


	GESTIÓN SERVICIOS BIBLIOTECARIOS					  	
	CARTA DE AUTORIZACIÓN						
CÓDIGO	AP-BIB-FO-06	VERSIÓN	1	VIGENCIA	2014	PÁGINA	1 de 2

Neiva, 31 de agosto de 2015

Señores

CENTRO DE INFORMACIÓN Y DOCUMENTACIÓN

UNIVERSIDAD SURCOLOMBIANA

Ciudad

El (Los) suscrito(s):

Jonathan Miguel Díaz Vargas, con C.C. No. 1'075.233.669,

Manuel Fernando Ossa Vargas, con C.C. No. 1'075.211.707,

autor(es) de la tesis y/o trabajo de grado o titulado DISEÑO ALGORITMO DETECCIÓN DE TRÁFICO VEHICULAR CON VISION ARTIFICIAL presentado y aprobado en el año 2015 como requisito para optar al título de Ingeniero Electrónico;

autorizo(amos) al CENTRO DE INFORMACIÓN Y DOCUMENTACIÓN de la Universidad Surcolombiana para que con fines académicos, muestre al país y el exterior la producción intelectual de la Universidad Surcolombiana, a través de la visibilidad de su contenido de la siguiente manera:

Los usuarios puedan consultar el contenido de este trabajo de grado en los sitios web que administra la Universidad, en bases de datos, repositorio digital, catálogos y en otros sitios web, redes y sistemas de información nacionales e internacionales “open access” y en las redes de información con las cuales tenga convenio la Institución.

- Permita la consulta, la reproducción y préstamo a los usuarios interesados en el contenido de este trabajo, para todos los usos que tengan finalidad académica, ya sea en formato Cd-Rom o digital desde internet, intranet, etc., y en general para cualquier formato conocido o por conocer, dentro de los términos establecidos en la Ley 23 de 1982, Ley 44 de 1993, Decisión Andina 351 de 1993, Decreto 460 de 1995 y demás normas generales sobre la materia.

- Continúo conservando los correspondientes derechos sin modificación o restricción alguna; puesto que de acuerdo con la legislación colombiana aplicable, el presente es un acuerdo jurídico que en ningún caso conlleva la enajenación del derecho de autor y sus conexos.

De conformidad con lo establecido en el artículo 30 de la Ley 23 de 1982 y el artículo 11 de la Decisión Andina 351 de 1993, “Los derechos morales sobre el trabajo son propiedad de los autores”, los cuales son irrenunciables, imprescriptibles, inembargables e inalienables.



GESTIÓN SERVICIOS BIBLIOTECARIOS

CARTA DE AUTORIZACIÓN



CÓDIGO

AP-BIB-FO-06

VERSIÓN

1

VIGENCIA

2014

PÁGINA





2 de 2

EL AUTOR/ESTUDIANTE: Jonathan M. Díaz V.

Firma: Jonathan Miguel Diaz Vargas

EL AUTOR/ESTUDIANTE: Manuel F. Ossa V.

Firma: Manuel F. Ossa V.

	GESTIÓN SERVICIOS BIBLIOTECARIOS					  	
	DESCRIPCIÓN DE LA TESIS Y/O TRABAJOS DE GRADO						
CÓDIGO	AP-BIB-FO-07	VERSIÓN	1	VIGENCIA	2014	PÁGINA	1 de 3

TÍTULO COMPLETO DEL TRABAJO: DISEÑO ALGORITMO DETECCIÓN DE TRÁFICO VEHICULAR CON VISION ARTIFICIAL

AUTOR O AUTORES:

Primero y Segundo Apellido	Primero y Segundo Nombre
Díaz Vargas	Jonathan Miguel
Ossa Vargas	Manuel Fernando

DIRECTOR Y CODIRECTOR TESIS:

Primero y Segundo Apellido	Primero y Segundo Nombre
Salgado Patrón	José De Jesús

PARA OPTAR AL TÍTULO DE: INGENIERO ELECTRÓNICO

FACULTAD: INGENIERIA

PROGRAMA O POSGRADO: INGENIERIA ELECTRÓNICA

CIUDAD: NEIVA AÑO DE PRESENTACIÓN: 2015 NÚMERO DE PÁGINAS: 60





TIPO DE ILUSTRACIONES (Marcar con una X):

Diagramas X Fotografías X Grabaciones en discos ___ Ilustraciones en general X Grabados ___ Láminas ___
 Litografías ___ Mapas X Música impresa ___ Planos ___ Retratos ___ Sin ilustraciones ___ Tablas o Cuadros X

SOFTWARE requerido y/o especializado para la lectura del documento: Adobe Acrobat Reader DC (PDF).

MATERIAL ANEXO:

La versión vigente y controlada de este documento, solo podrá ser consultada a través del sitio web Institucional www.usco.edu.co, link Sistema Gestión de Calidad. La copia o impresión diferente a la publicada, será considerada como documento no controlado y su uso indebido no es de responsabilidad de la Universidad Surcolombiana.

	GESTIÓN SERVICIOS BIBLIOTECARIOS					  	
	DESCRIPCIÓN DE LA TESIS Y/O TRABAJOS DE GRADO						
CÓDIGO	AP-BIB-FO-07	VERSIÓN	1	VIGENCIA	2014	PÁGINA	2 de 3

PREMIO O DISTINCIÓN (En caso de ser LAUREADAS o Meritoria):

PALABRAS CLAVES EN ESPAÑOL E INGLÉS:

<u>Español</u>	<u>Inglés</u>	<u>Español</u>	<u>Inglés</u>
1. <u>Imágenes</u>	<u>Images</u>	6. <u>Movilidad</u>	<u>Mobility</u>
2. <u>Visión artificial</u>	<u>Artificial vision</u>	7. <u>Flujo Vehicular</u>	<u>Traffic flow</u>
3. <u>OpenCv</u>	<u>OpenCV</u>	8. <u>Detección automática</u>	<u>Auto Detect</u>
4. <u>Video</u>	<u>Video</u>	9. <u>Cámara</u>	<u>Camera</u>
5. <u>Procesamiento</u>	<u>Processing</u>	10. <u>Flujo óptico</u>	<u>Optical flow</u>

RESUMEN DEL CONTENIDO: (Máximo 250 palabras)

Este proyecto surge con el propósito de crear una herramienta para cuantificar el flujo vehicular, indispensable en la determinación de soluciones más efectivas en el mejoramiento de la movilidad de la ciudad. El trabajo consiste primero en detectar automáticamente los automotores, y segundo, determinar la velocidad con la que circulan cada uno de ellos, mientras ingresan a una de las vías más importantes de Neiva, por medio de un sensor de imágenes, cuyos datos son obtenidos y procesados en tiempo real. La aplicación entrega un algoritmo trabajado sobre la librería OpenCv, diseñada exclusivamente para el procesamiento de imágenes y cuyo resultado no sólo se presenta en texto, sino que además lo presenta de forma gráfica sobre el video que se capta instantáneamente.



GESTIÓN SERVICIOS BIBLIOTECARIOS

DESCRIPCIÓN DE LA TESIS Y/O TRABAJOS DE GRADO



CÓDIGO

AP-BIB-FO-07

VERSIÓN

1

VIGENCIA

2014

PÁGINA

3 de 3

ABSTRACT: (Máximo 250 palabras)

This project was created with the purpose of creating a tool to measure traffic flow, it is essential in determining more effective solutions to improve the mobility of the city. The first actions is to detect cars automatically and second, to determine the speed each of them circulates, while they are entering one of the most important roads in Neiva; it is measured through an image sensor which data are obtained and processed in real time . The application delivers algorithm which is worked on the library programming function OpenCV, it was designed for image processing exclusively and the result is presented not only text but also graphically on the video that is captured instantly.

APROBACION DE LA TESIS

Nombre Jurado: Martin Bravo Obando

Firma:

Nombre Jurado: Johan Julián Molina

Firma:

DISEÑO DE ALGORITMO PARA DETECCIÓN DE TRÁFICO VEHICULAR
USANDO TÉCNICAS DE VISIÓN ARTIFICIAL

MANUEL FERNANDO OSSA VARGAS
JONATHAN MIGUEL DÍAZ VARGAS

UNIVERSIDAD SURCOLOMBIANA
FACULTAD DE INGENIERÍA
INGENIERÍA ELECTRÓNICA
NEIVA – HUILA
2015

DISEÑO DE ALGORITMO PARA DETECCIÓN DE TRÁFICO VEHICULAR
USANDO TÉCNICAS DE VISIÓN ARTIFICIAL

MANUEL FERNANDO OSSA VARGAS
JONATHAN MIGUEL DÍAZ VARGAS

DIRECTOR:
Ing. JOSÉ DE JESÚS SALGADO PATRÓN

Trabajo de grado presentado para optar al título de
Ingeniero Electrónico

UNIVERSIDAD SURCOLOMBIANA
FACULTAD DE INGENIERÍA
INGENIERÍA ELECTRÓNICA
NEIVA – HUILA
2015

A Dios, gran maestro y hacedor de lo imposible.
A mi esposa, mi compañera y ayuda idónea.
A los mejores papás del mundo.
Y a la niña de mis ojos.

Manuel Fernando Ossa

A Dios quien me ha dado fortaleza para continuar cuando a punto de desistir he
estado.
A mis padres que han sabido formarme con buenos sentimientos, hábitos y me
han dado su apoyo incondicional.

Jonathan Miguel Díaz

Doy gracias a Dios, que me ha permitido gozarme de esta y muchas más experiencias, en especial de la vida misma. Por escoger cuidadosamente a cada una de las personas que aquí menciono y a las que de una u otra forma estuvieron cerca. A mis padres, quienes con su esfuerzo y constancia me inculcaron determinación, disciplina y amor por el estudio. A mi esposa que no sólo ha sido mi compañera, sino también, mi inspiración, y en especial por ese regalo maravilloso de convertirme en papá. A mis compañeros y amigos con quienes compartimos muchos momentos estresantes y muchos más agradables. Al programa ingeniería electrónica, sus docentes, quienes además de compartir su conocimiento, me participaron una grata amistad, en especial al Ingeniero José de Jesús Salgado, director de este proyecto; A los ingenieros Martín Bravo y Julián Molina, por sus significativos y valiosos aportes a este trabajo; a Sonia, Chucho, Carlos y Eduardo por su presta colaboración y apoyo recibido; Y por supuesto al alma máter que un día me aceptó como miembro de esta gran familia Surcolombiana, de la cual hoy me siento muy orgulloso.

Manuel Fernando Ossa

Doy gracias a mis hermanas que siempre han estado junto a mí brindándome su apoyo. A mis amigos y compañeros por su apoyo incondicional a lo largo de esta trayectoria, a mis profesores, por enseñarme, aconsejarme e instruirme, pues gracias a ellos se hace realidad este logro.

A aquellas personas especiales que han pasado por mi vida y gracias a su apoyo hoy alcanzo este logro tan importante en mi vida y soy cada día una mejor persona.

Jonathan Miguel Díaz

RESUMEN

Este trabajo plasma paso a paso, el desarrollo del diseño de un algoritmo para detección de tráfico vehicular usando las propias técnicas de visión por computador, con el objetivo no solo de llevar a cabo el conteo de vehículos que transitan sobre determinada vía, sino que además permita conocer la velocidad media con la que viajan estos vehículos.

En la primera parte, se presenta la descripción del problema, junto con los requerimientos necesarios que se tuvieron en cuenta para la formulación de este proyecto. Ya en la segunda parte, se presenta en detalle del desarrollo, definiciones y el marco teórico del proyecto y como tal del algoritmo. Para luego pasar a la presentación de los resultados, donde se expone claramente los problemas enfrentados a lo largo del desarrollo del algoritmo y la solución encontrada para cada uno de ellos, Finalmente se muestran las conclusiones y recomendaciones sugeridas para mejorar la efectividad del mismo, con el objetivo de sentar las bases para futuras investigaciones al respecto.

Como se puede ver, el producto final de esta investigación es una aplicación, propuesta por la ingeniería electrónica que funcione como una herramienta indispensable para la ingeniería de tráfico, en la determinación de indicadores como el número de vehículos y la velocidad media en una vía de la ciudad, ya que al conocer cuál es el comportamiento de la movilidad, puede aportar soluciones muy eficientes a esta problemática actual y con ello, mejorar la calidad de vida de los ciudadanos.

ABSTRACT

This paper presents step by step development of the design of an algorithm for detection of vehicular traffic using own computer vision techniques, aiming not only to perform the counting of vehicles traveling on a particular route, but also allows to know the average speed with which these vehicles travel.

In the first part, the problem description is presented, along with the necessary requirements to be taken into account in the formulation of this project. In the second part, it presents in detail the development, definitions and the theoretical framework of the project and as such the algorithm. Then move to the presentation of the results, which clearly sets out the problems faced during the development of the algorithm and the solution found for each of them, Finally, conclusions and suggested recommendations are shown to enhance the effectiveness thereof with the objective of laying the groundwork for future investigations.

As you can see, the final product of this research is an application proposed by the electrical engineering, function as an indispensable tool for traffic engineering, in determining indicators such as the number of vehicles and average speed in a way of the city. Since knowing what is the behavior of the mobility, can provide highly efficient solutions to the current problems and thereby improve the quality of life of citizens.

