece la comunicación hacia cada uno de los nodos con base a la arquitectura antes mencionada (secciones 5.5 y 5.6).

5.9.2 Condición para modo inteligente

Es un proceso de validación en la que se le indica al sistema el modo de operación. El modo inteligente implica ejecutar los procesos de visión artificial y de ajuste en la duración del ciclo de los semáforos. El modo convencional ignora todos los procesos inteligentes del sistema, simplemente ejecuta una sincronización con duración de intervalos y ciclos fijos (como operan los semáforos comúnmente).

5.9.3 Ejecución de ciclo

Es un proceso en el cual el sistema realiza pausas cronometradas con base en los intervalos de tiempo previamente definidos. Durante cada iteración del ciclo, las luces de cada semáforo recorren solamente una secuencia completa, por ejemplo: (verde – amarillo – rojo) o (rojo-vede-amarillo).

La siguiente figura ilustra el ciclo para dos semáforos A y B, en este ejemplo la ejecución de cada iteración dura un total de 2 minutos (mientras que no se realice algún ajuste de ciclo).

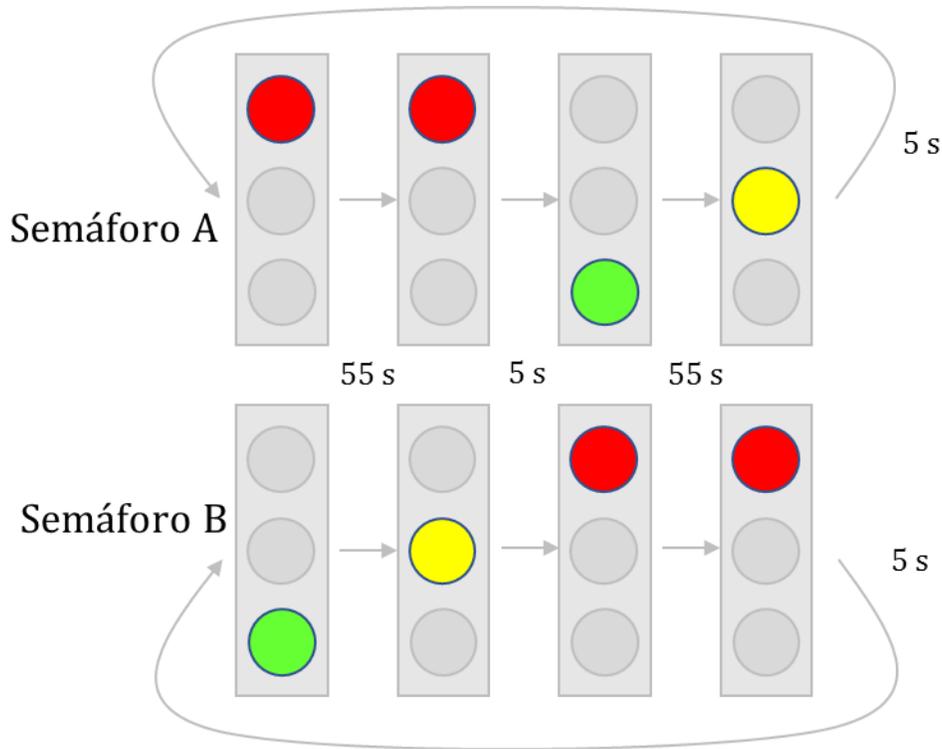


Figura 5.9 Ejecución de ciclo para dos semáforos A y B.
(Elaboración Propia)

5.9.4 Captura de Imágenes

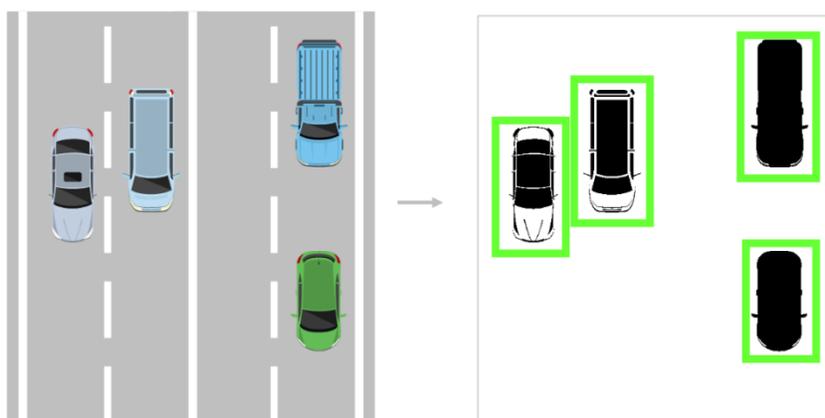
El ordenador principal envía una señal por medio de una conexión mediante sockets a cada uno de los nodos, como resultado, cada nodo envía como respuesta una imagen de lo que ocurre en ese momento mediante una cámara digital, por lo tanto, la comunicación entre el nodo y el ordenador principal es bidireccional. Es importante que el ángulo de visión de la cámara abarque un panorama casi completo de la avenida en el punto e intersección además de que la imagen debe mostrar una resolución adecuada y acorde a la distancia de los autos.

5.9.5 Procesamiento de Imágenes

Dependiendo del tipo de cámara digital que se llegue a utilizar en el procedimiento anterior (sección 5.9.4), el procesamiento de imagen consiste en aplicar filtros para mejorar la calidad o, según sea el caso, aplicar máscaras para ignorar regiones que no son de interés, por ejemplo, cuando el ángulo de visión de la cámara abarca zonas de otras avenidas o estacionamientos ubicados en camellones al aire libre, etc. Los algoritmos empleados para esta tarea se describen en la sección 3.1 de este documento.

5.9.6 Clasificación

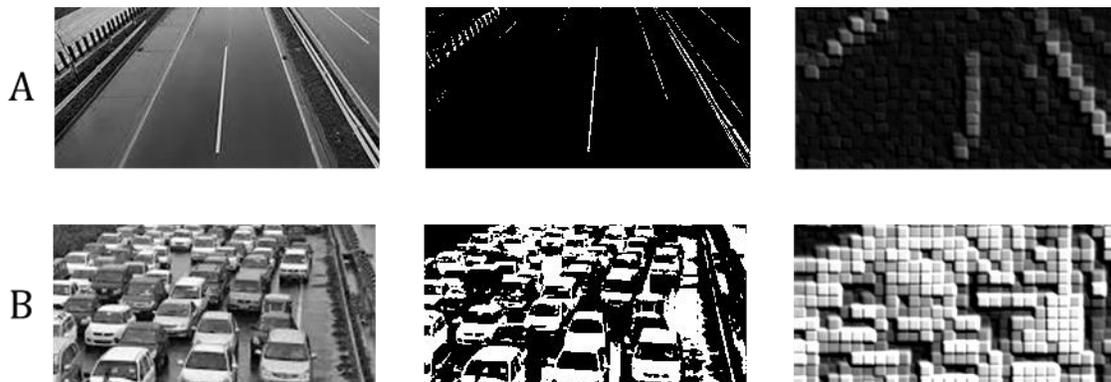
El proceso de clasificación requiere como parámetros dos archivos, el primero es la imagen sobre la cual se realiza la clasificación y detección de objetos, el segundo archivo es un conjunto de datos en formato XML (obtenido del proceso de entrenamiento con Open CV). Como resultado la clasificación arroja un conjunto de coordenadas, dimensiones ancho y alto de los bordes rectangulares que representan a los objetos sobre la imagen que han sido reconocidos por el clasificador (sección 3.2.4). La siguiente figura ilustra el diseño para el resultado obtenido de una clasificación de cascada donde se muestran de color verde los bordes rectangulares de los vehículos identificados.



*Figura 5.10 Diseño de imagen resultado de una clasificación.
(Elaboración Propia)*

5.9.7 Análisis de texturas

El análisis de texturas consiste en: dada una imagen como parámetro de entrada, se calcula la matriz GLMC de cada imagen (sección 3.2.1) y posteriormente se obtienen los valores de energía y entropía. Una imagen con muchos vehículos forma una textura más áspera, más heterogénea, y por lo tanto se obtiene un valor de entropía más alto. Caso contrario a las imágenes donde el tráfico es menor porque se genera una textura más lisa donde predomina un solo patrón de colores generando un valor de energía alto y un valor de entropía bajo. La siguiente figura con dos imágenes (A y B) muestra un ejemplo visual de las diferencias entre las texturas generadas por imágenes de avenidas con y sin tráfico.



*Figura 5.11 Ejemplo de texturas generadas a partir del tráfico.
(Elaboración propia)*

5.9.8 Comparación de resultados

En esta etapa el sistema compara los resultados del análisis y clasificación (número de vehículos encontrados y propiedades de textura) de cada una de las imágenes y con base en los resultados obtenidos el sistema elige los intervalos de tiempo para el ciclo de los semáforos priorizando la duración de luz verde a aquellas avenidas que presentan mayor afluencia vehicular.