CONTENIDO

	Pág.
INTRODUCCIÓN	11
1. DESCRIPCIÓN DEL PROBLEMA.....	13
1.1 REQUERIMIENTOS	14
1.1.1 Requerimientos de Hardware.	14
1.1.1.1 Sensor.....	14
1.1.1.2 Equipo de cómputo.	15
1.1.2 Requerimientos de Software.....	15
1.1.3 Requerimientos estructurales	15
2. DESARROLLO Y CUMPLIMIENTO DE REQUERIMIENTOS.....	16
2.1 DESARROLLO	16
2.1.1 Elementos del Tránsito.	16
2.1.1.1 Usuario.....	16
2.1.1.2 Vialidad.	16
2.1.1.3 Vehículos.	17
2.1.2 Características del Tránsito.	18
2.1.3 Flujo Vehicular.	18
2.2 TEORÍA BÁSICA VISIÓN ARTIFICIAL.....	19
2.2.1 Fundamentos	19
2.2.1.1 Imagen digital.....	19
2.2.1.2 Cuantificación	19
2.2.1.3 Representación del color.	20
2.2.2 Etapas de un proceso de visión artificial.	21
2.2.2.1 Adquisición de imagen.....	23
2.2.2.2 Pre-procesamiento.....	25
2.2.2.3 Segmentación.	27
2.2.2.4 Extracción de las características.....	31
2.2.2.5 Reconocimiento y Localización de Objetos.....	32
2.2.2.6 Interpretación de la Escena.	35
2.3 CUMPLIMIENTO DE REQUERIMIENTOS.....	36
2.4 DESARROLLO DEL SOFTWARE	37
2.4.1 Algoritmo de Procesamiento de Imágenes.	38
2.4.1.1 Adquisición de Imágenes.....	39
2.4.1.2 Preprocesamiento.....	39
2.4.1.3 Segmentación	39
2.4.1.4 Extracción de características.	40
2.4.1.4 Reconocimiento y localización de objetos.....	40
2.4.1.5 Interpretación de la escena.....	40

2.4.2 Desarrollo de Aplicativo	40
3. ANÁLISIS DE RESULTADOS	42
3.1 CONTEO VEHICULAR.....	42
3.1.1 Imágenes Diurnas.....	42
3.1.2 Imágenes Nocturnas.....	47
3.2 VELOCIDAD.....	52
3.2.1 Exactitud por Trayectoria.....	52
3.2.2 Por Sensor.....	54
3.3 SOMBRAS.....	55
4. CONCLUSIONES.....	57
5. RECOMENDACIONES	59
BIBLIOGRAFIA.....	60

LISTA DE FIGURAS

	Pág.
Figura 1. Ubicación espacial del sistema.....	17
Figura 2. Imagen RGB vs Imagen Binaria.	20
Figura 3. Dimensiones de las imágenes.....	21
Figura 4. Etapas del proyecto.....	22
Figura 5. Espectro electromagnético visible.....	23
Figura 6. Relaciones de Aspecto.....	25
Figura 7. Comparativo de cámaras utilizadas.....	26
Figura 8. Sectorización de la escena.....	27
Figura 9. Área de referencia.....	28
Figura 10. Sustracción de Fondo.....	28
Figura 11. Dilatación de la escena.....	29
Figura 12. Máscara de enfoque.....	30
Figura 13. Normalización máscara de Enfoque.....	30
Figura 14. Máscara de Enfoque invertida.....	31
Figura 15. Filtro de Canny.....	31
Figura 16. Área en píxeles de una moto.....	32
Figura 17. Condición inicial para el conteo.....	33
Figura 18. Suposiciones para el Método Lucas-Kanade.....	34
Figura 19. Interpretación de la escena.....	36
Figura 20. Acerca de Software de compilación.....	37
Figura 21 Diagrama de Flujo del Algoritmo.....	38
Figura 22. Presentación de aplicativo.....	41
Figura 23. Causas de errores en imágenes diurnas.....	43
Figura 24. Formas de puntos ciegos.....	44
Figura 25. Punto ciego Horizontal.....	44
Figura 26. Eliminación punto ciego horizontal.....	45

Figura 27. Geometría del punto ciego vertical.	45
Figura 28. Relación Distancia vehículo-sensor vs Longitud punto ciego.	46
Figura 29. Ejemplo de punto ciego.	46
Figura 30. Histograma escena diurna.	47
Figura 31. Histograma escena nocturna (Gris).	48
Figura 32. Secuencia detección faros.	49
Figura 33. Ilustración faro apagado.	49
Figura 34. Ilustración faros desalineados.	50
Figura 35. Causas de errores en imágenes nocturnas	51
Figura 36. Problemas de Iluminación.....	51
Figura 37. Trayectorias de desplazamiento.	52
Figura 38. Geometría de las trayectorias.	53
Figura 39. Registro de fin de desplazamiento.....	54
Figura 40. Error en la velocidad según los fotogramas.....	55
Figura 41. Sombras.	56

INTRODUCCIÓN

El incremento de vehículos en las vías de todo el mundo y hablando específicamente de Colombia, hace que la ingeniería se plantee nuevos retos en las soluciones a estos, pues la necesidad de vías más estructuradas y físicamente óptimas y resistentes es cada vez mayor, pues todo esto busca satisfacer las necesidades de los habitantes de las ciudades, sin detenerse a analizar que ése, es también el causante de uno de los aspectos más conflictivos del sistema urbano en función de su sostenibilidad: la contaminación ambiental en sus diferentes formas, la ocupación extensiva del suelo y la seguridad del tráfico.

Por esta razón, y tomando como referencia la ciudad de Neiva, en la que en los últimos años el flujo vehicular se ha incrementado notoriamente y literalmente las vías se han quedado cortas en capacidad especialmente en las horas pico del día, plantea y desarrolla una herramienta basada en técnicas de visión artificial que detecte de forma automática el tráfico vehicular, con el propósito de establecer las cifras reales de la cantidad de vehículos que transitan sobre una vía, la velocidad y por ende todos los indicadores necesarios para la implementación de soluciones efectivas a la problemática mencionada.

Es entonces, donde aparece la ingeniería de tránsito, “fase de la ingeniería de transporte que tiene que ver con la planeación, el proyecto geométrico y la operación del tránsito por calles y carreteras, sus redes, terminales, tierras adyacentes y su relación con otros modos de transporte” [1]

Por otro lado, la visión artificial, técnica que se desprende de la inteligencia artificial y que pretende facultar a un computador para que intérprete una imagen, lo más parecido a la interpretación que puede hacer un ser humano. Para lograr este fin es necesario tres componentes básicos, el primero es un sensor, encargado de capturar la imagen, escena que se desee analizar, y que se asemeja al órgano receptor más importante del hombre, el ojo. Como segunda medida se requiere de un equipo de cómputo encargado de procesar la información, las veces del cerebro humano. Por último se requiere de un software especializado que interprete la escena, lo que haría las veces de la experiencia del cerebro humano, que basado en experiencias anteriores, identifica objetos particulares en casi cualquier situación.

Según lo anterior, este proyecto se plantea un objetivo principal de diseñar un algoritmo efectivo que cuantifique el flujo vehicular sobre la avenida 26 en sentido

¹ CAL, Rafael y REYES, Mayor y CÁRDENAS, James. (1994). Ingeniería de tránsito: Fundamentos y aplicaciones. 7ª Ed. Alfaomega. México. Pág. 29.

Sur-Norte, usando las mejores técnicas y las más efectivas de visión artificial, y así tener por primera vez datos reales, sin usar estimaciones y además, obtenerlos en tiempo real. Esto con el fin de conocer de primera mano los indicadores sobre los que se debe centrar toda decisión en movilidad, como por ejemplo, los tiempos de duración de las luces verdes en los semáforos, e incluso la determinación de otras medidas de mayor trascendencia, como pico y placa, y medios de transporte masivo, entre otras.

Para alcanzar el objetivo principal de este proyecto fue necesario determinar tanto tipo de sensor para el funcionamiento idóneo del sistema, como las técnicas más eficientes en la detección de vehículos, y por supuesto los requerimientos de hardware necesarios para realizar el procesamiento de imágenes en el menor tiempo posible.

1. DESCRIPCIÓN DEL PROBLEMA.

La congestión vehicular, que se puede definir como aquella condición del flujo vehicular cuando éste se satura por exceso de demanda de las vías, debido a la gran afluencia de vehículos, en especial en horas pico^[2], y que aumenta significativamente los tiempos de viaje y el consumo de combustible, produce grandes efectos ambientales en la ciudad como aumento en la contaminación auditiva, al haber conductores que accionan sus bocinas constantemente tras presentarse un embotellamiento, mayores emisiones de gases tóxicos como dióxido y monóxido de carbono, óxido de azufre, etc. Y por supuesto, un aumento en la temperatura ambiente, al permanecer los vehículos detenidos por mucho más tiempo, favoreciendo con ello el calentamiento Global.

Otra consecuencia importante es el incremento en la accidentalidad, ya que en los embotellamientos los vehículos están muy cerca el uno del otro, sumado a la forma en que conducen las personas que pierden la calma mientras esperan salir de estas situaciones, producen choques leves, y aunque las lesiones normalmente no son graves por las bajas velocidades, los traumatismos por estos acontecimientos pueden llevar incluso a un colapso en el sector.

Muchas soluciones se han sugerido y muchas de ellas se han aplicado al sistema, entre las cuales están por ejemplo los sistemas de transporte masivos que no sólo buscan reducir la cantidad de vehículos en circulación, sino que buscan maximizar la eficiencia en el uso del espacio, transportando gran cantidad de pasajeros de una manera cómoda, rápida y segura en un solo vehículo.

Ciudades como Bogotá, Cali, Medellín, Armenia, entre otras, ya disfrutaban de varias de estas medidas, en Bogotá, por ejemplo, es notable que el sistema ofrece una fluidez en el servicio, pero se presentan fallas en cuanto a comodidad y a seguridad de los pasajeros. Por otro lado, la restricción del pico y placa, que exige que todos los vehículos estén algunos días por fuera del sistema vial de la ciudad, tuvo muchos tropiezos por el descontento de los propietarios de vehículos. Por último, y considerada dentro de las más eficientes, que aprovecha al máximo las condiciones físicas de la ciudad, aparece la implementación de nuevas tecnologías para el control del tránsito, que ya sea en los puntos más concurridos, ordenan el tráfico y determinan secuencialmente el derecho a la circulación en los respectivos sentidos del cruce, cada uno con un tiempo específico para cada situación.

²Artículo disponible en <http://es.wikipedia.org/wiki/Congesti%C3%B3n_vehicular>

En la ciudad de Neiva, es muy frecuente encontrarse con episodios de congestión vehicular, y aunque no se ha implementado el pico y placa ni se goza de un sistema de transporte público masivo, sí se han hecho inversiones considerables en el sistema de semaforización.

El gran problema luego de la implementación de cualquier de estas alternativas, obedece al riesgo de no corresponder la relación costo/beneficio, pues muchas veces, la problemática no se soluciona del todo, por desconocimiento de la causa raíz del problema, debido a que las cifras de movilidad se levantan de forma manual con conteo visual y estimaciones del personal técnico.

De allí surge primero la necesidad de un mecanismo que de forma automática establezca el número de vehículos (volumen) y la velocidad con que transitan sobre una vía específica (flujo).

1.1 REQUERIMIENTOS

1.1.1 Requerimientos de Hardware. En el desarrollo de este proyecto, fue necesario contar con los dispositivos necesarios, cuyas características técnicas permiten el cumplimiento de los objetivos propuestos.

1.1.1.1 Sensor. La determinación de la cámara idónea fue el primer desafío, teniendo en cuenta la importancia de dos parámetros fundamentales, resolución y velocidad. Ya que una alta resolución aumentaría el procesamiento y capacidad de almacenamiento, igual que tener más cuadros ó imágenes en un segundo (velocidad), pero al captar imágenes en una resolución menor, se podría perder información vital en la escena, como el detalle de los objetos capturados; en cuanto a la velocidad de la cámara, si en una escena, un objeto se mueve muy rápido con respecto a la velocidad de la cámara, sus bordes se verían muy borrosos o difusos, dificultando la detección automática de dicho objeto.

Por lo anterior se determinó usar una resolución en alta definición (1280x720p), para contar con la mayor información posible, aun cuando se sabe que puede ser una resolución muy alta para esta aplicación, por lo que se estudiará el alcance de reducir esta resolución sin afectar los resultados y así reducir costos.

En cuanto a la velocidad del dispositivo y teniendo en cuenta que sobre la vía escogida, se esperaban velocidades superiores a los 80km/h, se determinó la necesidad de usar dos dispositivos, uno de 30 cuadros/segundo y otro con el doble de capacidad (60fps).

Un último requerimiento del sensor, es la capacidad de capturar imágenes con muy bajos niveles de luz sin perder información, ya sea con un sistema de ganancia muy alto o con ayuda de otros sistemas como el infrarrojo.

1.1.1.2 Equipo de cómputo. En este ítem el requerimiento se centra en la capacidad de procesamiento, pues al tener resoluciones en alta definición, la exigencia de esta capacidad es alta.

De tal forma, se hace necesario trabajar con un procesador de al menos 2 núcleos, y 2.8GHz de velocidad de procesamiento, parámetros que cumplen los procesadores de la familia Core i7, con el cual se desarrollará este proyecto.

1.1.2 Requerimientos de Software. Para este requerimiento se necesita primero una librería que trabaje con imágenes y que además sea lo suficientemente robusta para realizar procesamientos extensos y específicos de visión artificial, y que a la vez, permita la modificación de su código, para mejorar los resultados de este proyecto.

Por tal razón se requiere trabajar sobre la librería OpenCV, que además maneja un código abierto, ofreciendo una muy extensa gama de posibilidades para el procesamiento de imágenes.

Otro de los requerimientos para este proyecto, se basa en el compilador, que debe ser compatible con la librería seleccionada y el lenguaje de programación base (lenguaje C). Así es que se requiere un compilador como el QtCreator, con el cual se desarrollará toda la programación tanto del algoritmo como del aplicativo.

1.1.3 Requerimientos estructurales. El último requerimiento, es el soporte o estructura física sobre el cual se instalará la cámara de video, el cual, debe tener por lo menos una altura de 4,5 metros; Debe ser lo suficiente resistente, para soportar el peso de las cámaras y ofrecer resistencia a los movimientos del viento, y en lo posible, resistir el peso del operador tecnológico, para efectos de la instalación y mantenimiento del sensor.

2. DESARROLLO Y CUMPLIMIENTO DE REQUERIMIENTOS

2.1 DESARROLLO

Dentro del desarrollo de este proyecto, se destaca la importancia de la ingeniería electrónica, como herramienta de gran utilidad en otros ámbitos o áreas diferentes, como es el caso de la ingeniería de Tránsito, que con su teoría busca solucionar los inconvenientes que trae el crecimiento acelerado de la población en una área urbana, que con sus redes viales, se rezaga a este crecimiento.

2.1.1 Elementos del Tránsito. Dentro de la ingeniería de tránsito se identifican tres elementos básicos que la componen, y sobre los cuales se desarrollará este proyecto. A continuación se ofrece una pequeña introducción de estos elementos que la conforman, y la forma en que interactúan.³

2.1.1.1 Usuario. Es el ser humano, el usuario del sistema, en cualquiera de sus tres formas, conductor, peatón y pasajero, aunque es necesario aclarar que no sólo es el usuario del sistema, sino también actor del mismo, pues del comportamiento de los conductores y peatones, depende la eficiencia del sistema en los demás usuarios.

2.1.1.2 Vialidad. Corresponde a toda la infraestructura de la red vial, diseñada y construida, para soportar el flujo de los vehículos que circulan de un punto a otro. Para el caso de este proyecto, hace referencia a las vías de la ciudad de Neiva, en particular la avenida 26, sobre la que se desarrollará este proyecto, punto crítico en cuanto movilidad, pues es éste un punto de intersección determinante para el tráfico hacia el norte de la ciudad.

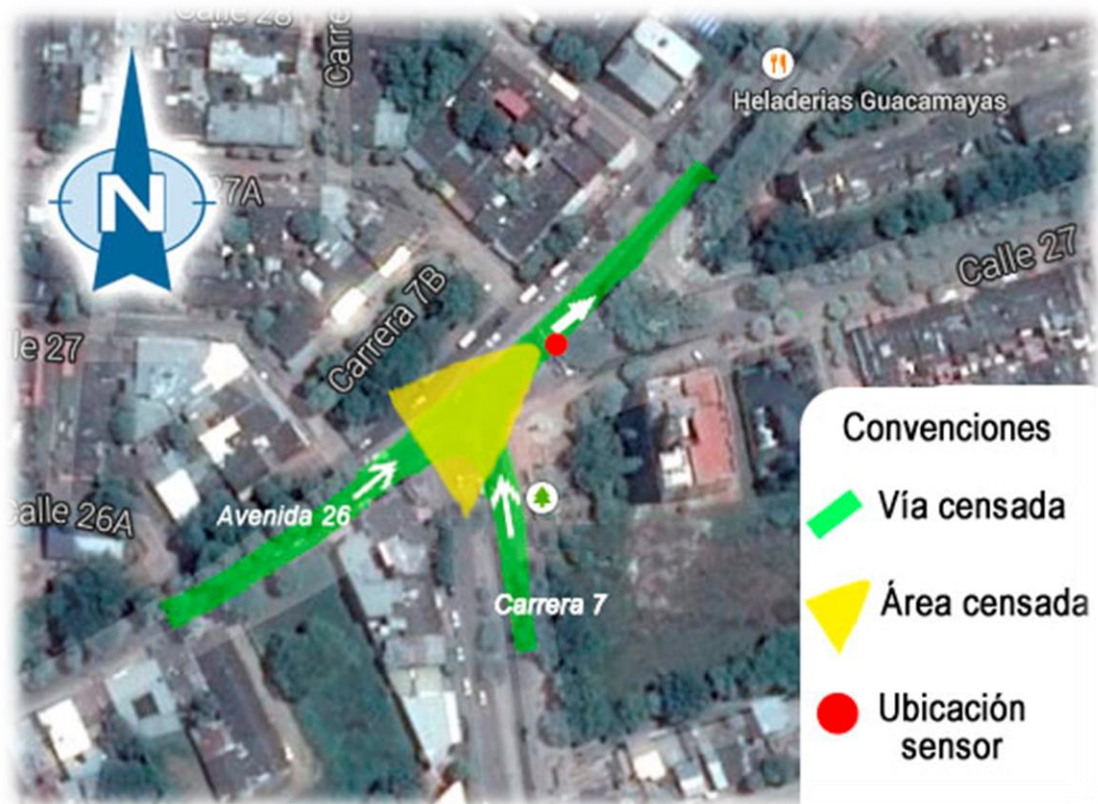
Es el elemento, con los costos más elevados tanto para el diseño y construcción, como de su mismo mantenimiento, y es posiblemente la razón por la cual, su crecimiento y progreso es bastante lento.