5.9.9 Ajuste de ciclos

Actualmente, para la instalación de semáforos las autoridades correspondientes realizan estudios mediante técnicas de ingeniería de tránsito o ingeniería de transporte para obtener el intervalo mínimo y el intervalo máximo que puede tener cada cambio de luces en los semáforos de una intersección y con base en ello se establece la duración del ciclo. En los estudios intervienen muchos factores como ancho de la avenida, número de carriles, velocidad promedio e incluso mediciones de la actividad peatonal. La propuesta para el diseño de este sistema solamente sustituye el ciclo fijo por ciclos dinámicos, pero respeta los intervalos mínimos y máximos establecidos que hayan surgido de los estudios y de las normas que rigen la implementación actual. Por lo anterior, el sistema mantiene un diseño configurable que se adapta a las necesidades específicas de la intersección que se desea controlar. A continuación, se describe el diseño para el procedimiento de ajuste de ciclo con base en intervalos de tiempo previamente definidos.

5.9.9.1 Diseño para control de respuesta

El cruce más simple se presenta en aquellas intersecciones donde únicamente se cruzan dos caminos y cada camino tiene únicamente un solo sentido. La siguiente figura muestra las posibles combinaciones que ocurren en un cruce simple considerando solamente dos valores: baja afluencia y alta afluencia.

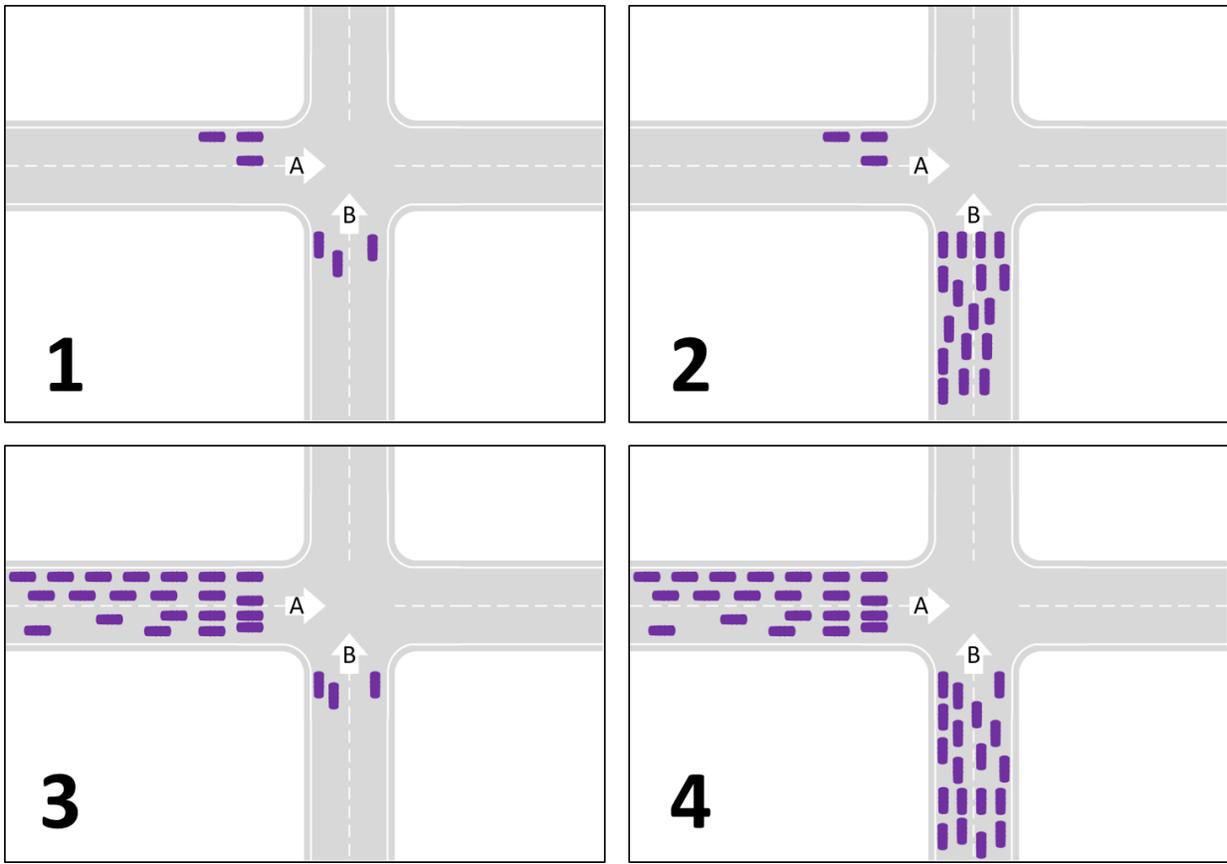


Figura 5.12 Combinaciones en un cruce simple de dos sentidos.
(Elaboración propia)

En la imagen se aprecia que con dos valores de tráfico (alta afluencia y baja afluencia) en un cruce simple de dos sentidos (A y B) se obtienen cuatro posibles combinaciones:

Combinación	Afluencia en A	Afluencia en B
1	BAJA	BAJA
2	BAJA	ALTA
3	ALTA	BAJA
4	ALTA	ALTA

Tabla 5.1 Posibles combinaciones en un cruce simple de dos sentidos.
(Elaboración propia)

Como se mencionó anteriormente, el diseño del sistema respeta los intervalos de tiempo derivados de los estudios o de las mediciones realizadas previamente por las autoridades en la intersección de avenida. Para describir el diseño del sistema, a continuación, se muestra una tabla con datos (solamente de ejemplo) con intervalos de tiempo mínimo y tiempo máximo aplicados para un cruce simple.

Semáforos	intervalo mínimo de luz verde	intervalo máximo de luz verde	intervalo para luz ámbar
Semáforo en A	40 s	60 s	5 s
Semáforo en B	35 s	55 s	5 s

*Tabla 5.2 Intervalos de tiempo mínimo y tiempo máximo.
(Elaboración propia)*

En la tabla anterior solo se muestran los intervalos de tiempo de función de la duración de la luz verde y se omiten los intervalos de tiempo para luz roja, esto con el fin de simplificar la información debido a que en un semáforo de cruce simple el intervalo de luz verde más el intervalo de luz ámbar es igual al intervalo de luz roja del semáforo opuesto. Considerando para este caso los intervalos de la tabla anterior y las 4 combinaciones que existen en un cruce simple, la respuesta del sistema está conformada por 4 posibles eventos:

Respuesta	Verde en A, rojo en B	Amarillo en A	Rojo en A, verde en B	Amarillo en B	Duración del ciclo
Evento 1	40 s 45 s	5 s	40 s 35 s	5 s	85 s
Evento 2	40 s 45 s	5 s	60 s 55 s	5 s	105 s
Evento 3	60 s 65 s	5 s	40 s 35 s	5 s	105 s
Evento 4	60 s 65 s	5 s	60 s 55 s	5 s	125 s

*Tabla 5.3 Eventos para el control de respuesta del sistema.
(Elaboración propia)*

Con los cuatro eventos que muestra la tabla anterior, el sistema ofrece una respuesta con base en la situación del cruce, al finalizar cada ciclo el sistema realiza una nueva evaluación de la afluencia vehicular y automáticamente toma la decisión de continuar con el mismo ciclo o cambiar a otro ciclo de la tabla de eventos. La siguiente ilustración muestra las diferentes respuestas que el sistema ofrecería ante cada una de las cuatro combinaciones del cruce simple descritas anteriormente en la Figura 5.12.