Dentro de las vías de la ciudad de Neiva, y al resaltar la importancia de la intersección entre las carreras Séptima y avenida 26, al norte de la ciudad, se escoge como el sector donde se captarán las imágenes con las que se desarrollará el algoritmo propuesto, buscando obtener al final, los indicadores

³ CAL, Rafael y REYES, Mayor y CÁRDENAS, James. (1994). Ingeniería de tránsito: Fundamentos y aplicaciones. 7ª Ed. Alfaomega. México.

reales del flujo vehicular sobre la avenida 26 hacia el Norte, luego que recibe el tráfico proveniente del centro de la ciudad por la carrera séptima.

Figura 1. Ubicación espacial del sistema.



2.1.1.3 Vehículos. Elemento esencial en el funcionamiento de todo el sistema, ya que son éstos los de mayor crecimiento, aumentando a su vez los usuarios, reduciendo cada vez más la capacidad de las redes viales de todo el mundo, incluyendo las de esta región.

Prueba de ello, son los más de 7000 vehículos matriculados en los últimos dos años en el departamento del Huila, aumentando la ocupación en las vías del departamento, de los cuales sólo en Neiva fueron matriculados 6211 automotores. Estas cifras⁴ no sólo demuestran el gran crecimiento del parque automotor en la ciudad, sino que al compararlas con las cifras del crecimiento o desarrollo de la malla vial, en el mismo período de tiempo, se evidencia más un atraso en el

⁴ Comité Automotor colombiano, Informes del sector automotor 2013 y 2014.

progreso de la ciudad, pues al aumentar el parque automotor sin que exista un crecimiento o mejoramiento de las vías, la movilidad en Neiva, durante en los próximos años se vería seriamente comprometida.

2.1.2 Características del Tránsito. Si se observa el tránsito, como un sistema dinámico, es necesario analizar además de los tres elementos básicos que la conforman, dos factores determinantes que influyen en el comportamiento del mismo, ya que las vías son las que soportan la carga de los vehículos, que en cantidades y estando en movimiento, se convierten en un volumen de tránsito, con características muy bien definidas, pues ocupan un espacio (*espaciales*) y tardan un tiempo (*temporales*).⁵ La característica espacial, es propia de cada trayecto escogido por los usuarios que actúan como conductores, aunque al final lo que se tiene en cuenta es el comportamiento de cada vehículo. La característica temporal se refiere al tiempo empleado en realizar cada uno de los trayectos.

Estas características, al ser esenciales en el planteamiento del problema de cualquier proyecto encaminado a mejorar la movilidad de una localidad, deben ser muy bien identificadas y cuantificadas, teniendo en cuenta, que las vías al ser las de menor crecimiento, se pueden tomar prácticamente como constantes, quedando entonces el vehículo como actor principal dentro de este sistema tan dinámico. De esta forma es el tiempo, la característica determinante dentro del sistema, pues no sólo se refiere a la velocidad de desplazamiento, sino que además contempla los tiempos de espera en los semáforos y filas en tramos congestionados.

2.1.3 Flujo Vehicular. Combinando los elementos y el comportamiento del tráfico, se hace necesario conocer la forma en que circular los vehículos por las redes viales, conociendo los componentes que más inciden en la determinación de este comportamiento.

Dentro de los indicadores más importantes y a la vez determinante, y con los cuales se trabajarán en este proyecto están: El volumen y velocidad del flujo, considerando el volumen, como la cantidad de vehículos que circulan por un punto o sección transversal de una vía en un solo sentido, en un tiempo dado que puede ser un minuto, día, mes o incluso un año, y aunque el volumen tiene otros componentes más específicos como espacio y tiempo (horas pico, festividades, etc), es éste el que se pretende conocer una vez finalizado este proyecto.

⁵ CAL, Rafael y REYES, Mayor y CÁRDENAS, James. (1994). Ingeniería de tránsito: Fundamentos y aplicaciones. 7ª Ed. Alfaomega. México.

Por otro lado la velocidad es uno de los indicadores más utilizados en la determinación del nivel de eficiencia de un sistema vial, y corresponde a la relación entre la distancia recorrida y el tiempo empleado en el mismo trayecto, aunque cabe aclarar que esta definición es bastante genérica, ya que este indicador no es constante, requiriendo conocer primero la velocidad de punto, de recorrido, de marcha, etc. El segundo indicador que se busca determinar es la velocidad de punto.

2.2 TEORÍA BÁSICA VISIÓN ARTIFICIAL

La visión artificial, una técnica que se desprende de la inteligencia artificial y que pretende facultar a un computador para que interprete una imagen, lo más parecido a la interpretación que puede hacer un ser humano.

Para lograr este fin es necesario tres componentes básicos, el primero es un sensor, encargado de capturar en una imagen, la escena que se desee analizar, y que analógicamente a la visión humana se asemeja al órgano más importante, el ojo. Como segunda medida se requiere de un equipo de cómputo con suficiente capacidad de procesamiento, las veces del cerebro humano. Por último, se requiere de un software especializado para obtener la interpretación deseada de la escena, lo que haría las veces del cerebro humano, la cual identifica el objeto, basado en experiencias anteriores.

2.2.1 Fundamentos. Para establecer la importancia de la visión artificial dentro de este proyecto, es necesario realizar un bosquejo teórico acerca de los temas y definiciones más relevantes.

2.2.1.1 Imagen digital⁶. Una imagen digital es una representación de un arreglo bidimensional o matriz que almacena una cantidad determinada de números que establecen la intensidad lumínica capturada en una escena cualquiera donde exista luz. La cantidad de elementos de esta matriz la determina la resolución del sensor, que determina cuantos elementos utilizará para almacenar la información, y ésta al ser un arreglo bidimensional se expresa en medidas de ancho por largo, siendo el pixel, su unidad principal. Entre más pixeles tenga una imagen, más detalles son capturados de la escena, sin embargo, el uso excesivo de estos pixeles, demanda grandes capacidades de almacenamiento, como también extiende los tiempos de procesamiento.

2.2.1.2 Cuantificación. La cuantificación se define como la técnica que comprime un rango de valores a un único valor, y pretende darle un valor a la intensidad de

⁶ https://es.wikipedia.org/wiki/Imagen_digital

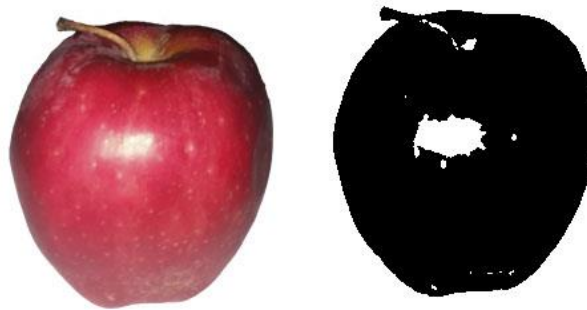
luz captada por un área determinada en la escena o pixel. El número de valores de intensidad depende de la cantidad de información que se requiera para su respectivo procesamiento.

En las imágenes digitales la intensidad lumínica se puede cuantificar en 2^n valores, donde n puede valer desde 1 hasta 8, siendo el máximo valor (8) el más usado, ya que permite trabajar con 256 valores diferentes de intensidades (2^8). Cantidad que excede fácilmente la capacidad humana de cuantificación

$$f(x) = 2^n ; n=1, 2, 3$$

La cuantificación con la menor cantidad de valores es la binaria, cuyos valores sólo tienen dos opciones, “0” cuando la iluminación es muy baja, que se representa con el color negro, y “1”, para las regiones de la escena donde hay suficiente luz y que se representa con el blanco.

Figura 2. Imagen RGB vs Imagen Binaria.



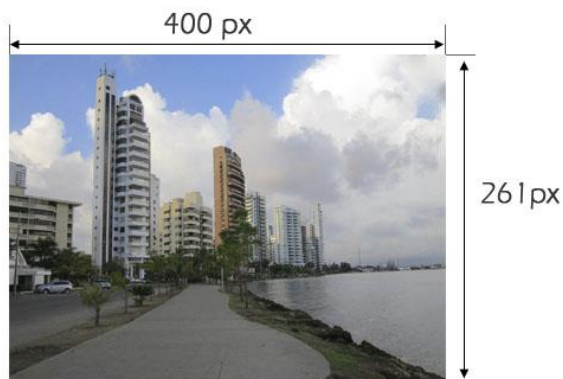
La cuantificación a 8 bits, permite observar más detalles de una imagen, ya que la intensidad de luz se puede cuantificar en 256 valores diferentes. Es la más usada en el procesamiento de imagen pues si se tiene una imagen de color gris, el ojo humano sólo alcanzaría a identificar poco más de 60 niveles de este color. De allí el nombre de escala de grises, asignando el valor “0” en decimal ó “00000000” en binario al color negro verdadero, ó a la ausencia de luz, y el valor de “256” en decimal, ó “11111111” al elemento o región de la escena con la mayor iluminación, o al color más brillante, (color blanco). Lo que significa que entre más claro sea el pixel de la imagen, más alto será el valor de su intensidad al codificarla.

2.2.1.3 Representación del color. Está comprobado que el ojo humano detecta toda la gama de colores visibles con sensores de sólo tres colores, rojo, verde y azul. Por esta razón, para representar una imagen a color, se debe tener no solamente una matriz de intensidad de luz, sino tres, cada una con la intensidad

lumínica de intensidades color rojo, verde y azul, este sistema es conocido como RGB, de las siglas de estos colores en inglés (Red, Green, Blue).

Cabe anotar que una imagen a color de n pixeles de ancho, y m de alto, contiene tres veces más información que una imagen de las mismas dimensiones en escala de gris, y por consiguiente tarda tantas veces más el procesamiento de la misma.

Figura 3. Dimensiones de las imágenes.



Dimension = 400 x 261

$D = 104400 \text{ px} = 0.1 \text{ Megapíxeles}$

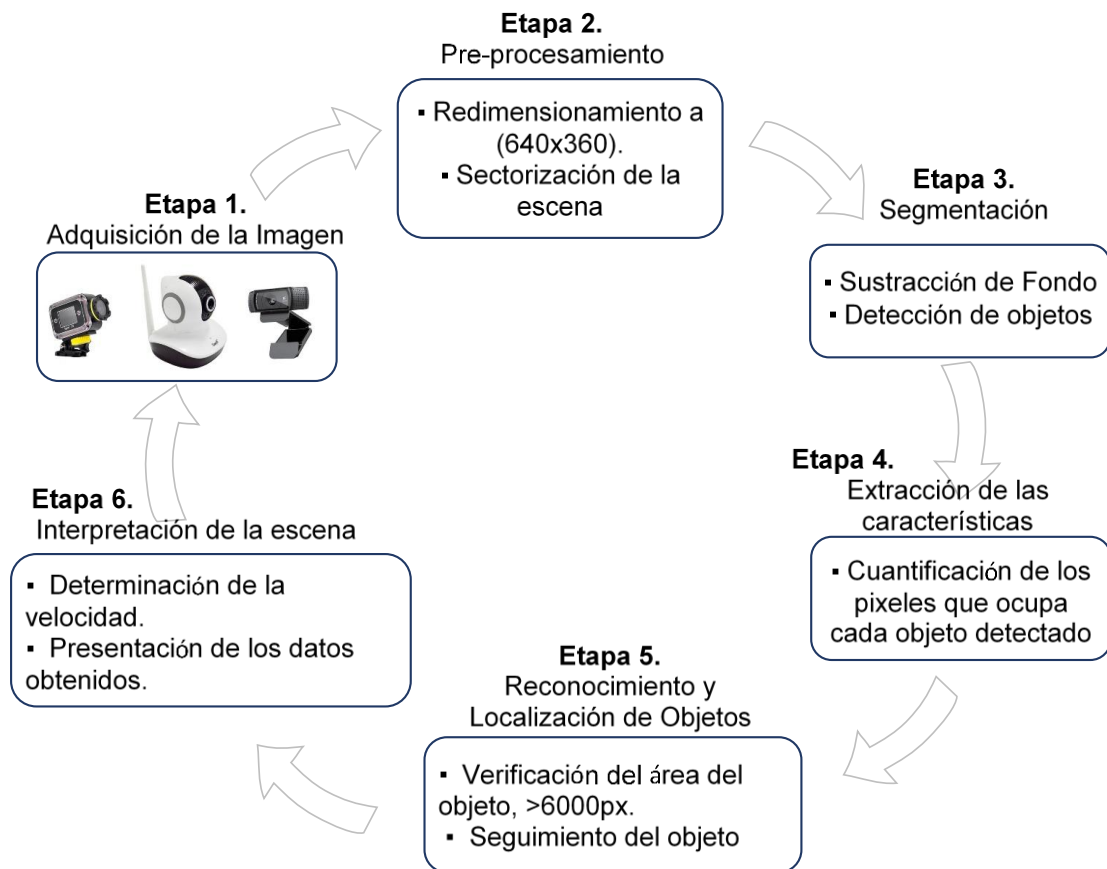
2.2.2 Etapas de un proceso de visión artificial⁷. La visión artificial, para lograr la interpretación de una escena de forma exitosa, y como se mencionó anteriormente, siendo una sub-área de la inteligencia artificial, requiere afinar tanto hardware como software, y para ello se han establecido unas etapas que llevan al sistema por este proceso para conseguir la interpretación, lo más cercana posible a la realidad. Son las siguientes las etapas a seguir en este proyecto.

Primero, se requiere adquirir la imagen, y para ello es necesario tanto un elemento sensor, como de un sistema de codificación adecuada (cuantificación), donde el primer elemento, se encarga de recoger las diferentes frecuencias del espectro de luz, en una matriz bidimensional con elementos fotosensibles, y que normalmente usan tres filtros, para capturar las tonalidades rojo, azul y verde, casi simultáneamente para posteriormente ser codificados, y así, poder procesarse digitalmente.

⁷ PLATERO, C. (2009). Apuntes de visión artificial. Cap. 1. Dpto. Electrónica, Automática e Informática Industrial. Pág 27.

Luego, la imagen pasa por una etapa de pre-procesamiento, donde se corrigen algunos imperfectos, que provienen tanto de la iluminación de la escena como del hardware de adquisición, tales como ruido, contraste, brillo, entre otras; Es aquí, donde también se recorta la imagen, para trabajar dentro de una región específica. Todo en procura de perfeccionar la imagen para obtener resultados más confiables al procesarla. En este punto empieza a tener vital importancia el tiempo del proceso, pues entre mayor sea el algoritmo de pre-procesamiento, mayor será el tiempo en la ejecución del programa. Por lo anterior es de vital importancia escoger muy bien el hardware, tanto de captura como del procesamiento, especialmente cuando cada aplicación tiene sus requerimientos específicos para la captura de la imagen.

Figura 4. Etapas del proyecto.



2.2.2.1 Adquisición de imagen.

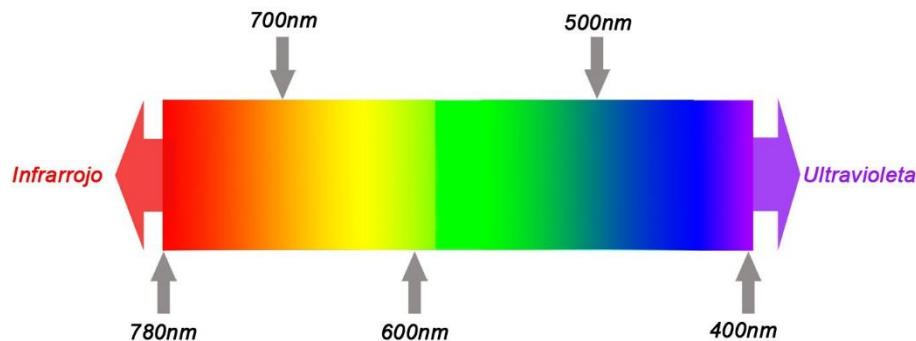
- **Conceptos básicos de óptica**⁸. El concepto luz se define como una onda electromagnética compuesta por fotones (partículas energizadas), cuya frecuencia y energía determinan la longitud de onda de un color que puede ser percibido por el ojo humano.

La óptica tradicionalmente es la rama de la física que estudia los fenómenos relacionados con la luz. De la manera como es emitida por los cuerpos luminosos, la forma en la que se propaga a través de los medios transparentes y de la forma en que es absorbida por otros cuerpos. Así mismo, la óptica ha sido definida históricamente, como “la ciencia de la luz”. En la actualidad, se define como la parte de la física que estudia los fenómenos relacionados con la propagación de la radiación electromagnética en un rango determinado del espectro, denominado rango de frecuencias ópticas.

Este rango, habitualmente descrito en la escala equivalente de la longitud de onda, incluye tres franjas (o, genéricamente, espectros): el ultravioleta (desde 10nm hasta 400 nm), el visible (desde 400 nm hasta 780 nm) y el infrarrojo (desde 780 nm hasta 1mm). Dentro del espectro visible, el conjunto de frecuencias a las que es sensible el sistema visual humano, se denominan colores. (Ver Figura 4).

- **Conceptos básicos de video.** Dimensiones. Es el tamaño del video (ancho x alto) expresado en píxeles cuando se visualiza al 100%, sin agrandar ni reducir, tal como se miden las imágenes. Hoy día se han alcanzado resoluciones muy altas denominadas de alta definición ó High Definition, de 1280x720(Hd) y 1920x1080 (Full Hd), muy usadas en televisión digital.

Figura 5. Espectro electromagnético visible.



⁸ PLATERO, C. (2009). Apuntes de visión artificial. Pág 13.

Para el desarrollo de este proyecto se realizaron tomas de video con una resolución de 1280 x 720p, para poder contar con la mayor información posible, esperando un óptimo resultado del procesamiento, como el espacio necesario para almacenarlo y el tiempo empleado para ello.

Cuadros por segundo. Un video resulta de la exposición de imágenes o cuadros uno detrás de otro. Es entonces la cantidad de cuadros que se muestran en un segundo (fps) durante su reproducción. Comercialmente se ha estandarizado el uso de 30fps en las cámaras de video, aunque existen también con velocidades de 15fps para aplicaciones no muy exigentes (webcams o vigilancia), pero cuando la aplicación requiere captar movimientos muy rápidos, es necesario aumentar este valor, alcanzando velocidades comerciales de hasta un millón de cuadros por segundo⁹, e incluso un trillón de fps, para aplicaciones especializadas en el estudio de la luz¹⁰, por ejemplo.

Considerando velocidades cercanas a los 100Km/h en el tráfico a estudiar, se determina usar una cámara de 60fps, y no perder estos movimientos tan rápidos, aunque se decide grabar un segundo video con una velocidad de 30fps, para someterlo a pruebas para determinar si esta velocidad es suficiente para las velocidades registradas.

Proporción o relación de aspecto. Es la relación entre la anchura y altura de un video. Por ejemplo la relación 3:5, hace referencia a una imagen de 3 tantos de ancho, por 5 de alto. Cuando se reproduce un video se suele mantener por defecto esta proporción para evitar deformación de las imágenes. Por este motivo cuando se elige la visualización de una imagen en una pantalla con diferente proporción, aparecen franjas negras rellenando por arriba y debajo de la imagen hasta ocupar todo el espacio de la pantalla. (Figura 6).

Las proporciones existentes varían en su relación de 3:5, (imágenes altas) hasta 16:9, relación muy utilizada en la televisión y cine, permitiendo un cubrimiento más envolvente.

Para interpretar la relación de aspecto, teniendo la resolución de la imagen o video, basta con dividir el ancho sobre el alto, para obtener cualquiera de estas relaciones (3/5, 2/3, 4/3 y 16/9).

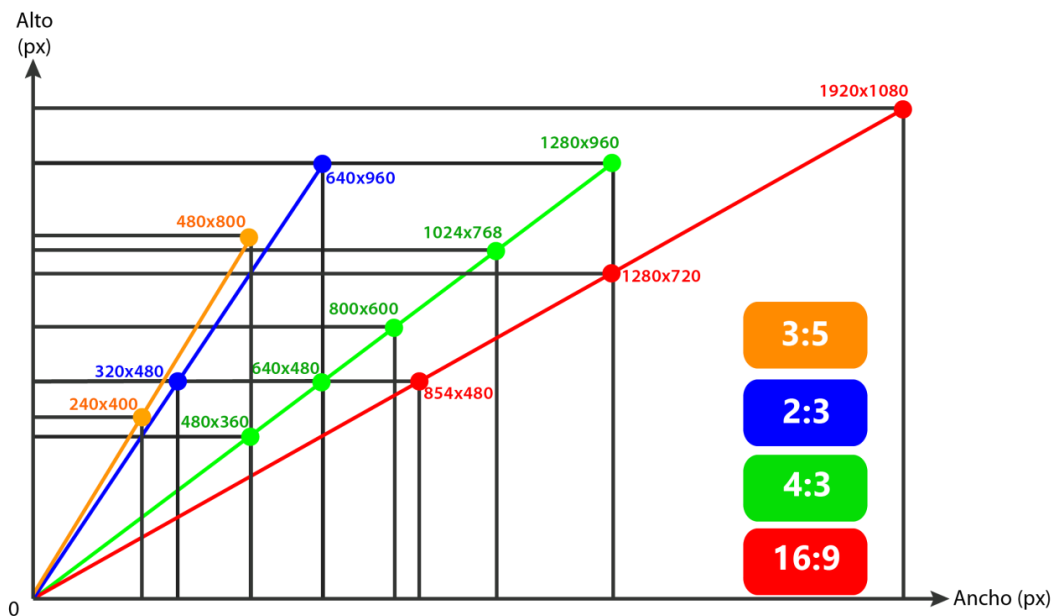
Cámaras utilizadas. Al escoger las cámaras fueron determinante factores como resolución, velocidad y nivelación de intensidad (para grabaciones nocturnas). Fue así que se adquirió, la **Logitech WebCam c920**, con una velocidad de 30fps a

⁹ <http://www.visionresearch.com/Products/High-Speed-Cameras/>

¹⁰ <http://newsoffice.mit.edu/2011/trillion-fps-camera-1213>

1280x1080p, y corrección automática de iluminación escasa (modo nocturno); Por otro lado, una **Sport Cam DV**, con velocidad de 60fps a la misma resolución y lente de 170°, capturando un área mayor que la anterior. Y por último la **Easy-N H3 V10D** con una velocidad de 25fps a la misma resolución hd, pero con sistema infrarrojo de visión nocturna. De esta forma se cumplen los requisitos, tanto en velocidad, como resolución y modo nocturno (Figura 7).

Figura 6. Relaciones de Aspecto.



2.2.2.2 Pre-procesamiento. Generalidades. Una vez que la imagen ha sido adquirida por el elemento sensor, es muy fácil que sufran algunas degradaciones en la información, debida a factores de la escena o a defectos en el mismo elemento sensor, e incluso en el canal de transmisión, como ruido o interferencia, perdiendo parte de la información.

Por esta razón es necesario utilizar algunos mecanismos para contrarrestar estos efectos, pero esta vez con ayuda del software. Dentro de los mecanismos utilizados, para tal fin se encuentran: Mejoramiento de brillo, contraste, reducción de ruido, entre otros. El inconveniente es que cualquiera de estos mecanismos, extiende el número de líneas de código, aumentando a su vez el tiempo de ejecución del programa, por lo que se recomienda evitar al máximo este tipo de correcciones por software, haciéndolo preferiblemente desde el hardware directamente.

Figura 7. Comparativo de cámaras utilizadas.



Fuentes: [http://listado.mercadolibre.com.co/camara-60fps#D\[A:camara-60fps\]](http://listado.mercadolibre.com.co/camara-60fps#D[A:camara-60fps])
<http://www.logitech.com/es-roam/product/hd-pro-webcam-c920>
<http://www.aliexpress.com/item-img/EasyN-H3-V10D-IP-camera-1-0-Megapixel-Built-in-IR-Cut-support-Mobile-view-Support/911287431.html#>

Para este proyecto, se resolvió en esta etapa realizar solamente operaciones que se redujeran el procesamiento de las imágenes, teniendo en cuenta que el video no presentaba deficiencias significativas en cuanto a brillo, nitidez ni ruido, pero sí se utilizó para efectos de redimensionamiento de la imagen y sectorización por regiones.

- **Desarrollo.** El primer paso dado en el pre-procesamiento se dió sobre la resolución, determinando la mínima resolución necesaria para realizar el procesamiento de reconocimiento de objetos, sin que se vea afectado el resultado. De esta forma se estableció que la resolución mínima, sin afectar la relación de aspecto fue de 640x360p, por lo que las primeras líneas del algoritmo se dedican a este propósito.

A continuación, se sectoriza la escena, recortando todo aquel espacio innecesario en el procesamiento, como la parte superior y lateral izquierdo y derecho del espacio ocupado por el carril escogido. Al final la región de interés se reduce al espacio marcado en color rojo de la Figura 8. Con esta sectorización, se pretende ahorrar tiempo de procesamiento innecesario, al no existir información de interés en las regiones excluidas.

Figura 8. Sectorización de la escena.

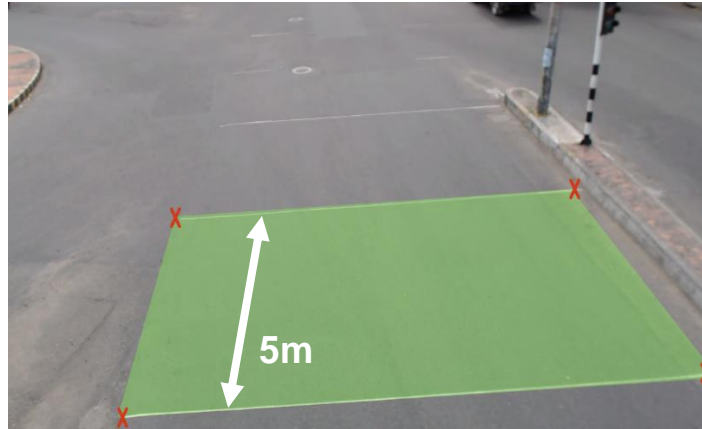


Marcada el área principal de trabajo, sólo resta referenciar los 5 metros que servirán para calcular la velocidad de cada vehículo, determinando el tiempo que emplea cada uno en recorrer esta distancia. Para conseguirlo, fue necesario utilizar marcas físicas sobre la vía, que consisten en líneas transversales a la dirección de los vehículos, separadas cinco (5) metros, una de la otra. De esta forma es mucho más fácil la referenciación en el software, pues para el caso de este proyecto, sólo basta con dar clic en cada una de las cuatro esquinas que forman el recuadro, entre dos líneas de referencia y los andenes que delimitan la calzada de la Avenida 26. (Figura 9).

De esta forma, el algoritmo realizará el conteo, exclusivamente a los vehículos que atraviesen esta región de interés, desde arriba hasta abajo, creando la primera condición en el programa, para validar que el objeto detectado es entonces un vehículo.

2.2.2.3 Segmentación. En este punto es muy importante determinar primero, cuál es el fondo de la escena (imagen estática), ya que una vez establecido, cualquier cambio en los valores de este fondo, será automáticamente identificado como objeto, al cual se le hará el posterior análisis, para determinar si el objeto cumple con las características propias de un vehículo.

Figura 9. Área de referencia.



El método para sustracción de fondo, busca reconocer dentro de la escena los pixeles que no varían de un cuadro al siguiente, o que al menos su variación es muy leve, comportamiento exclusivo del fondo de la escena, de esta forma se crea entonces una imagen de referencia sobre la que se reconoce fácilmente cualquier cambio, como objeto sobre ella. Para tal fin, se utiliza el algoritmo MOG (Mixture of Gaussian distribution) que detecta cuales son los pixeles, cuya variación en el tiempo es casi nula. El MOG entrega un “lienzo negro” sobre el cual será mucho más fácil detectar cualquier cambio brusco dentro de las intensidades de los pixeles de la imagen (Figura 10).

Figura 10. Sustracción de Fondo.



Para evitar que las alteraciones en el fondo sin ser detectadas, tales como cambios en las sombras, reflejos, o incluso la variación de la iluminación, se programa una renovación del MOG cada 200 cuadros.

Una vez identificados los píxeles que conforman el fondo de la escena, se procede a la identificación de cada objeto detectado, usando por lo menos, dos métodos diferentes para detección de bordes, ya que realmente no existe un método que sea 100% efectivo.

El primer procedimiento utilizado es una máscara de enfoque que destaca los bordes de la escena, complementándose con el algoritmo de Canny, que gracias a su robustez, ofrece una definición de bordes muy efectiva¹¹.

- **Máscara de enfoque.** Este proceso consiste, primero, en aplicar una técnica de erosión, que aumenta las regiones con altos valores de intensidad, reduciendo así, las áreas más oscuras de la escena, coincidiendo éstas últimas con el fondo extraído anteriormente.

Figura 11. Dilatación de la escena.