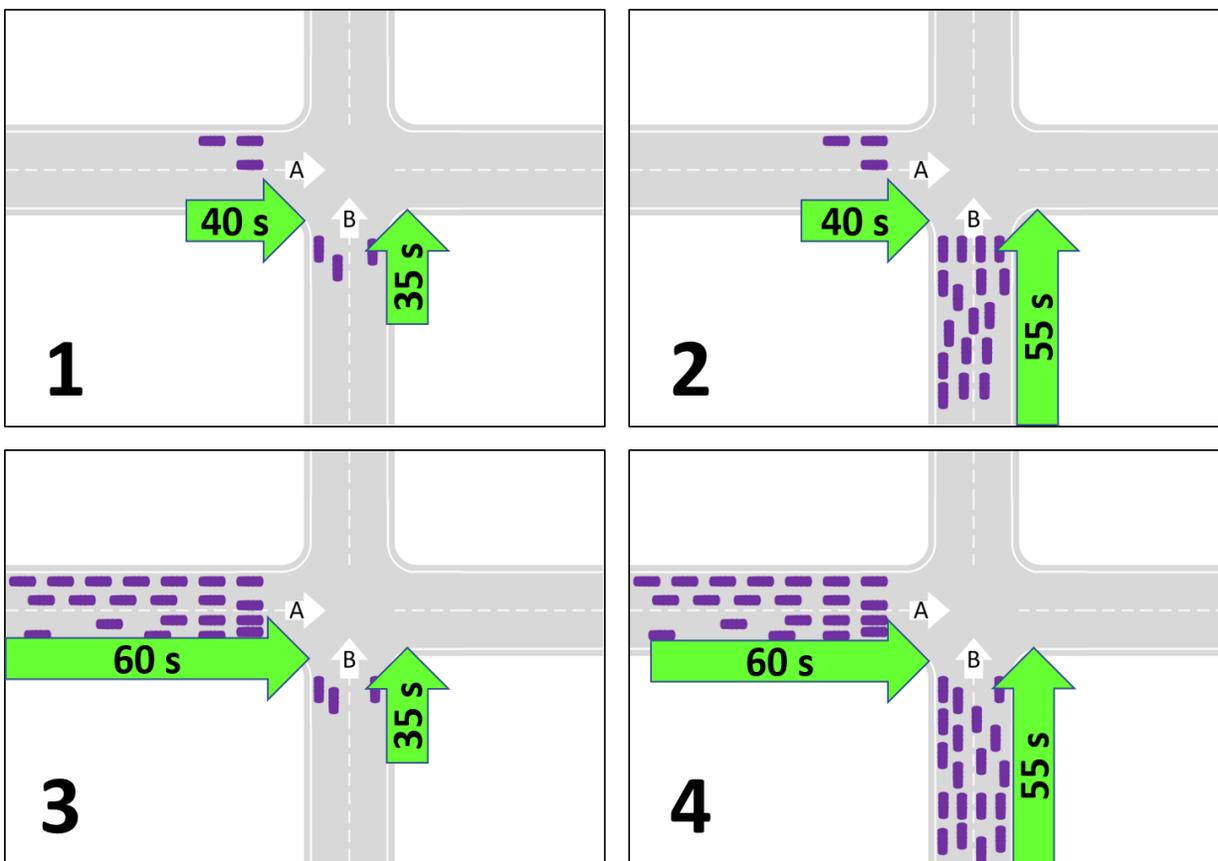


Figura 5.13 Combinaciones de respuesta para un cruce simple.
(Elaboración propia)

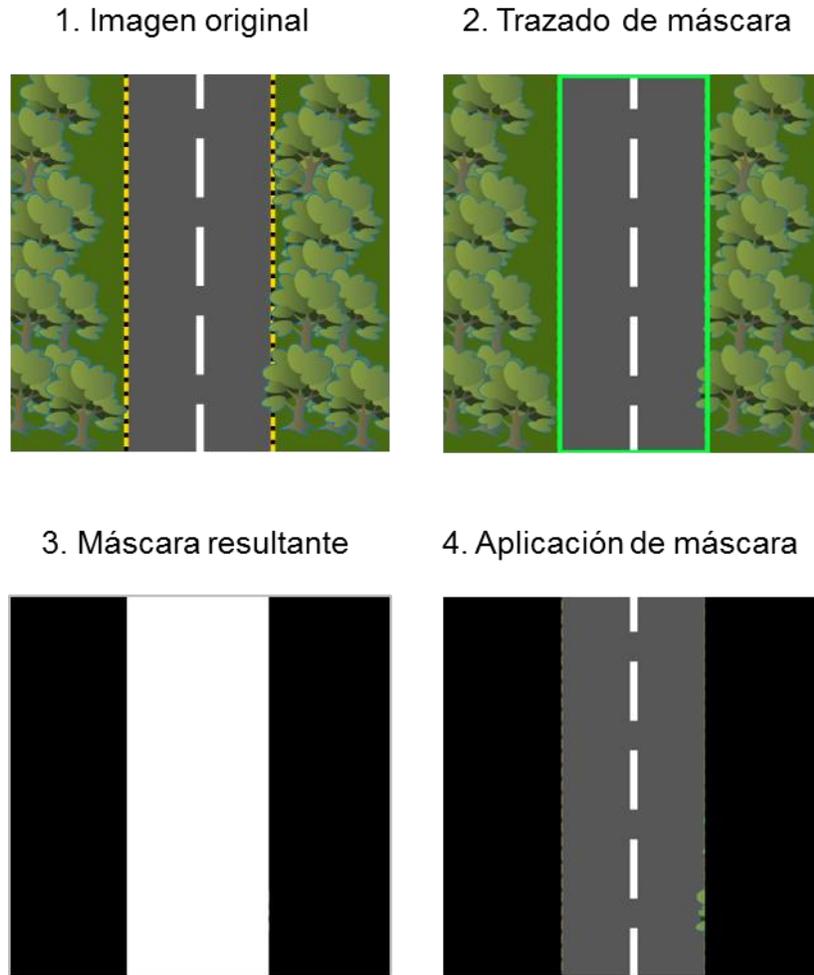
Existen otros tipos de cruce, entre más complejo es el cruce más aumenta el número de posibilidades, como consecuencia también aumenta el número de eventos para la respuesta del sistema.

La respuesta puede simplificarse e incluso se puede llegar a tener el mismo número de eventos que hay en un cruce simple. La simplificación se basa en que algunas de las combinaciones son equivalentes. Por ejemplo, como se muestra en la siguiente tabla, cuando el cruce es de dos caminos, pero con cuatro sentidos (A, B, C y D), el número de combinaciones posibles es de 16.

Combinación	Afluencia en A	Afluencia en B	Afluencia en C	Afluencia en D
1	BAJA	BAJA	BAJA	BAJA
2	BAJA	BAJA	BAJA	ALTA
3	BAJA	BAJA	ALTA	BAJA
4	BAJA	BAJA	ALTA	ALTA
5	BAJA	ALTA	BAJA	BAJA
6	BAJA	ALTA	BAJA	ALTA
7	BAJA	ALTA	ALTA	BAJA
8	BAJA	ALTA	ALTA	ALTA
9	ALTA	BAJA	BAJA	BAJA
10	ALTA	BAJA	BAJA	ALTA
11	ALTA	BAJA	ALTA	BAJA
12	ALTA	BAJA	ALTA	ALTA
13	ALTA	ALTA	BAJA	BAJA
14	ALTA	ALTA	BAJA	ALTA
15	ALTA	ALTA	ALTA	BAJA
16	ALTA	ALTA	ALTA	ALTA

*Tabla 5.4 Posibles combinaciones en un cruce de cuatro sentidos.
(Elaboración propia)*

Las combinaciones equivalentes reducen los eventos y como se muestra en la siguiente figura, para el caso en el que los conductores tengan prohibido dar vuelta a la izquierda (independientemente de la afluencia vehicular que presenten B y D) las combinaciones entre A y C son equivalentes ya que, al tratarse de sentidos diferentes, pero de una misma arteria, el intervalo de luz verde en A es exactamente el mismo intervalo de luz verde para C, por lo tanto, si se consideran todas las equivalencias, la tabla anterior se puede simplificar hasta obtener una tabla de 4 combinaciones nuevamente.

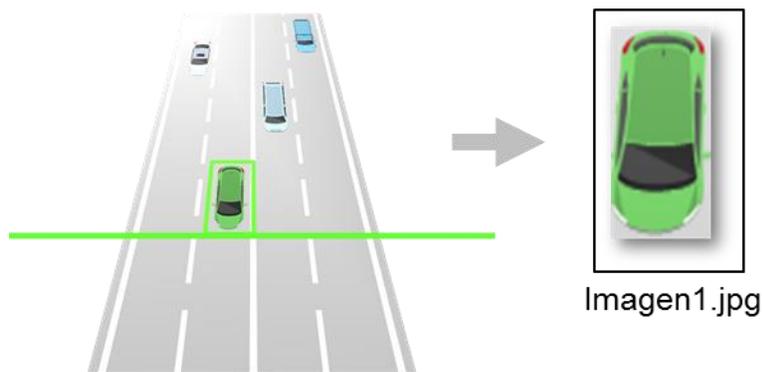


*Figura 5.15 diseño de fases para la aplicación de máscaras de segmentación.
(Elaboración propia)*

5.9.11 Recolección de imágenes para entrenamiento

El diseño del sistema permite trazar manualmente una línea recta como límite de intersección. El objetivo es que cuando el borde rectangular de un objeto en movimiento colisione con la línea recta, automáticamente se guarde una imagen del objeto en un directorio que posteriormente servirá para la fase de entrenamiento (sección 3.2.4.1) del clasificador de cascada.