El siguiente paso, consiste en realizar una comparación, entre la imagen dilatada y la original, aclarando, que es la segunda, la que tiene los valores de intensidad más bajos, tomándola como referencia para este caso, dividiendo cada píxel entre el valor del mismo píxel en la imagen dilatada, presentándose sólo dos casos:

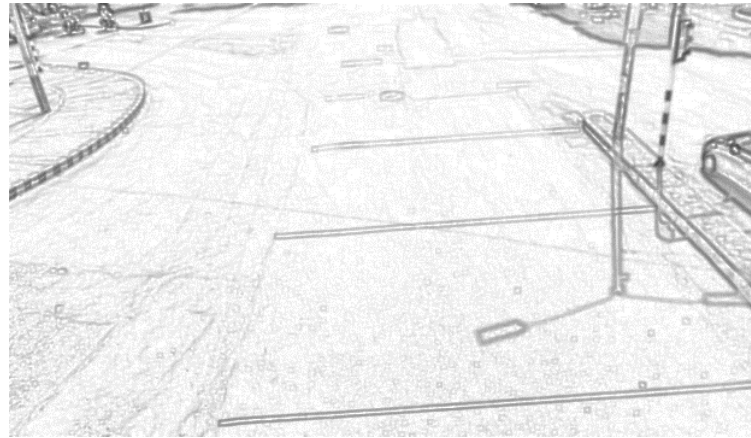
Si el píxel real es igual al píxel dilatado, se evalúa en $1. \frac{\text{img real}}{\text{img dilatada}} = 1$

Si el píxel real es menor al píxel dilatado sería menor que $1 \frac{\text{img real}}{\text{img dilatada}} < 1$

El resultado entonces se multiplica por el máximo valor (255), obteniendo entonces la detección de bordes por medio de la máscara de enfoque, por la que se conocen cuáles son los objetos dentro de la escena (Figura 12).

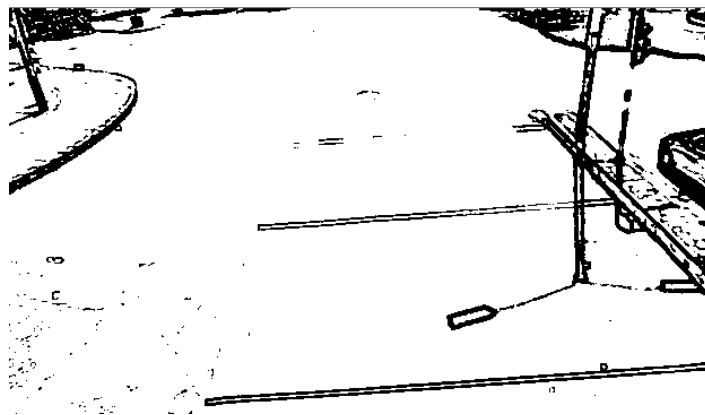
¹¹ MAINI, RAMAN & DR. AGGARWAL, HIMANSHU. Study and Comparison of Various Image Edge Detection Techniques

Figura 12. Máscara de enfoque.



Hasta el momento se pueden visualizar los objetos claramente, pero dentro del procesamiento, como el fondo se simbolizará con el color negro, es necesario que ésta imagen, cumpla con este requerimiento, para que no genere conflicto con el resto del algoritmo. Por esta razón, sumada a la necesidad de trabajar con imágenes binarias, primero se realiza una normalización, que asigne a los píxeles mayores a 230 un valor de "1", y a los restantes "0" (Figura 13), para proceder a invertir estos valores y obtener entonces la imagen binaria, cuyo valor de "0", corresponda al mismo fondo detectado anteriormente. (Figura 14).

Figura 13. Normalización máscara de Enfoque



Ya en estas condiciones, es muy fácil detectar cualquier objeto dentro de la misma. Pero como se mencionó anteriormente, para aumentar la eficiencia del algoritmo en la detección de objetos hace falta aplicar otro método, de forma paralela para garantizar que todos los objetos (bordes) sean censados.

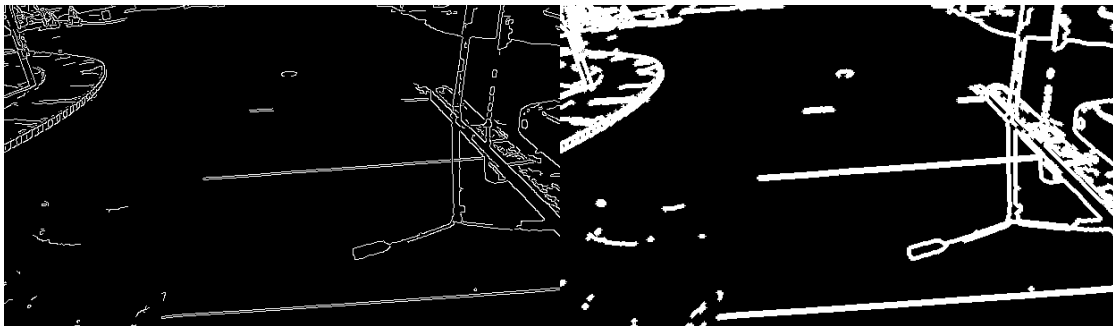
Figura 14. Máscara de Enfoque invertida



- **Algoritmo de Canny**¹². Este método toma las variaciones o cambios bruscos en la intensidad de los píxeles, derivando sus valores a lo largo de la imagen, suprimiendo luego, cualquier valor que no sea el máximo, dejando solamente aquellos píxeles que arrojaron un valor muy alto en su derivada (cambios muy bruscos de intensidad) (Ver •a).

Para mejorar la visualización y para efectos de identificación de objetos, se realiza una dilatación al filtro de Canny (Fig 15b).

Figura 15. Filtro de Canny



15 (a) Filtro de Canny

15(b) Canny dilatado

2.2.2.4 Extracción de las características. Una vez se tengan los objetos reconocidos en las áreas definidas previamente, es importante extraer las características de cada objeto reconocido, para ello se determinó primero el área

¹² MAINI, RAMAN & DR. AGGARWAL, HIMANSHU. Study and Comparison of Various Image Edge Detection Techniques

que ocupa cada objeto que aparece en escena, cuantificando el número de píxeles que ocupa.

Como primera medida, se observó que los vehículos livianos (automóviles) cómo mínimo deben ocupar un área cercana a los 6000 píxeles.

Las motos, ocupan un espacio menor, alrededor de 2000 píxeles como mínimo. Con las bicicletas cabe resaltar, que la diferencia no es muy significativa. Para el caso de los peatones, se encontró que el paso de dos peatones a la vez, puede ocupar la misma área de un vehículo, por lo que se tuvo que discriminar cualquier objeto que no cruce primero por la parte superior de la región de interés.

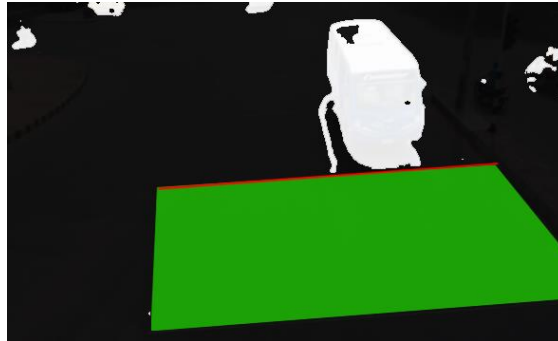
Figura 16. Área en píxeles de una moto



2.2.2.5 Reconocimiento y Localización de Objetos. Para reconocer que tipo de objeto son los que se han detectado, se realiza un filtro de áreas, que primero determina la cantidad de píxeles que ocupa el objeto, y segundo, realiza una comparación con valores ya establecidos, y que según su valor se clasifican en peatones, motocicletas y vehículos, siendo los dos últimos, los de interés en el mecanismo de conteo vehicular.

Para que el conteo se inicie, como se mencionó anteriormente es necesario asegurar primero que el objeto pase sobre la línea de referencia ubicada en el límite superior de la región de interés, y una vez se cumple esta condición, se realiza la clasificación, que busca determinar qué tipo de objeto es el que se encuentra en la escena, ya sea un automóvil, una moto, una bolsa que arrastra el viento, etc.

Figura 17. Condición inicial para el conteo.



Una vez el objeto, cruza esta línea, se realiza un conteo de los pixeles que lo conforman, y se hace su respectiva comparación, con los valores establecidos para cada tipo de vehículo, que de ser positivo el resultado, se procede entonces con el etiquetado del objeto que de ahora en adelante se considerará como un vehículo.

Para el etiquetado de cada vehículo, se asigna un número consecutivo, en orden de aparición, junto con el número del cuadro en el que ingresa a la región de interés.

Lo siguiente, es hacer el respectivo seguimiento de cada automóvil detectado hasta que salga de la región de estudio, con el objetivo primero de no perderlo de vista, evitando que cuente dos veces el mismo objeto al cambio del fotograma y segundo, asegurar que el objeto se desplaza en la dirección esperada. Toda vez que aunque el algoritmo detecta el objeto y reconoce que es un vehículo, lo hace cada vez por imagen, más no reconoce la secuencia como tal. Para tal fin se utiliza un algoritmo especializado en seguimiento de objetos en el tiempo (secuencia de fotogramas) llamado flujo óptico u Optical Flow escrito en su idioma original y cuyo funcionamiento se describe a continuación.

- **Flujo Óptico:** Éste algoritmo ofrece dos opciones de seguimiento, para flujo denso y otro para flujo escaso, siendo la última, la que se ajusta a los requerimientos de este proyecto, ya que normalmente el seguimiento se hará como máximo a dos o tres objetos a la vez, además el coste computacional de esta opción es mucho menor que la primera, ya que sólo analizaría una porción (variable, pero normalmente menor que 50%) de la región de interés.

Dentro de los métodos que realizan seguimiento para flujo escaso, es el método de Lukas-Kanade, el más utilizado¹³. El cual se basa en información local de una

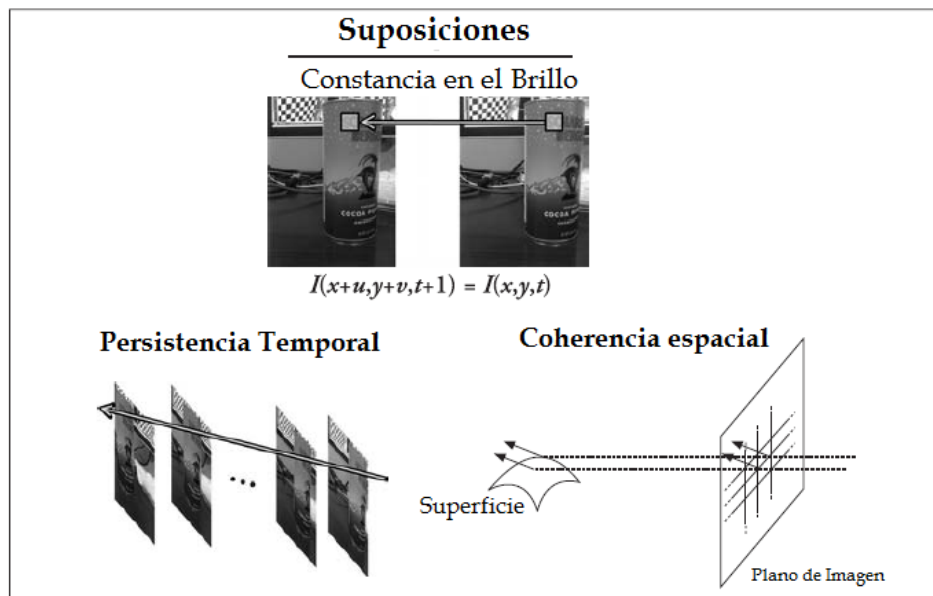
¹³ Bradski, Gary y Kaehler, Adrian. Learning OpenCV, 1ª Edición. 2008

pequeña ventana que se crea alrededor de un punto de referencia (punto más bajo del vehículo cuando hace contacto con la línea de referencia) y parte de tres suposiciones en todo movimiento de un objeto en una secuencia de imágenes: constancia en el brillo, persistencia temporal y coherencia espacial.

La **constancia en el brillo** es la persistencia en el valor de su intensidad, sin importar la variación de su ubicación (x,y) en el tiempo $(t + 1)$; La **Persistencia Temporal**, asume que los movimientos de una pequeña superficie dentro de la imagen cambian lentamente; por último, la **coherencia espacial**, supone, una vecindad de puntos con movimientos similares permitiendo una fácil proyección de los mismos en las escenas siguientes (Ver Figura 18).

De esta forma se garantiza un seguimiento óptimo, que inicia escogiendo un pixel de referencia (Punto más bajo del vehículo cuando ingresa a la región de interés), se almacena toda su información (ubicación, intensidad), como también la de su vecindad, así pues en la siguiente imagen, inicia la búsqueda de las intensidades almacenadas, desde la misma posición, hacia afuera, para que una vez se hayan encontrado, el algoritmo almacene su nueva posición, determinando la dirección en la que se ha desplazado. Con este dato se proyecta una posible ubicación en los cuadros siguientes, evitando que realice búsquedas en direcciones diferentes, ahorrando así tiempo valioso de procesamiento.

Figura 18. Suposiciones para el Método Lucas-Kanade



Fuente: Bradski, Gary and Kaehler, Adrian. Learning OpenCV, 1ª Edición. 2008. Pág. 24

2.2.2.6 Interpretación de la Escena. Es el punto más importante y básicamente el último que comprende el algoritmo de visión artificial, restando sólo el cálculo de la velocidad, manejo de datos, impresión en pantalla y salvado de los registros en un documento plano de texto.

La información procesada se presenta de tal forma que el usuario pueda comprender lo que está ocurriendo en la escena en tiempo real, y consiste en visualizar el resultado del procesamiento, en la misma imagen capturada, enmarcando los vehículos detectados con un recuadro color azul mientras que recorren el área de interés, e imprimiendo además el consecutivo de etiquetado del último vehículo que salió de la región de interés.

En este punto, finaliza el procesamiento de conteo, pero aún no se ha calculado la velocidad. Para conseguirlo, sólo es necesario conocer cuántos cuadros tarda cada vehículo que ha ingresado en la región de censado, en salir de ella, pues el sensor, como se mencionó en la primer parte de este capítulo, funciona con un estándar de 30 ó 60 cuadros/segundo, permitiendo establecer la segunda constante para este cálculo, que sumado a la constante de la distancia en la región de interés ($x=5m$) y las operaciones de conversión de unidades, se establece la ecuación a utilizar para establecer la velocidad real de cada vehículo.

A continuación, se presenta matemáticamente el desarrollo de esta ecuación.

$$Vel = \frac{x}{t} ; \text{ donde } x = 5m$$

t = tiempo que tarda el vehículo en salir de la escena

El tiempo de cada cuadro se obtiene de la velocidad de grabación de la cámara:

$$Tf = \frac{1}{vel} \quad vel = \text{velocidad sensor (fps)}$$

Si la cámara graba a 30fps:

$$Tf = \frac{1f}{30f/s} = 0.033s \text{ ó } 33,3ms$$

Si la cámara graba a 60fps:

$$Tf = \frac{1f}{60f/s} = 0.0166s \text{ ó } 16,6ms$$

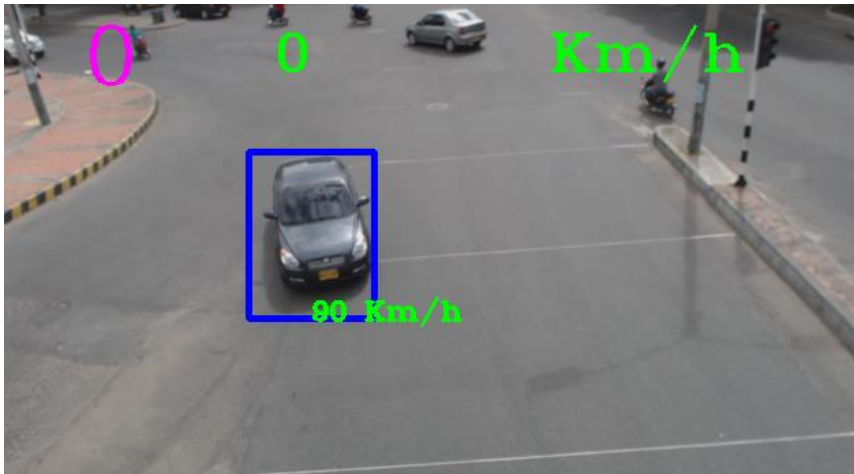
Así entonces, teniendo como constantes x con un valor de 5m y el tiempo de cada cuadro, el tiempo queda sujeto entonces al número de cuadros que tarda en salir

el vehículo de la región demarcada. Aplicando conversiones de unidades para obtener los resultados en Km/h se obtiene entonces:

$$Vel = \frac{540}{\#f} [Km/h] \quad \text{Donde } \#f = \text{No. cuadros}$$

De esta forma, cuando se obtiene la cantidad de cuadros por vehículo, se obtiene la velocidad real y se imprime en la parte superior derecha de la pantalla, en la aplicación creada para la visualización, a la izquierda de la pantalla, pero en la misma línea de la velocidad y en color magenta, el número de vehículos detectados, mientras que en color verde se muestra el número de los vehículos que además de ser detectados, cruzaron por toda la región de interés.