El primer paso es trazar manualmente un segmento de línea recta mediante una interfaz gráfica, el sistema detecta continuamente el movimiento de objetos mediante técnicas de visión artificial para la substracción de fondo (sección 3.2.3), posteriormente se obtiene el borde rectangular de cada uno de los vehículos en movimiento para que finalmente, el sistema determine si el borde del objeto colisiona con el segmento de línea entonces la imagen del objeto delimitada por el borde rectangular es almacenada en un directorio de recolección de imágenes positivas (sección 3.2.4.1)



*Figura 5.16 Diseño de proceso para recolección de imágenes positivas.
(Elaboración propia)*

El video de este primer experimento tiene una duración de 19 minutos, fue proporcionado por el municipio de Nezahualcóyotl en un archivo con formato DVT, la captura de imágenes se realizó el día miércoles 21 de febrero de 2018 a las 18:15 horas mediante una cámara giratoria de video-vigilancia que opera actualmente. La siguiente figura ilustra los ángulos de cámara (1-A, 1-B, 1-C y 1-D) contenidos en el video.



*Figura 6.2 Ángulos de cámara: primera configuración.
(C-4 Centro de mando, 2018)*

El segundo experimento se realizó mediante capturas de imágenes propias que fueron retransmitidas vía streaming desde Av. Chimalhuacán y Av. Adolfo López Mateos.



Figura 6.3 Ubicaciones 2, 3, 4 y 5 de la transmisión de video para pruebas.
(Google Maps. INEGI, 2018)

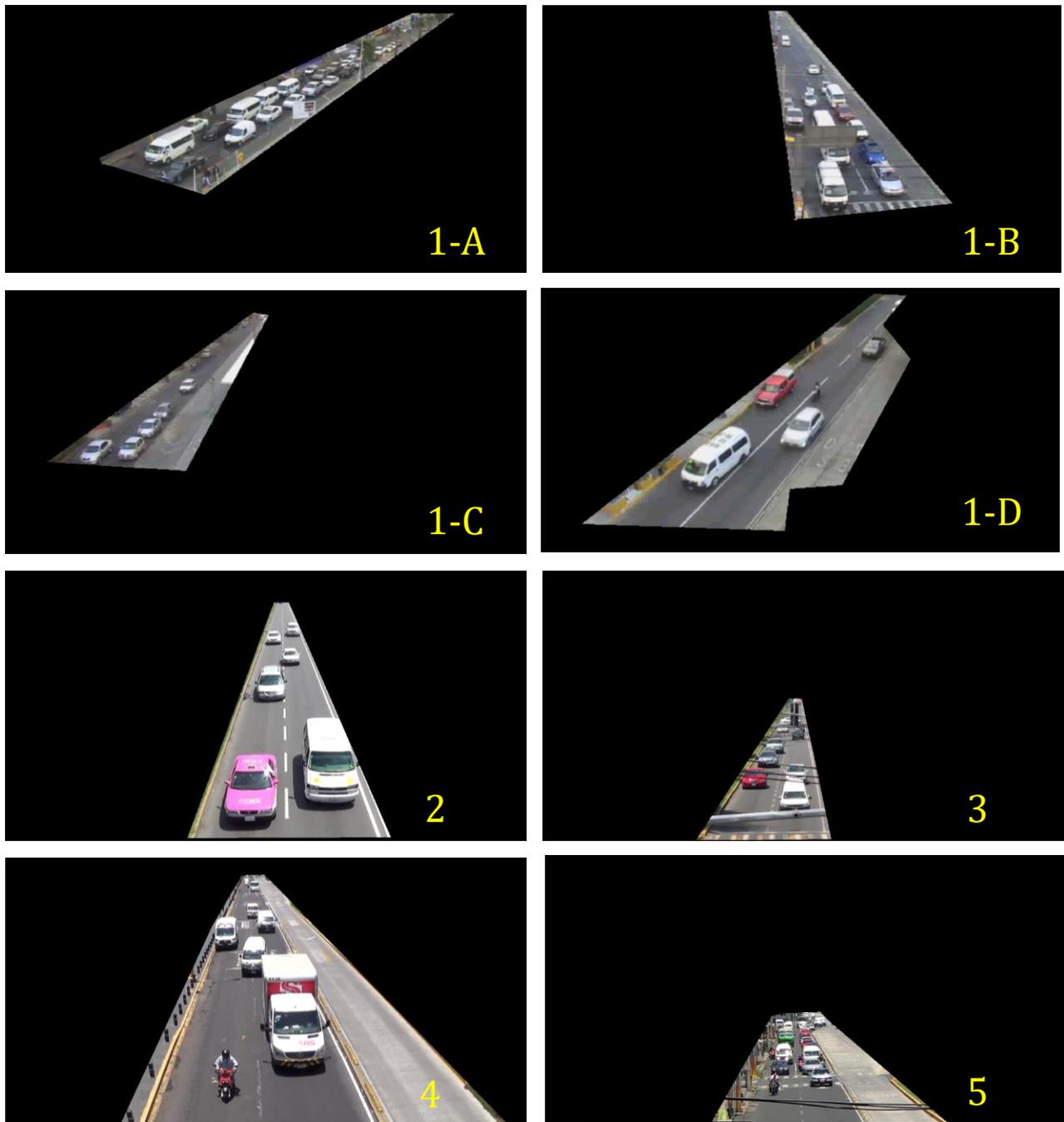
Los videos del segundo experimento tienen una duración de 30 minutos, las capturas se realizaron de manera simultánea el día sábado 28 de julio de 2018 a las 14:45 horas mediante cámaras de dispositivos móviles Android con resolución 4K en formatos .MOV y .MTS. La siguiente figura muestra los ángulos de cámara (2, 3, 4 y 5) obtenidos en la segunda configuración.



*Figura 6.4 Ángulos de cámara: segunda configuración.
(Elaboración propia)*

6.1.1 Aplicación de máscaras de segmentación

En ambos experimentos se trazaron máscaras de segmentación de forma manual a cada uno de los ángulos de cámara para ignorar las zonas no deseadas (sección 5.9.10).



*Figura 6.5 Conjunto de máscaras de segmentación.
(Elaboración propia)*

6.1.2 Detección de movimiento

Se aplicó el algoritmo de sustracción de fondo KNN (sección 3.2.3.4) para la detección de movimiento con los siguientes parámetros: (history=500, dist2Threshold=400.0, detectShadows=False).

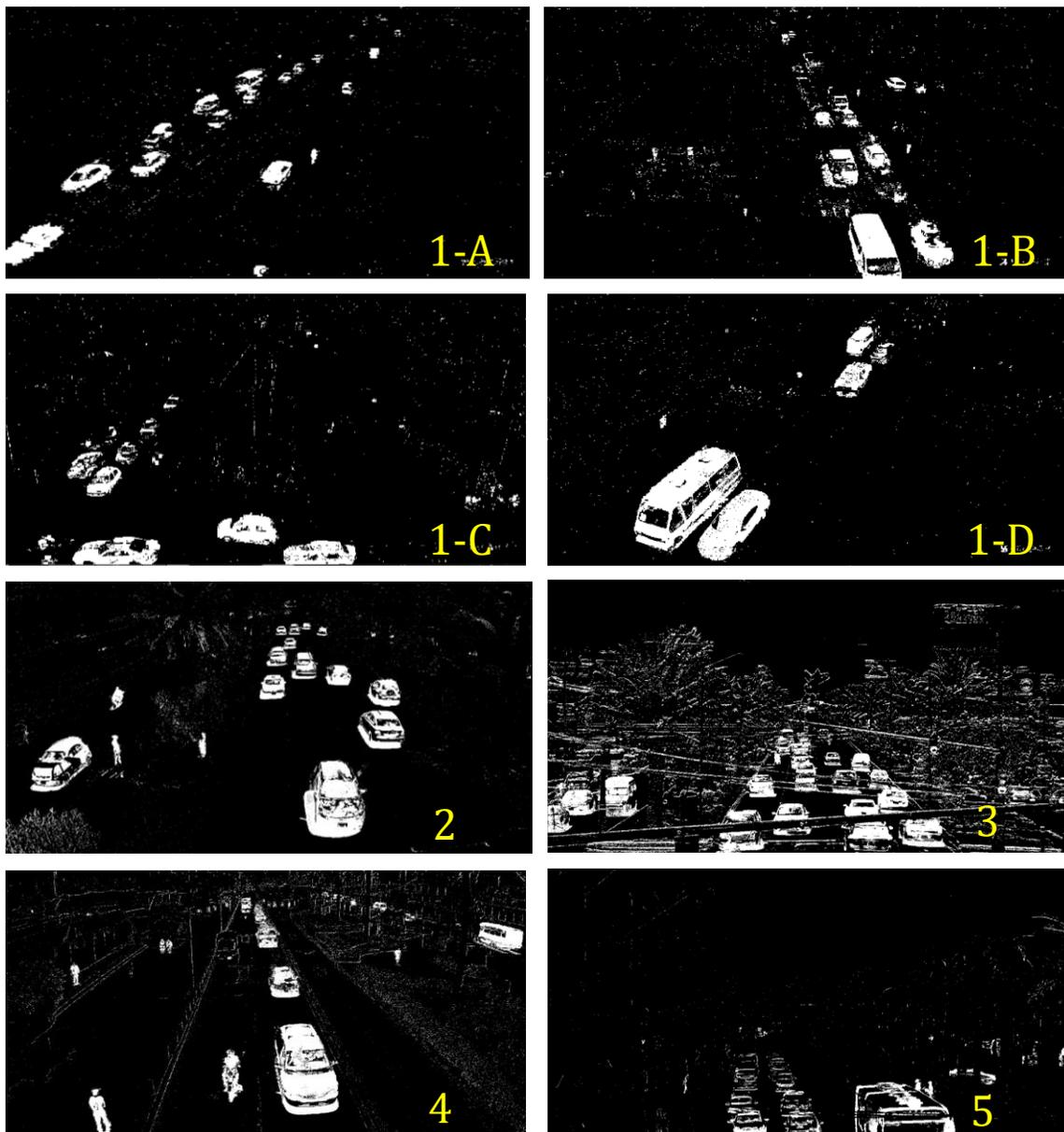


Figura 6.6 Detección de movimiento.
(Elaboración propia)

6.1.3 Recolección de imágenes para entrenamiento

Se trazaron manualmente segmentos de línea recta como límite de intersección, se aplicó el proceso de recolección de imágenes positivas (sección 5.9.11) durante 5 minutos en cada ángulo de cámara.



Figura 6.7 Recolección de imágenes para entrenamiento de Haar Cascade.
(Elaboración propia)

6.1.4 Clasificación

A partir de la recolección de imágenes se generaron archivos con formato .XML mediante la herramienta de entrenamiento de Open CV (sección 5.9.6). Una vez concluida la etapa de entrenamiento se aplicó el algoritmo de clasificación Haar Cascade.

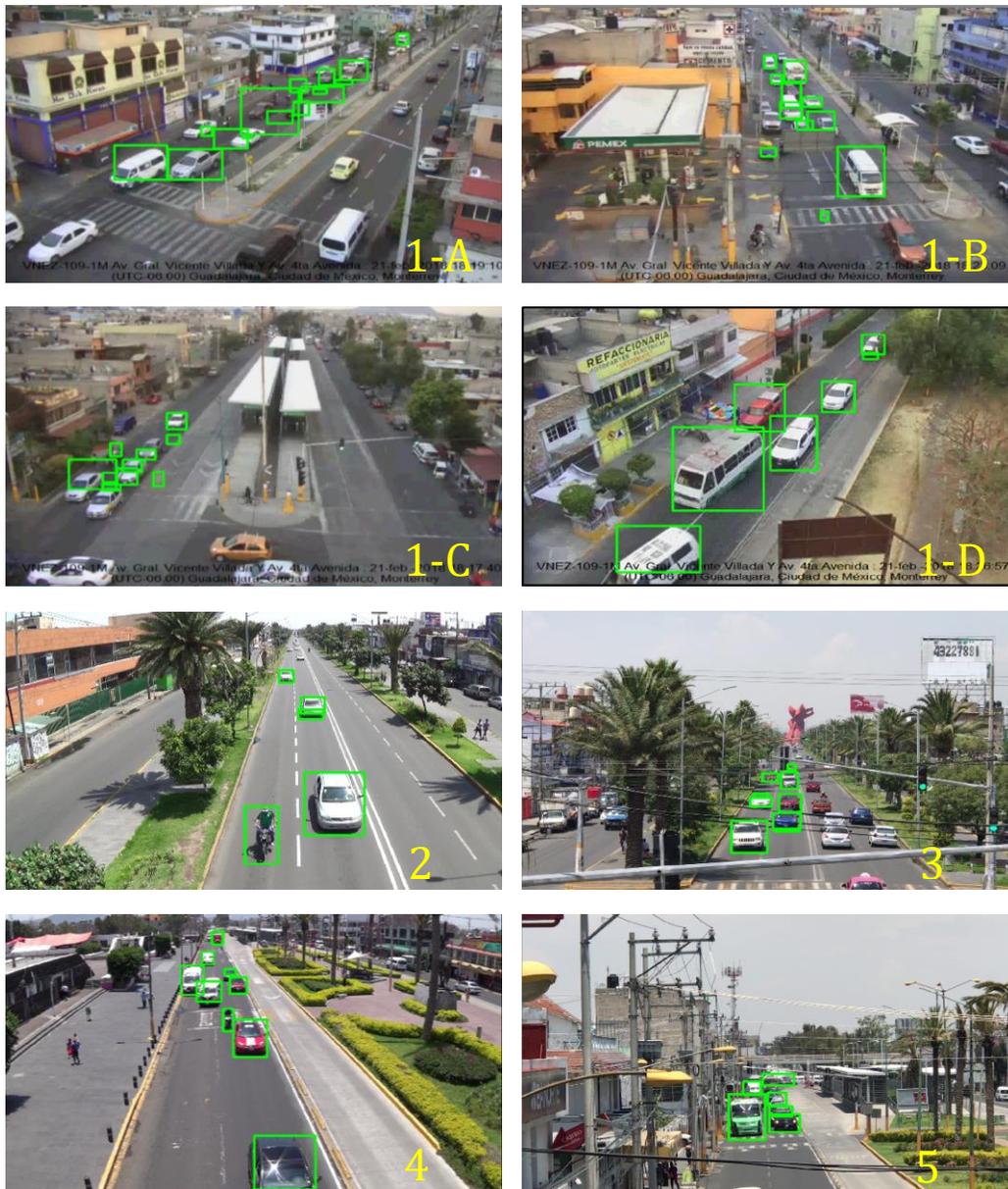


Figura 6.8 Aplicación de clasificador Haar Cascade Open CV.
(Elaboración propia)

6.1.5 Análisis de texturas

Se eligieron intencionalmente 48 imágenes (3 imágenes por cada ángulo de cámara) de las cuales 24 mostraban una cantidad no mayor a 3 vehículos y las otras 24 mostraban niveles de tráfico mayores a 10 vehículos para evaluar el proceso de análisis de textura (sección 5.9.7). Para el cálculo de la matriz de co-ocurrencia de niveles de gris se utilizaron los siguientes parámetros con la librería `scikit-image` (`scikit-image.org`, 2018):

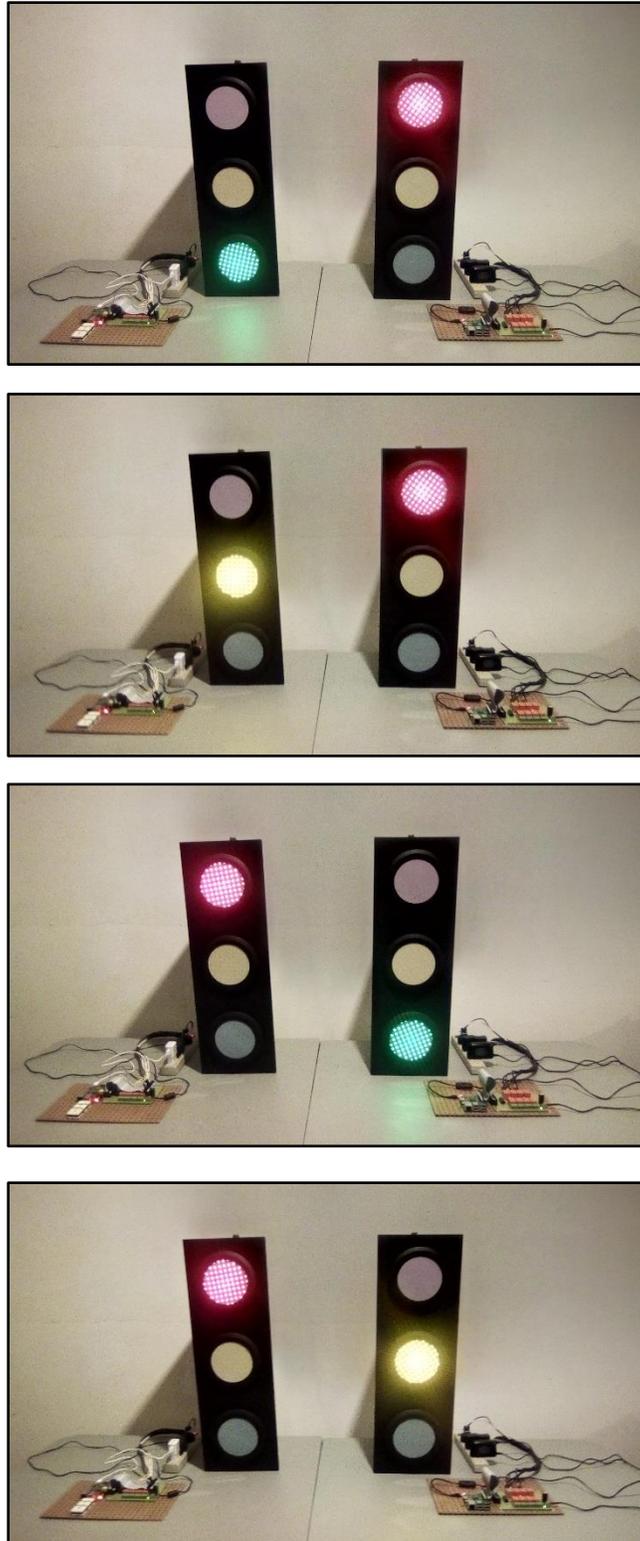
- `glcm = greycomatrix(image, [1,5,10,50,100], [0, np.pi/4, np.pi/2, 3*np.pi/4], symmetric=True, normed=True)`