Figura 19. Interpretación de la escena.



2.3 CUMPLIMIENTO DE REQUERIMIENTOS

Tal como se ha propuesto en este proyecto, el objetivo fundamental, es el algoritmo encargado de realizar la interpretación de un video capturado previamente, aunque su aplicación también se puede realizar en tiempo real.

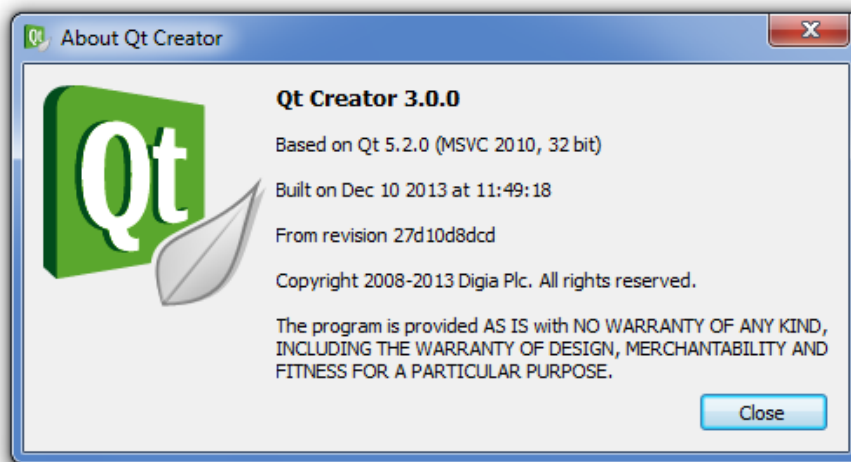
En el cumplimiento de los objetivos, el resultado se presenta de dos formas diferentes, la primera se hace de forma gráfica, en el aplicativo creado, que integra tanto el algoritmo de procesamiento como el entorno gráfico, donde se muestra el progreso del procesamiento de cada video seleccionado. Por otro lado, se presenta en un archivo de texto plano, donde se registra la hora de aparición de cada vehículo, su etiquetado consecutivo y velocidad calculada.

2.4 DESARROLLO DEL SOFTWARE

Como se mencionó anteriormente, el proyecto se desarrolló bajo las bondades, fortalezas y resultados óptimos para este tema, que entrega la librería OpenCV, creada bajo el lenguaje C y C++, que corre bajo plataforma Linux, Windows y Mac OS. Aunque actualmente se están desarrollando interfaces para Phyton, Ruby, Matlab y otros lenguajes.

Toda la programación se llevó a cabo, usando el software Qt Creator V3.0.0

Figura 20. Acerca de Software de compilación

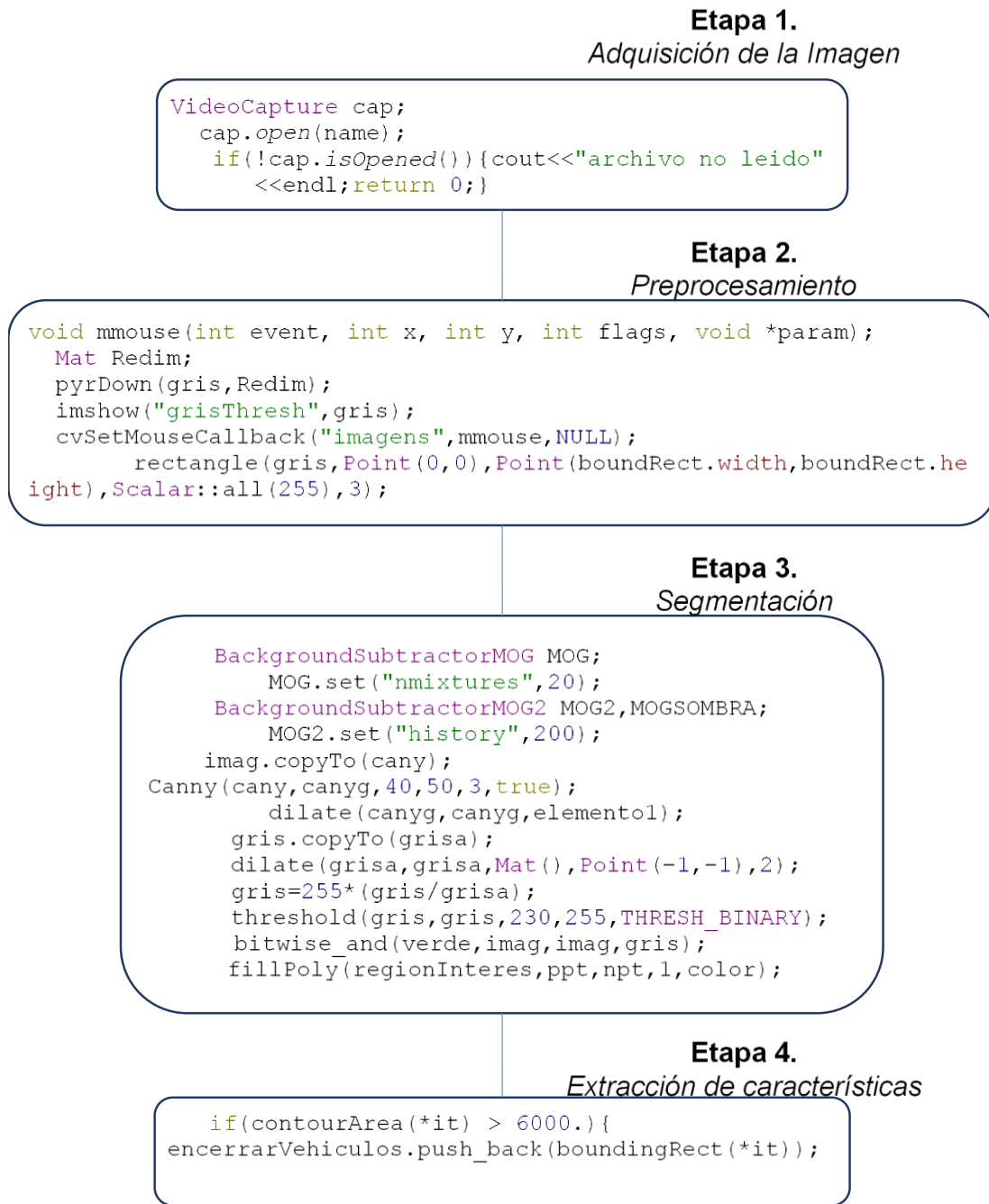


Una vez se obtuvieron los resultados esperados en el algoritmo en la detección vehicular, se implementó un aplicativo, donde se pudiera apreciar el procedimiento de identificación y conteo vehicular de forma gráfica. La idea surge por la necesidad de monitorear la funcionalidad del sistema, aunque no sea estrictamente necesaria. Pues como se mencionó al inicio del capítulo, lo más importante del proyecto, son los datos que al final arroja el algoritmo al procesar cada imagen capturada.

El Software como tal se compone de dos elementos de desarrollo, el primero consiste en el algoritmo de procesamiento de imágenes, y el segundo, el aplicativo que presenta el algoritmo dentro de un entorno gráfico, donde para este caso se selecciona el video a estudiar, con la posibilidad de parametrizar varias regiones de interés para el caso en que se requieran de distancias diferentes.

2.4.1 Algoritmo de Procesamiento de Imágenes. Éste algoritmo tal como se mencionó anteriormente, se desarrolla bajo la librería OpenCV, y a continuación se presenta las principales funciones que se utilizaron en este proyecto.

Figura 21 Diagrama de Flujo del Algoritmo



Etapa 5.
Reconocimiento de objetos

```
calcOpticalFlowPyrLK(prevGray, blob,  
points[0], points[1], status, err, winSize, 3,  
termcrit, 0, 0.001);
```

Etapa 6.
Interpretación de la Escena

```
stringstream intStr, tint;  
int velocidad=18/(0.0333*tiempo.at(i));  
intStr<<velocidad;  
tint<<tiempo.at(i);  
putText(ima, tint.str()+"..", points[1][i], CV_FONT_HERSHEY_COMPLEX_SMALL, 1,  
Scalar(0, 255, 0), 2);  
int velocidad=18/(0.0333*frams);  
intStrp<<velocidad;  
putText(ima, intStrp.str()+"  
Km/h", Point(400, 50), CV_FONT_HERSHEY_COMPLEX_SMALL, 2, Scalar(0, 255, 0), 2);
```

2.4.1.1 Adquisición de Imágenes. Se utiliza primero la función de lectura del video, VideoCapture(), que captura las imágenes ya sean de un archivo, o de una cámara en tiempo real:

2.4.1.2 Preprocesamiento. Aquí se redimensiona con la función pyrDown(); que conserva la relación de aspecto del video original y reduce a la mitad, las dimensiones originales. Además, se inscribe la región de interés por medio de la lectura del mouse, cuando se hace clic en cada uno de los cuatro puntos que la conforman, con la función cvSetMouseCallback():

2.4.1.3 Segmentación. Empieza con la detección del fondo de la escena, utilizando el método gaussiano con la función BackgroundSubtractorMOG(). Continuando con la detección de objetos, por medio de sus bordes por dilatación, dilate(), y el algoritmo de Canny, canny(). Claro está que aunque son las funciones principales, se complementan con otras operaciones como umbralización threshold(), además se utilizan varias funciones punto a punto, o pixel a pixel entre imágenes con la función bitwise(), como invertir los valores en una imagen binaria, usando bitwise_not() y operaciones and y or con bitwise_and() y bitwise_or() respectivamente. Como también se realizan operaciones de sustracción con la función subtract(). otras ayudas en la detección efectiva de objetos, son utilizadas, como findContours(), que busca contornos en imágenes binarias y fillPoly(), que rellena las áreas delimitadas por los contornos:

2.4.1.4 Extracción de características. Antes de reconocer el objeto es necesario un condicional que detecta que el objeto ha ingresado a la región de interés, para ubicar y almacenar el punto que está más abajo del objeto y calcular el área que ocupa con la función `contourArea()`.

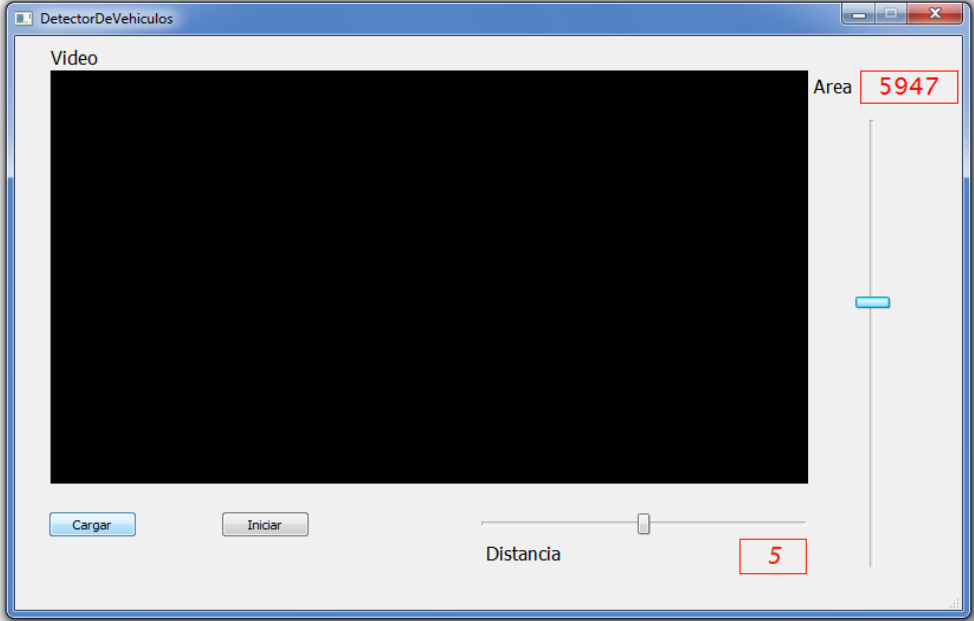
2.4.1.4 Reconocimiento y localización de objetos. Ya con el área del objeto cuantificada, se compara con el valor de referencia mínimo escogido por el usuario, y en el caso que sea mayor se asigna una etiqueta temporal, considerándose dicho objeto como un vehículo, por tanto se enmarca en color azul para efectos de una mejor visualización. Para no perder el objeto referenciado en los cuadros siguientes se aplica la función `calcOpticalFlowPyrLK()`, que sigue cada movimiento tanto del punto de referencia como de su vecindad por toda la región de interés, hasta que el objeto llega al límite inferior. Paralelamente al movimiento, se lleva la cuenta de los cuadros empleados en este trayecto:

2.4.1.5 Interpretación de la escena. Para cuando el vehículo ha salido de la región de interés, se etiqueta nuevamente, pero esta vez definitivamente y siguiendo la secuencia (conteo vehicular). Con el número de cuadros empleados por los vehículos para realizar el desplazamiento se procede a calcular entonces la velocidad del mismo, e imprimirlo tanto en pantalla (gráficamente) como registrarlo en un archivo plano de texto:

2.4.2 Desarrollo de Aplicativo. Como parte esencial del aplicativo, primero se determina y ubica el espacio para la visualización de las imágenes del video, con dimensiones de 640 x 320px.

Luego se escogien dos controles básicos, “Cargar”, que al activarse, abre una ventana titulada “abrir video” para que el usuario navegue por el equipo en busca del que desea analizar. El segundo es “Iniciar”, que al momento de ser activado inicia el algoritmo de procesamiento de video, cambiando el texto del botón a “Detener”, a la espera de ser activado si el usuario desea pausar el programa. Se agrega además dos controles deslizables, el primero determina la distancia del área de interés que permite seleccionar desde 1 hasta 10m, que por supuesto, deben coincidir con la distancia real en los puntos de referencias en la vía. El otro control permite escoger el valor mínimo de referencia que determina el área en pixeles para ser considerados en la detección ya sea, vehículos, motos e incluso peatones. La idea de utilizar este segundo control y no usar un valor fijo, se debe a que cada cámara usada en este proyecto tiene características diferentes, en particular, el ángulo de visión, entregando cada una, imágenes con tamaños (en pixeles) diferentes, haciendo que el área de los objetos también varíe.

Figura 22. Presentación de aplicativo



3. ANÁLISIS DE RESULTADOS

Para presentar los resultados obtenidos, fue necesario abordar algunas situaciones que en algunos casos no fueron previstas, y en otros casos la afectación fue mayor que la esperada. A continuación se presenta en detalle tanto la problemática encontrada, como la solución más efectiva.