6.1.6 Ajuste de ciclos

Se construyó una maqueta de pruebas con dos semáforos que simulan la comunicación y la sincronización entre el sistema propuesto y los semáforos que operan actualmente. Se utilizaron dos micro ordenadores `raspberry-pi` configurados en red (sección 5.6) para el procedimiento de ajuste de ciclos (sección 5.9.9) con los siguientes conectores GPIO para el control de luces de cada semáforo:

Color de luz	GPIO	Valor
Verde	23	HIGH / LOW
Amarillo	24	HIGH / LOW
Rojo	25	HIGH / LOW

*Tabla 6.1 Configuración de conectores GPIO.
(Elaboración propia)*



*Figura 6.9 Maqueta para simulación de ciclos.
(Elaboración propia)*

6.2 Discusión de resultados

La cámara implementada actualmente (Figura 6.1) es una cámara de rotación que no entrega al sistema imágenes de dos o más ángulos simultáneamente, la baja resolución dificulta y vuelve inestables los procesos de entrenamiento y clasificación del sistema, además de que tiene que ser manipulada por un operador. Por lo contrario, los resultados que derivaron de las imágenes del segundo experimento mostraron estabilidad en el proceso de clasificación para el conteo de automóviles.

Las máscaras de segmentación (sección 6.1.1) optimizaron considerablemente el funcionamiento del sistema, desde el proceso de detección de movimiento y entrenamiento del clasificador, ya que al ignorar las zonas no deseadas como banquetas y camellones se reduce en más del 50% el volumen de información innecesaria, al mismo tiempo aumenta la velocidad de respuesta del sistema.

En el proceso de detección de movimiento (sección 6.1.2) los parámetros utilizados en el método para sustracción de fondo arrojaron resultados precisos incluso ante factores que no estaban contemplados como ligeras vibraciones en la cámara por viento. Durante los 5 minutos que se ejecutó el proceso de recolección para imágenes de entrenamiento (sección 6.1.3) se obtuvieron entre 50 y 100 imágenes positivas y más de 1000 imágenes negativas por cada ángulo de cámara.

El proceso de clasificación (sección 6.1.4) arrojó un promedio de error de ± 3 vehículos en el conteo de autos en distancias menores a 100 metros, aumentando hasta obtener un reconocimiento nulo en distancias a partir de los 200 metros. Mientras tanto en los procesos de análisis de texturas (sección 6.1.5) la diferencia de resultados fue mayor entre el conjunto de imágenes con tráfico y el conjunto de imágenes sin tráfico en distancias mayores a los 150 metros.

La respuesta en el proceso de ajuste de ciclos (sección 6.1.6) por medio de microcontroladores es inmediata por lo tanto se obtuvo como resultado una sincronización perfecta.

Capítulo 7

Conclusiones y futuras líneas de desarrollo

En este capítulo se presentan las aportaciones y las limitaciones del sistema, además, se describen las futuras líneas de desarrollo para la escalabilidad del mismo.

7.1 Conclusiones

Mediante del diseño y la programación se logró desarrollar un sistema de semáforos inteligentes con procesamiento y algoritmos aplicados a secuencias de imágenes que controla el funcionamiento de los semáforos simulando la inteligencia humana. Este desarrollo brinda la oportunidad de conjugar la tecnología con servicios para la sociedad.

A través de la Secretaría de Comunicaciones y Transportes, así como de la Delegación Regional Secretaría de Movilidad Zona IV, se logró obtener el conocimiento sobre el funcionamiento de los semáforos que actualmente se encuentran operando en el municipio, al mismo tiempo se logró identificar una de las intersecciones con mayor congestionamiento vehicular en el municipio, por lo tanto, es posible generar propuestas de tecnología para la resolución de problemas sin discrepar con la reglamentación actual.

Así como lo han hecho otras ciudades de mundo, es altamente viable que el tráfico de la ciudad Nezahualcóyotl sea controlado por sistemas inteligentes sobre todo en intersecciones que presentan mayor afluencia vehicular como las identificadas en este trabajo, además es posible mediante tecnologías de uso libre con la ventaja de que reducen considerablemente los costos de desarrollo.

Debido a la baja resolución y nitidez de las imágenes generadas por la cámara que opera actualmente en la intersección de 4ª. Avenida y Av. Vicente Villada. En ciudad Nezahualcóyotl no es posible una completa reutilización de recursos, sin embargo, se pueden utilizar los semáforos existentes de tecnología de LED y los módulos de control para la implantación del sistema propuesto.

El desarrollo de sistemas mediante metodologías recientes como SCRUM permite una planeación adecuada, escalable y ágil del sistema a diferencia de metodologías predictivas con modelamiento UML que no garantizan la escalabilidad de los sistemas e implican mayor tiempo y recursos en las etapas de documentación.

El sistema propuesto no soluciona la congestión de tráfico vehicular en la ciudad, solo contribuye a disminuirlo especialmente en el caso de intersecciones que presentan una diferencia contrastante en los niveles de afluencia vehicular de cada una de las arterias que conforman la intersección ya que el problema de congestión se deriva también de otros factores como: accidentes automovilísticos, capacidad de volumen de las avenidas limitada, falta de educación y cultura vial.

A pesar de que el sistema propuesto ha sido evaluado mediante pruebas de percepción humana y criterios propios, solamente se podrá conocer una magnitud exacta del nivel de disminución de tráfico cuando el sistema opere de manera permanente el control de semáforos en alguna intersección y le sean aplicados los estudios de ingeniería de tránsito pertinentes.

7.2 Futuras líneas de desarrollo

A partir del sistema propuesto se derivan otras líneas de escalabilidad y de desarrollo como son:

7.2.1 Implantación del sistema propuesto

Consiste en instalar el sistema en una de las intersecciones de la ciudad Nezahualcóyotl para controlar permanentemente el cambio de luces en los semáforos. Los beneficios que se obtienen con la implantación son:

- Se obtendría una base de entrenamiento sólida y robusta ya que el sistema tendría un periodo muy largo (días o semanas) para la recolección de información (un entrenamiento largo significa una clasificación con mayor precisión) al mismo tiempo, el banco de información generada es reutilizable para el desarrollo de otros proyectos o investigaciones.
- Se generarían mediciones exactas en cuanto al rendimiento del sistema ya que la implantación permitiría realizar estudios de ingeniería de tránsito que justifiquen la instalación en más intersecciones de la ciudad.

7.2.2 Escalabilidad del sistema propuesto

Consiste en integrar otros módulos e incluso otros sistemas al sistema propuesto, algunos ejemplos son:

- Desarrollo de aplicación móvil para monitoreo y control remoto del sistema.
- Diseño y arquitectura para red de intersecciones.
- Implementación de modelos estadísticos para la predicción del tráfico y toma de decisiones complejas.
- Integración de módulos de seguridad para vigilancia mediante streaming y reconocimiento de números de matrícula.
- Implementación de ecotecnias (tecnologías ecológicas sostenibles) mediante paneles solares u otras alternativas para reducir el consumo de energía.

Capítulo 8

Referencias

- Alaimo, D. M. (2013). *Proyectos ágiles con Scrum : flexibilidad, aprendizaje, innovación y colaboración en contextos complejos* (primera ed.). Kleer.
- Algarabía. (junio de 2013). *Semáforo Night*. Recuperado el 2 de junio de 2018, de Algarabía: <https://algarabia.com/wp-content/uploads/2013/06/s9-semaforo-Knight.jpg>
- Azhar, H., & Widjanarko, T. (2002). *Comparison of Two Binary Image Thresholding Methods*. EECE Department, University of Memphis. Obtenido de http://www.academia.edu/1197638/Comparison_of_Two_Binary_Image_Thresholding_Methods_2002_
- Bautista Rojas, C. (29 de febrero de 2016). *El origen del semáforo*. Recuperado el 2 de junio de 2018, de Algarabía: <https://algarabia.com/desde-la-redaccion/el-origen-del-semaforo>
- C-4 Centro de mando. (2018). Centro de mando de la dirección de seguridad pública municipal. Nezahualcóyotl.
- Canny, J. (1986). A Computational Approach to Edge Detection. (IEEE, Ed.) *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-8(6)*, 679 - 698.
doi:10.1109/TPAMI.1986.4767851
- De Los Santos, A. (2011). *La teoría del color*. Grupo IDAT diseño gráfico.
- De Los Santos, P. J. (4 de agosto de 2016). *matlab typ*. Recuperado el 2 de junio de 2018, de matlab-typ.blogspot: <http://matlab-typ.blogspot.com/2016/08/convertir-una-imagen-escala-de-grises.html>
- Gonzalez, R., & Woods, R. (2008). *Digital Image Processing* (tercera ed.). Pearson Education, Inc.
- Google Maps. (abril de 2017). *Av Bordo de Xochiaca*. Recuperado el 2 de junio de 2018, de Google Maps: <https://goo.gl/maps/NLDEwBHpD142>

- Google Maps. INEGI. (2018). *Nezahualcóyotl, Méx.* Recuperado el 2 de junio de 2018, de Google Maps: <https://www.google.com.mx/maps/place/Nezahualcóyotl>
- Haralick, R., Shanmugan, K., & Dinstein, I. (1973). Textural features for image classification. *IEEE Trans. Syst., Man, Cybern.*, vol SMC-3.
- Hong Kong Transport Department. (2018). *Traffic Control and Surveillance Systems.* Recuperado el 2 de junio de 2018, de Transport Department ITS Achievements: https://www.td.gov.hk/en/transport_in_hong_kong/its/its_achievements/traffic_control_and_surveillance_systems/index.html
- INEGI. (2016). *Vehículos de motor registrados en circulación.* Recuperado el 2 de junio de 2018, de Instituto Nacional de Estadística y Geografía (INEGI): http://www.inegi.org.mx/est/lista_cubos/consulta.aspx?p=adm&c=8
- Jiménez Guerrero, M. D. (2015). *Extracción de características de textura basada en la transformada Wavelet discreta.* Recuperado el 2 de junio de 2018, de <http://bibing.us.es/proyectos/abreproy/11494/direccion/PROYECTO%252F>
- LEHMANN, C. (1989). *Geometría Analítica* (Decimatercera ed.). LIMUSA.
- López-Portilla Vigil, B. M., Menéndez Alonso, R. J., & Iglesias Martínez, M. E. (2016). Implementación del Algoritmo de Otsu sobre FPGA. *Revista Cubana de Ciencias Informáticas.* Recuperado el 2 de junio de 2018, de http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2227-18992016000300002
- MeteoPoint. (2016). *ARMM Traffic Counter.* Recuperado el 2 de junio de 2018, de MeteoPoint: <http://www.meteopoint.com/attach/1265721400.pdf>
- Miguel Lozano, D. (2016). *go-bees.readthedocs.io/es/develop.* Obtenido de http://go-bees.readthedocs.io/es/develop/documentacion/3_ConceptosTeoricos.html#art-zivkovic-efficient-2006
- OpenCV.org. (2018). *Open Source Computer Vision.* Recuperado el 2 de junio de 2018, de docs.opencv.org: https://docs.opencv.org/3.4.2/d6/d00/tutorial_py_root.html
- OTSU, N. (1979). A threshold selection method from gray-level histogram. *IEEE Transactions on Systems, Man and Cybernetics.*

- Raspberry Pi Foundation. (2018). *Documentation*. Obtenido de www.raspberrypi.org:
<https://www.raspberrypi.org/documentation/hardware/>
- SCATS. (2018). *SCATS The benchmark in urban traffic control*. Recuperado el 1 de junio de 2018, de NSW Government: <http://www.scats.com.au/>
- scikit-image.org. (2018). *Scikit-Image Image processing in python*. Obtenido de scikit-image.org: <https://scikit-image.org/>
- SCT vial. (2012). *Sistema de Control de tránsito*. Recuperado el 2 de junio de 2018, de SCT vial: <http://www.sctvial.com/>
- Secretaría de Comunicaciones y Transportes. (mayo de 2014). Semáforos. En *Manual de señalización vial y dispositivos de seguridad* (6 ed., págs. 24-51). México. Recuperado el 2 de junio de 2018
- Siemens. (2016). *Intelligent Traffic Management System*. Recuperado el 2 de junio de 2018, de Siemens ingenuity for life: <https://www.siemens.com/innovation/en/home/pictures-of-the-future/mobility-and-motors/urban-mobility-worlds-largest-cloud-based-traffic-control-center.html>
- Sucar, L. E., & Gómez, G. (2011). *Visión Computacional*. Obtenido de Instituto Nacional de Astrofísica, Óptica y Electrónica: <https://ccc.inaoep.mx/~esucar/Libros/vision-sucar-gomez.pdf>
- Sumitomo Electric. (2018). *ITS: Intelligent Transport Systems*. Recuperado el 2 de junio de 2018, de Sumitomo Electric Industries: <http://global-sei.com/its/products/itcs.html>
- Surtrac. (2018). *Intelligent Traffic Signal Control*. Recuperado el 2 de junio de 2018, de Surtrac: <https://www.surtrac.net/>
- Takeuchi, H., & Nonaka, I. (1986). *The New New Product Development Game*. Recuperado el 2 de junio de 2018, de <https://hbr.org/1986/01/the-new-new-product-development-game>
- Trafictec. (2018). *Semáforo LED para uso vial con lámpara de 30 cm*. Recuperado el 2 de junio de 2018, de Trafictec S.A. de C.V.: http://www.trafictec.com/semaf_led_30.php

- Trigas Gallego, M. (2012). *Metodología Scrum*. Obtenido de <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/17885/1/mtrigasTFC0612memoria.pdf>
- Valverde Rebaza, J. (2007). *Detección de bordes mediante el algoritmo de Canny*. Obtenido de http://www.academia.edu/15368015/Detecci%C3%B3n_de_bordes_mediante_el_algoritmo_de_Canny
- Vélez Serrano, J. F., Moreno Díaz, A. B., Sánchez Calle, Á., & Esteban Sánchez-Marín, J. L. (2003). *Visión por computador* (2da ed.). (Dykinson, Ed.) Madrid, España.
- Viola, P., & Jones, M. (2001). *Rapid Object Detection using a Boosted Cascade of Simple Features*. Obtenido de Accepted conference on computer vision and pattern recognition 2001: <https://www.cs.cmu.edu/~efros/courses/LBMV07/Papers/viola-cvpr-01.pdf>

Apéndice A:

Glosario de Términos

- **Algoritmo:** Es un conjunto prescrito de instrucciones o reglas bien definidas, ordenadas y finitas que permiten llevar a cabo una actividad mediante pasos sucesivos que no generen dudas a quien deba hacer dicha actividad.
- **Aplicación:** Es un programa informático diseñado como herramienta para permitir a un usuario realizar uno o diversos tipos de tareas.
- **Afluencia Vehicular:** Abundancia, gran cantidad de vehículos.
- **Avenida Principal:** Vía importante de comunicación dentro de una ciudad o asentamiento urbano.
- **Binarización:** Una reducción de información en la que los únicos valores posibles son verdadero y falso (1 y 0). En caso de una imagen digital los valores verdadero y falso se corresponden a dos colores: blanco y negro.
- **Bit:** Binary digit (o sea de 'dígito binario', en español denominado como bit, y en menor medida como bitio).¹ Un bit es un dígito del sistema de numeración binario.
- **Cálculo:** Consiste en un procedimiento mecánico, o algoritmo, mediante el cual podemos conocer las consecuencias que se derivan de unos datos previamente conocidos debidamente formalizados y simbolizados.
- **Cámara Digital:** Es una cámara fotográfica que, en vez de captar y almacenar fotografías en película química como las cámaras de película fotográfica, recurre a la fotografía digital para generar y almacenar imágenes.
- **Camellón:** Lomo continuo de materiales destinados a la construcción o conservación de una obra vial, colocado a lo largo de un tramo, mientras dura la obra. En una calzada o camino, faja separadora limitada por rayas de pintura o guarniciones, que se construye para separar el tránsito de vehículos en sentidos opuestos o en el mismo sentido.
- **Ciclo:** Lapso necesario para una secuencia completa de indicaciones de un semáforo, hasta que vuelve al color o indicación inicial.