El análisis del resultado obtenido, luego de correr el algoritmo de detección y conteo vehicular, se centra en la evaluación de cerca de 3 horas de grabación tanto diurnas como nocturnas, haciendo énfasis en las imágenes donde se presenta mayor actividad (de día). Por esta razón se observa detalladamente 78 minutos de grabación diurna, haciendo el conteo manual de cada vehículo (824), se compara con los datos obtenidos con el algoritmo e incluso, se revisa la interpretación que hace el algoritmo con cada vehículo detectado y con cada situación particular acontecida. Encontrando que la efectividad del algoritmo alcanza un 97,57% en el conteo vehicular. Dentro del 2,43% de interpretaciones erróneas que se encontraron, se destacan tres causas: Cambios bruscos en la iluminación de la escena (5%, 1 vehículo), Conteo de vehículos que circulan por carriles adjuntos (15%, 3 vehículos), y presencia de puntos ciegos (80%, 16 vehículos).

Estos datos, comparados con otras tesis de identificación de vehículos¹⁴ con muy buenos resultados y que utilizan otros métodos (redes neuronales) cuya efectividad alcanza un 96%, indican que el método escogido para el desarrollo de este algoritmo es lo suficientemente robusto para este trabajo.

A continuación se presenta en detalle del análisis de los resultados obtenidos:

3.1 CONTEO VEHICULAR

Como primera medida se analizan los resultados obtenidos sólo en la detección y conteo vehicular, parte fundamental del objeto de este proyecto y base para el cálculo de velocidad, que también se analizará más adelante.

3.1.1 Imágenes Diurnas. Al ser tomadas como referencia para el desarrollo del algoritmo, las imágenes diurnas son las que presentan mejores resultados al ser interpretadas, sin embargo de los errores en la interpretación del software, y las causas de estas situaciones luego de analizar detenidamente 78 minutos de grabación en dos videos, se contaron de forma manual 824 vehículos y aunque el

¹⁴ Gonzalez Z. Andrés Felipe. Medición de Flujo vehicular basado en visión artificial. julio de 2006, Quindío Colombia

conteo automático arrojó la misma cifra, se presentaron 20 situaciones donde la interpretación del algoritmo fue errónea, que para este caso, coincidieron las situaciones que contaron vehículos que realmente no existían con las que omitieron vehículos reales.

De los 20 casos de interpretación errónea del algoritmo, se estudiaron uno a uno con el objetivo de identificar la causa real que provoca el error, encontrando que 16 de estas situaciones se generaron por la presencia de puntos ciegos en la escena, otros 3 por invasión del espacio censado por vehículos que se desplazan por otros carriles, y 1 caso por cambios bruscos en la iluminación de la escena, particularmente cuando vehículos muy altos con carrocerías muy brillantes pasan debajo del sensor.

Figura 23. Causas de errores en imágenes diurnas.

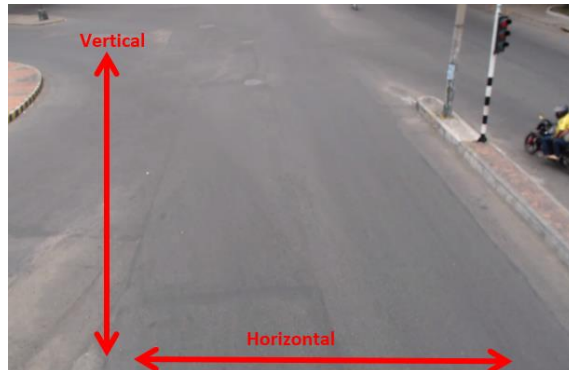


A continuación, se presenta en detalle estas causas y el tratamiento adecuado para minimizar estos efectos negativos en la interpretación del algoritmo, haciéndolo mucho más confiable.

- **Puntos Ciegos.** Esta situación, aunque al inicio era esperada, fue en el desarrollo donde se logró comprobar la gran afectación en la interpretación del algoritmo, pues consiste en aquellas áreas donde los vehículos de alguna forma no pueden ser detectados, debido a la superposición de otros objetos, incluso otros vehículos, alterando drásticamente las características propias de cada objeto. Este fenómeno se presenta cada vez que un objeto está muy cerca de un vehículo, viendo la cámara que ambos parecieran hacer contacto, ya sea cuando se ubican uno detrás del otro, o uno al lado del otro, que referenciados en la imagen capturada, se nombran Puntos ciegos verticales y horizontales, respectivamente.

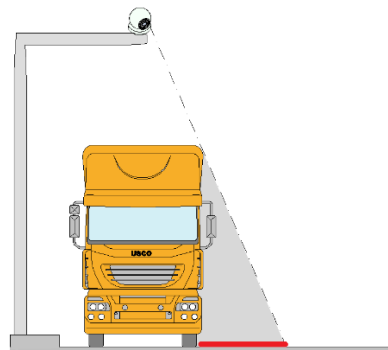
Para efectos de ejemplo, se toma como referencia el peor de los casos, un vehículo pesado, con una altura de 4 metros, cuyos puntos ciegos son los más grandes.

Figura 24. Formas de puntos ciegos.



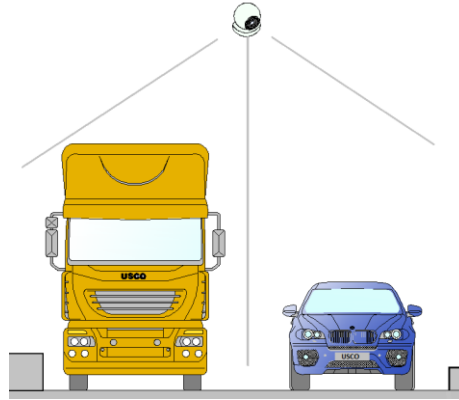
Puntos ciegos Horizontales. Son aquellas áreas que se ocultan a los costados de los vehículos, normalmente los de gran tamaño, que se sobreponen a otros vehículos o parte de ellos. Cuando se presenta esta situación, y a pesar que los vehículos quedan registrados en el video, el algoritmo interpretará el conjunto como un solo vehículo (Figura 25).

Figura 25. Punto ciego Horizontal.



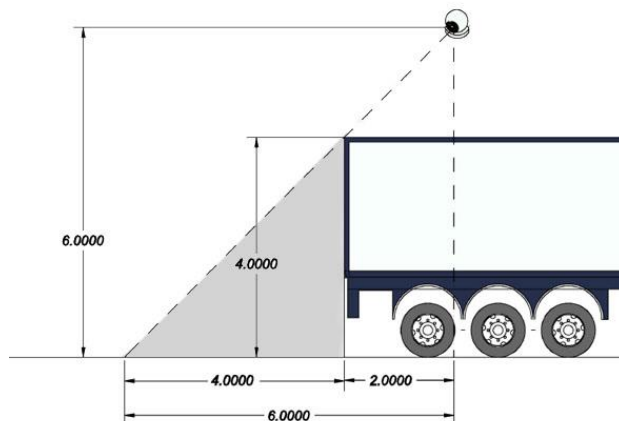
La solución de este problema, está en la ubicación del sensor, el cual debe situarse sobre el punto medio de la calzada en estudio, en este caso que la conforman sólo dos carriles, es necesario que se ubique sobre la línea que los divide (Figura 26).

Figura 26. Eliminación punto ciego horizontal.



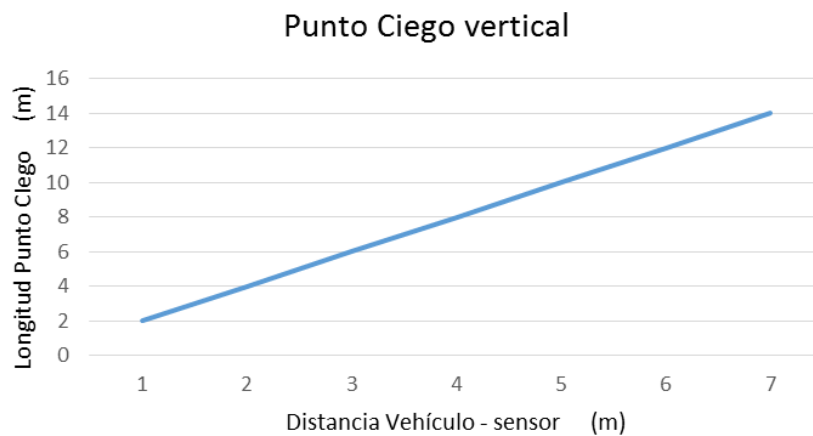
Puntos ciegos Verticales. Son aquellos espacios que se ocultan detrás de cada vehículo, y que afectan la interpretación que hace el algoritmo, cuando otro vehículo se ubica en este espacio, situación que empeora, a medida que el primer vehículo se ubica a mayor distancia de la cámara. Aquí entran en juego tres factores determinantes, elevación del sensor, inclinación con la que funciona, y la distancia de la región de interés con respecto a la componente vertical del sensor. Por esta razón es necesario revisar la geometría de esta situación en la escena (Figura 27), de donde se debe asumir primero que la altura mínima sea de 6 metros, parámetro que se debe manejar como una constante, en cuanto al ángulo del sensor con respecto a la componente horizontal es completamente dependiente del tercer factor, pero se debe aclarar que cuando el ángulo se hace muy grande, reduce el espacio registrado por la cámara, mientras que al acercarse a la componente horizontal, aumenta drásticamente los espacios de los puntos ciegos.

Figura 27. Geometría del punto ciego vertical.



Para determinar con exactitud, la distancia que oculta un vehículo, como el de referencia justo detrás de él, y a medida que se acerca al sensor, se calcula las diferentes posibilidades, se tabula y presenta de forma gráfica, comparando la distancia entre el borde trasero del vehículo y la componente vertical del sensor, con la longitud del punto ciego creado justo detrás del automotor.

Figura 28. Relación Distancia vehículo-sensor vs Longitud punto ciego.



Basados en material fílmico y el análisis realizado anteriormente, la línea que delimita la región de interés por la parte superior (punto más lejano del sensor), no debería superar los 10 metros, con los mismos 5 metros de región de interés. Para el caso que se requiera una región más grande (10m, por ejemplo), se hace necesario elevar el sensor por lo menos 2m más, para contrarrestar los efectos de este fenómeno.

Figura 29. Ejemplo de punto ciego.

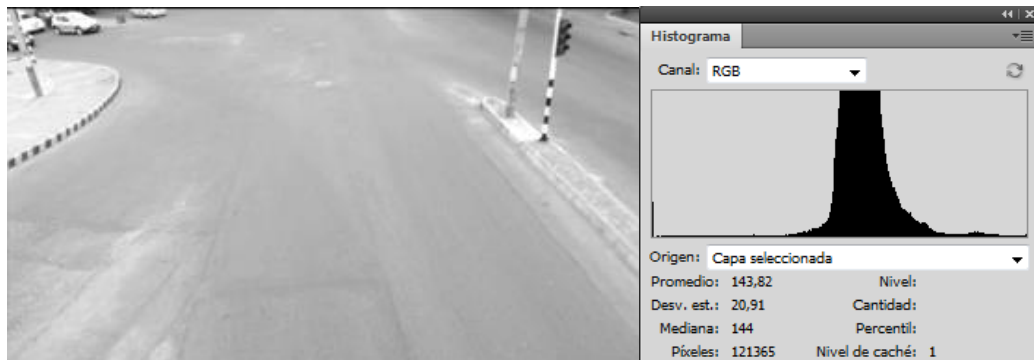


De la ubicación del sensor se deduce que entre más alto esté la cámara, mejor será el cubrimiento de la vía, sin embargo los costos de elevar este dispositivo aumentan, toda vez que la estructura debe soportar y resistir los embates de los fuertes vientos, requiriendo no sólo más material sino también materiales mucho más resistentes.

3.1.2 Imágenes Nocturnas. Uno de los objetivos de este proyecto es automatizar tanto el conteo como la medición de la velocidad, y esto significa una funcionalidad las 24 horas del día, lo que significa que debe ser tan efectivo de día como de noche. Pero desafortunadamente las condiciones óptimas en el ambiente de la escena en el día se vuelven adversas en la medida que va llegando la noche.

En este punto la iluminación juega un papel determinante en la efectividad de los resultados del algoritmo. Pues al caer la noche desencadena una serie de inconvenientes que dificultan tanto el reconocimiento de objetos, como su seguimiento y por consiguiente la determinación de la velocidad. Para ilustrar este problema se acude al histograma, gráfico que registra la distribución de pixeles según su intensidad en un rango de 0 a 255, y que presenta en su gráfica una distribución que normalmente, describe una forma gaussiana para cada región cuyos pixeles tienen similares intensidad (objetos), destacándose los pixeles que conforman el fondo de la escena, en la forma gaussiana más prominente.

Figura 30. Histograma escena diurna.



Del histograma de la Figura 30, se observa una distribución gaussiana, correspondiente a los pixeles cuyas intensidades coinciden con color del pavimento, cuyo promedio es 143, y aunque el fondo de la escena comprende otros objetos, como andenes, aceras, postes, etc, es la vía, la que ocupa la gran mayoría de la escena, y donde prácticamente se desarrollará toda la acción. Gracias a la iluminación natural de la escena, la mayoría de los pixeles tienen una intensidad muy cercana a la mitad de la escala (128), valor ideal que permite que la distancia con los puntos máximos y mínimos sea la misma, permitiendo detectar

objetos más oscuros con igual facilidad que los más claros que el fondo. Para el caso que la iluminación natural disminuye, el fondo se hace más oscuro, desplazando esta distribución gaussiana hacia la izquierda, reduciendo la brecha de niveles disponibles para objetos más oscuros, dificultando su reconocimiento al tener cambios muy sutiles, mientras que al otro lado de la escala este rango aumenta, facilitando la detección para los objetos más claros.

Figura 31. Histograma escena nocturna (Gris).



Tal como se aprecia en la Figura 31. Histograma escena nocturna (Gris)., el promedio de la distribución gaussiana correspondiente al fondo es 49, muy por debajo del primer cuarto del rango completo, quedando bastante espacio a la derecha, lugar para las imágenes más brillantes, apreciándose con más facilidad, mientras a la derecha los objetos más oscuros, estarán más cerca de ser simplemente objetos color negro, desencadenando una serie de problemas, en especial con la carrocerías de los vehículos de colores oscuros.

Solución 1: Detección de focos. En el análisis del problema mencionado anteriormente, se requirió encontrar otra forma de reconocimiento de vehículos, y más si se tiene un amplio espacio en el histograma para los pixeles más brillantes, se decide detectar los focos de los vehículos, aprovechando la facilidad de reconocerlos por sus altas intensidades, por consiguiente se decide desarrollar un algoritmo adicional, que detecte faros de vehículos, los empareje, y determine si cumplen con las características de iluminación de todo vehículo, que se resumen en dos focos con tamaños iguales, con una separación entre ellos relativamente constante (20 – 40px).

El reto no era fácil pues la idea de agregar más código al algoritmo, aumenta el tiempo de procesamiento, por lo que se empieza a trabajar en la idea de adicionar un bloque para reconocimientos de faros, conservando el segmento de sustracción de fondo.

El nuevo bloque realiza una umbralización de la escena con niveles por encima de 200, se ejecuta un filtro de área para eliminar los pequeños reflejos u otras fuentes de luz, dejando sólo las regiones cuyo tamaño coincidan con los faros. Restando solamente determinar la distancia entre ellos, validando si coincide con la distancia de referencia (20 a 40px).

Figura 32. Secuencia detección faros.