- **Congestionamiento de tráfico vehicular:** Se refiere, tanto urbana como interurbanamente, a la condición de un flujo vehicular que se ve saturado debido al exceso de demanda de las vías, produciendo incrementos en los tiempos de viaje.
- **Control Maestro:** Control de estructura jerarquizada en dos niveles en el que el nivel de menor jerarquía, denominado esclavo, realiza tareas de control y el de mayor jerarquía, denominado maestro, realiza tareas de control y/o supervisión.
- **Cruce:** Un cruce puede ser aquel punto en el cual dos líneas, dos calles, vías o caminos se cortan mutuamente.
- **Clasificador:** Se utiliza en referencia al algoritmo utilizado para asignar un elemento entrante no etiquetado en una categoría concreta conocida.
- **CMYK:** Siglas de Cyan, Magenta y Key) es un modelo de color sustractivo que se utiliza en la impresión en colores. Es la versión moderna y más precisa del antiguo modelo tradicional de coloración (RYB), que se utiliza aún en pintura, impresiones a 3D y serigrafía. Permite representar una gama de colores más amplia que este último, y tiene una mejor adaptación a los medios industriales.
- **Codificación:** Cualquier operación que implique la asignación de un valor de símbolos o caracteres a un determinado mensaje verbal o no verbal con el propósito de transmitirlo a otros individuos o entidades que compartan el código.
- **Compuerta lógica:** Es una función booleana.
- **Convolución:** Es un operador matemático que transforma dos funciones f y g en una tercera función que en cierto sentido representa la magnitud en la que se superponen f y una versión trasladada e invertida de g .
- **Desfasamiento:** El número de segundos o porcentaje del ciclo que tarda en aparecer la indicación de luz verde en un semáforo, después de un instante dado, que se toma como punto de referencia de tiempo.
- **Diodo:** Válvula electrónica de dos electrodos que solo deja pasar la corriente en un sentido.
- **Ecotecnología:** Es un conjunto de técnicas aplicadas, y garantizan el uso de los recursos naturales de manera limpia, derivadas de algunas ciencias, que integra los campos de estudio de la ecología y la tecnología, usando los principios de la

ecología permacultura. Su objetivo es satisfacer las necesidades humanas minimizando el impacto ambiental a través del conocimiento de las estructuras y procesos de los ecosistemas y la sociedad. Se considera ecotecnología a todas las formas de ingeniería ecológica que reducen el daño a los ecosistemas, adopta fundamentos permaculturales, holísticos y de desarrollo sostenible, además de contar con una orientación precautoria de minimización de impacto en sus procesos y operación, reduciendo la huella ambiental.

- **Ecuación:** Igualdad entre dos expresiones que contiene una o más variables.
- **Entropía:** Es una magnitud física para un sistema termodinámico en equilibrio. Mide el número de microestados compatibles con el macro estado de equilibrio, también se puede decir que mide el grado de organización del sistema, o que es la razón de un incremento entre energía interna frente a un incremento de temperatura del sistema.
- **Escalabilidad:** Es la propiedad deseable de un sistema, una red o un proceso, que indica su habilidad para reaccionar y adaptarse sin perder calidad, o bien manejar el crecimiento continuo de trabajo de manera fluida, o bien para estar preparado para hacerse más grande.
- **Fotograma:** Imagen cinematográfica considerada aisladamente.
- **Gradiente:** Variación de una magnitud en función de la distancia, a partir de la línea en que esta variación es máxima en las magnitudes cuyo valor es distinto en los diversos puntos de una región del espacio.
- **Hardware:** Conjunto de elementos físicos o materiales que constituyen una computadora o un sistema informático.
- **Histéresis:** Es la tendencia de un material a conservar una de sus propiedades, en ausencia del estímulo que la ha generado. Podemos encontrar diferentes manifestaciones de este fenómeno.
- **HSL:** (del inglés Hue, Saturation, Lightness – Matiz, Saturación, Luminosidad), también llamado HSI (del inglés Hue, Saturation, Intensity – Matiz, Saturación, Intensidad), define un modelo de color en términos de sus componentes constituyentes.

- **HSV:** Transformación no lineal del espacio de color RGB, y se puede usar en progresiones de color.
- **Implantación:** Establecimiento y puesta en práctica de una norma, institución o costumbre.
- **Implementación:** Ejecución u/o puesta en marcha de una idea programada, ya sea, de una aplicación informática, un plan, modelo científico, diseño específico, estándar, algoritmo o política.
- **Interfaz:** La conexión funcional entre dos sistemas, programas, dispositivos o componentes de cualquier tipo, que proporciona una comunicación de distintos niveles permitiendo el intercambio de información.
- **Intersección:** Área donde dos o más vías se unen o cruzan, ya sea a nivel o desnivel, permitiendo o no la mezcla de las corrientes del tránsito.
- **Intervalo:** Cualquiera de las diversas subdivisiones del ciclo correspondiente a las indicaciones o colores del semáforo.
- **IT:** Tecnología de la información por sus siglas en inglés (Information Technology).
- **Iteración:** Acto de repetir un proceso con la intención de alcanzar una meta deseada, objetivo o resultado.
- **ITS:** Sistemas inteligentes de transporte por sus siglas en inglés (intelligent transport systems).
- **Matlab:** (abreviatura de MATrix LABoratory, "laboratorio de matrices") es una herramienta de software matemático que ofrece un entorno de desarrollo integrado (IDE) con un lenguaje de programación propio (lenguaje M).
- **Patrón:** El modelo que se emplea al establecer alguna magnitud o su muestra para replicarla.
- **Pixel:** Superficie homogénea más pequeña de las que componen una imagen, que se define por su brillo y color.
- **Ratio:** De una imagen, es la proporción entre su ancho y su altura. Se calcula dividiendo el ancho por la altura de la imagen visible en pantalla, y se expresa normalmente como (X, Y).

- **RGB:** Modelo de color basado en la síntesis aditiva, con el que es posible representar un color mediante la mezcla por adición de los tres colores de luz primarios.
- **Reconocimiento de Objetos:** Es la tarea para encontrar e identificar objetos en una imagen o secuencia de video.
- **Segmentación:** Acto y consecuencia de segmentar (es decir, de dividir o formar segmentos o porciones).
- **Semáforos:** Los semáforos son dispositivos electrónicos que sirven para ordenar y regular el tránsito de vehículos y peatones en calles y carreteras por medio de luces generalmente de color rojo, ámbar y verde, operados por una unidad de control.
- **Sensor:** Es un objeto capaz de variar una propiedad ante magnitudes físicas o químicas, llamadas variables de instrumentación, y transformarlas con un transductor en variables eléctricas.
- **Sistemas inteligentes de transporte:** Es un conjunto de soluciones tecnológicas de las telecomunicaciones y la informática (telemática) diseñadas para mejorar la operación y seguridad del transporte terrestre, tanto para carreteras como para vialidades urbanas.
- **Subsistema:** Es un sistema que es parte de otro sistema mayor que lo contiene.
- **Software:** Conjunto de programas y rutinas que permiten a la computadora realizar determinadas tareas.
- **Tecnologías de Uso Libre:** La tecnología libre surge como una extensión del concepto y filosofía del software libre aplicado a las tecnologías. La tecnología libre promueve los preceptos del conocimiento libre al protegerse con licencias de derechos de autor que respetan la libertad del usuario o consumidor de la tecnología.
- **Textura:** Es la disposición y el orden de los hilos en una tela. En el sector textil, el término también se utiliza para nombrar a la operación de tejer y a la superficie de una prenda.
- **Tránsito:** El flujo de vehículos en una vía de comunicación.

- **Tupla:** Representa un objeto único de datos implícitamente estructurados en una tabla.
- **Umbralización:** Es uno de los métodos más simples y eficientes de segmentación, segmentar quiere decir, separar o extraer las regiones de una imagen que nos interese estudiar o analizar, para lograr separar la región deseada se establece un valor que define el umbral, los pixeles cuya intensidad superen el umbral serán rechazado o aceptados, según sea el caso.
- **Vialidad:** Cualquier vía rural o urbana por donde transitan los vehículos de un lugar a otro.
- **Vialidad urbana:** Conjunto integrado de vías de uso común que conforman la traza urbana, su función es facilitar el tránsito seguro y eficiente de personas y vehículos.
- **Visión Artificial:** Es una disciplina científica que incluye métodos para adquirir, procesar, analizar y comprender las imágenes del mundo real con el fin de producir información numérica o simbólica para que puedan ser tratados por un computador.
- **Zona urbana:** Área habitada o urbanizada, definida por los aspectos geográfico, ecológico, demográfico, social, económico, etc. es decir, es la ciudad misma más el área contigua edificada, con uso de suelo no agrícola.