Evaluando la efectividad del bloque de código, se encontraron un par de situaciones complejas. Vehículos con faros apagados y otros desalineados, haciendo casi imposible la detección de estos vehículos, pues para el primer caso, el algoritmo nunca encontrará su pareja (Figura 33), mientras que para el caso siguiente, los faros alineados por encima del estándar, normalmente, aumentan el área brillante, haciendo al menos un faro más grande, y en el caso que apuntaban por debajo del estándar, hacia el suelo, se formaba un área mucho más grande que el mismo faro, incluso del tamaño promedio de un vehículo.

Figura 33. Ilustración faro apagado.



Figura 34. Ilustración faros desalineados.



Es aquí donde se determina, basados en observación, que el índice de discrepancias en la alineación de los faros, y funcionamiento óptimo de ellos, se acerca al 10%, cifra que aumentaría en el mismo tanto el error en la interpretación del algoritmo.

Otro factor, aún más importante que se suma los dos anteriores, obedece a la imposibilidad de poder determinar en momento se realiza el cambio de los bloques para el día y la noche, toda vez que el anochecer o amanecer ocurre en una transición que se puede extender hasta dos horas, imposibilitando la determinación del momento en que empieza la noche o el día. Esto sumado a las diferentes horas en que los conductores encienden o apagan las luces de los vehículos, hacen realmente imposible determinar la un punto exacto para el cambio del bloque.

Resultados: Es aquí cuando se confirma que la propuesta no es efectiva en la solución del problema, por lo que se debe replantear la problemática nuevamente, toda vez que las interpretaciones erróneas del algoritmo se disparan drásticamente.

De regreso a los resultados que entrega el algoritmo para imágenes diurnas, pero esta vez mientras se corren dos videos nocturnos. Se determina entonces que en los 104 minutos de video, de los 1229 vehículos que cruzaron por la región de interés, el algoritmo registró 1254, sin embargo y al igual que en el caso de procesamiento de imágenes diurnas, las interpretaciones erróneas son mucho mayores que la diferencia de estas dos cifras, pues se encontró un total de 265 errores, de las cuales se destacan tres causas: Deficiencias en la iluminación de la escena, donde la luz de los faros vehiculares, crean regiones con bordes muy acentuados, que terminan engañando al sistema haciéndolo creer que son vehículos (239 casos), que equivalen al 90% del total de errores; Intrusión en la región de estudio de vehículos que circulan por otros carriles (17 casos), 6,42%; y

por último la presencia de puntos ciegos en la escena, al igual que en caso de imágenes diurnas (9 casos), 3,40%.

Figura 35. Causas de errores en imágenes nocturnas

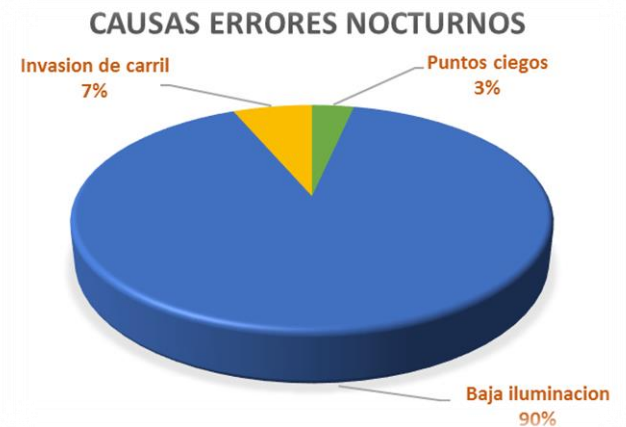


Figura 36. Problemas de Iluminación.



Solución 2. Aunque al principio se pretendía trabajar en la solución a través de programación, se determina que la escena no cumple con los requerimientos mínimos para tal procesamiento, por lo que se plantea una solución que se debe implementar en la escena, aunque vale la pena aclarar que para esta situación, el sensor que mejor rendimiento ofrece para la captura de estas imágenes es la WebCam Logitech C920, con su función de corrección automática de iluminación y ganancia.

Se encuentra que la mejor solución es en efecto el mejoramiento de la iluminación de la escena, a través de fuentes de luz como reflectores halógenos o led, o

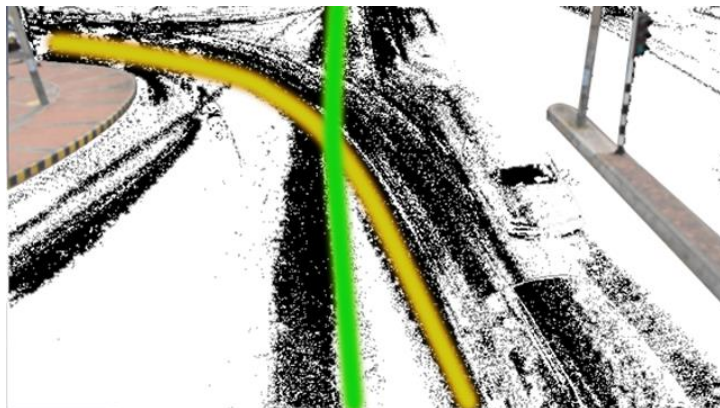
cualquier otro dispositivo de iluminación. De esta forma el fondo del escenario aumenta su intensidad, acercándose a la intensidad de los píxeles iluminados por los faros vehiculares, desvaneciendo los bordes, minimizando las interpretaciones erróneas por esta situación.

3.2 VELOCIDAD

En el cálculo de este parámetro se encontraron dos inconvenientes, el primero radica en las diferentes trayectorias que siguen los vehículos dentro de la escena y la segunda al factor velocidad del sensor.

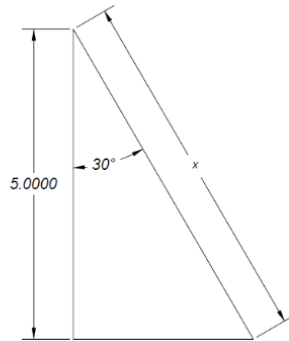
3.2.1 Exactitud por Trayectoria. Durante la evaluación de los resultados en el cálculo de la velocidad, se encontró que algunos vehículos que ingresan al carril en estudio desde la carrera séptima, se desplazan con una trayectoria diferente a la trayectoria que siguen los que vienen por la misma avenida 26 (), aumentando para el primer caso la trayectoria de desplazamiento en el mismo tramo de 5 m, al aparecer un ángulo de inclinación.

Figura 37. Trayectorias de desplazamiento.



Dándole una revisión a la geometría de esta situación, se determina la variación de los parámetros en escena y el resultado que arroja esta variación en la interpretación que hace el algoritmo (Figura 38). Los detalles se presentan a continuación.

Figura 38. Geometría de las trayectorias.



$$X = 5m / \cos 30^\circ$$

$$X = 5.7825m.$$

Esta variación en la distancia real de recorrido, entrega en el cálculo de la velocidad, un valor inferior de la real, ya que como se mencionó anteriormente, el algoritmo calcula la velocidad de todos los vehículos, asumiendo que el recorrido es de 5m.

Para este ejemplo se asume un tiempo de recorrido de 0.54s (tiempo promedio de desplazamiento entre las 11:28 y 12:15hrs del día de grabación)

$$Vel = \frac{x}{t}; \quad x = 5m ; t = 0.54s$$

Velocidad según el algoritmo, para un vehículo con trayectoria inclinada 30°:

$$Vel = \frac{5m}{0.54s} = 9.26 \text{ m/s} \times \frac{3600s}{1h} \times \frac{1km}{1000m} = 33.3 \text{ km/h}$$

Velocidad real:

$$Vel = \frac{5.7825m}{0.54s} = 10.71 \text{ m/s} \times \frac{3600s}{1h} \times \frac{1km}{1000m} = 38.5 \text{ km/h}$$

De tal forma se encuentra que el error en la velocidad calculada es el siguiente:

Error absoluto:

$$e_r = |33.3 - 38.5| = 5.2$$

Error relativo:

$$e_r = \frac{V_m - V_r}{V_r} = \frac{33.3 - 38.5}{38.5} = 0.13$$

Luego de haber analizado estos cálculos, se intenta agregar al algoritmo un segmento de código que calcule la trayectoria que sigue cada vehículo, determinando el ángulo y a su vez, la distancia real recorrida, y así el cálculo de la velocidad sea más confiable. Al analizar la causa básica de esta situación, surge nuevamente la escena, pues al momento de escoger la estructura a instalar los sensores de este proyecto no se tuvo más alternativa, aun cuando se puede apreciar fácilmente que la región de interés se encuentra prácticamente en el radio de giro de los vehículos que vienen por la carrera séptima, por lo que la solución está en asegurar que la región de interés se encuentre dentro de una sola vía, y no en un punto de intersección, entre otra palabras, se necesitaría que la cámara estuviera ubicada unos 20m, detrás del lugar donde se ubicó inicialmente.

De esta forma, se corrige automáticamente el error en el cálculo de la velocidad por esta causa.

3.2.2 Por Sensor. Ésta situación se ve reflejada en el momento en que el vehículo sale de la región de interés, pero el sensor no lo detecta instantáneamente, sino un momento después (Figura 39), aumentando el tiempo que realmente tarda el vehículo en terminar el desplazamiento de censado.

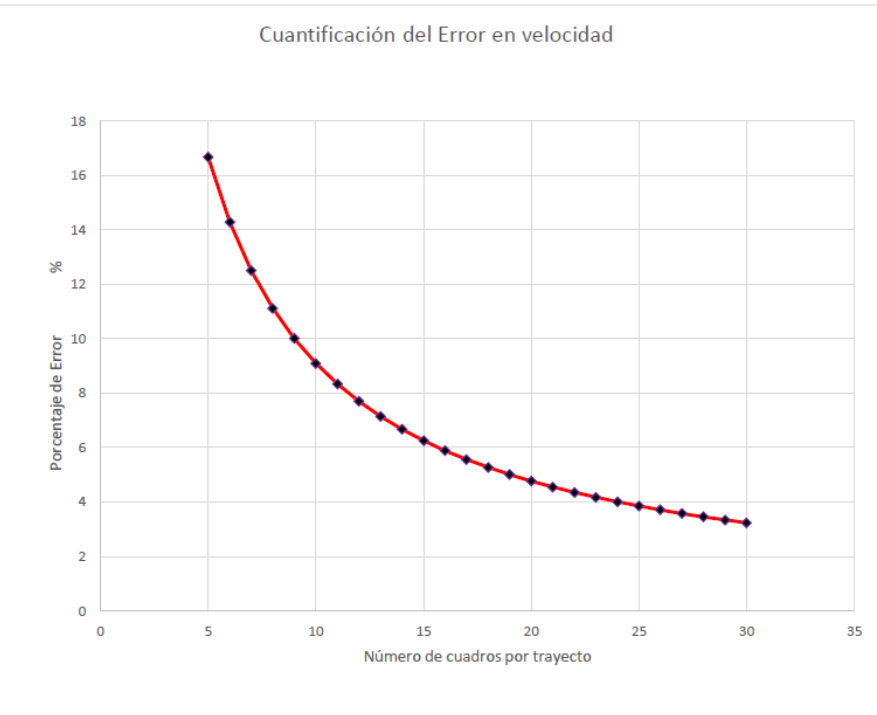
Figura 39. Registro de fin de desplazamiento.



En la cuantificación de esta diferencia, cabe resaltar que dicha situación se presenta en casi todos los casos, sin embargo al presentarse también en el momento de entrada del vehículo, es casi improbable, que esta diferencia sea mayor a 1 fotograma, y que al conocer el tiempo de este fotograma (0.033s), sin embargo la afectación al valor de la velocidad calculado depende de la velocidad real con la que se desplaza cada vehículo, que al final se traduce en la cantidad de cuadros o fotogramas empleados en ese desplazamiento, ya que por ejemplo una trayectoria que se realiza en 6 cuadros, el algoritmo calculará una velocidad de 90km/h, sin embargo, si ocurre esta situación, el error sería de 14% ($\pm 12,6$ Km/h) mientras que una trayectoria de 20 cuadros, la velocidad registrada sería de 27Km/h, pero el error por un fotograma de más, apenas llega al 4.8% ($\pm 1,2$ Km/h). El detalle se presenta en la Figura 40.

Figura 40. Error en la velocidad según los fotogramas

# Cuadros	Velocidad Km/h	Error (%)
2	270,0	-
3	180,0	-
4	135,0	-
5	108,0	16,7
6	90,0	14,3
7	77,1	12,5
8	67,5	11,1
9	60,0	10,0
10	54,0	9,1
11	49,1	8,3
12	45,0	7,7
13	41,5	7,1
14	38,6	6,7
15	36,0	6,3
16	33,8	5,9
17	31,8	5,6
18	30,0	5,3
19	28,4	5,0
20	27,0	4,8
21	25,7	4,5
22	24,5	4,3
23	23,5	4,2
24	22,5	4,0
25	21,6	3,8
26	20,8	3,7
27	20,0	3,6
28	19,3	3,4
29	18,6	3,3
30	18,0	3,2



3.3 SOMBRAS

Dentro de los factores determinantes en el éxito de este código, obedece a la identificación y eliminación de las sombras, especialmente cuando hay presencia de nubes, haciendo la iluminación muy difusa, dificultando la detección de los bordes de la sombra.

Este factor fue prácticamente el que requirió más tiempo, pues consistió en encontrar el valor mínimo de cada sombra la cual hacía parte del vehículo, en el levantamiento de las superficies gaussianas, tanto para extracción del fondo, como en la identificación de objetos. Considerando que una sombra con bajos niveles de intensidad, aumenta considerablemente el área de los vehículos, uniéndolos y creando espacios que desaparecen rápidamente afectando el resultado del programa; mientras que por otro lado, una sombra con valores mínimos elevados, altera la forma resultante del objeto, confundiendo la carrocería de los carros con el fondo de la escena eliminando partes de los automóviles, dificultando su reconocimiento.

Figura 41. Sombras.



4. CONCLUSIONES

Este proyecto desarrolló una herramienta para el conteo de vehículos y la medición de la velocidad de los mismos, de una forma automatizada, permitiendo conocer por primera vez, y en tiempo real, el comportamiento de la movilidad al menos en un punto de la ciudad de Neiva, con cifras reales, y tomadas de forma automática por un algoritmo creado combinando varias técnicas de visión artificial, procesando cada imagen en un tiempo lo suficientemente menor al tiempo que tarda en aparecer el siguiente cuadro, permitiendo que este algoritmo sea utilizado en aplicaciones en tiempo real.

Otra característica importante de este sistema, es la utilización de la menor cantidad de funciones posibles, alcanzando al final un programa con relativamente, pocas líneas de programación, pero con resultados muy satisfactorios las 24 horas del día, que en caso de seguir las recomendaciones sugeridas, se puede asegurar mínimo un 95% de efectividad promedio. Aunque en las pruebas realizadas a los videos capturados con anterioridad se alcanzó una efectividad del 97,57% para las imágenes diurnas, es en la noche donde cae al 78,4%. Sin embargo el promedio de 88% es un buen indicador de confiabilidad del sistema, teniendo en cuenta las dificultades con las que se instaló el sensor.

De todas las situaciones presentadas que afectaban el resultado del algoritmo, son los puntos ciegos los más significativos, que afortunadamente, en esta vía el tránsito de vehículos pesados (de mayor altura), no es muy común, favoreciendo los resultados de este proyecto, pero que deja una conclusión determinante: Es determinante una altura mayor del sensor para el estudio de tráfico pesado (carreteras intermunicipales) en comparación con el tráfico liviano estudiado para este caso (vías urbanas).

Con este proyecto se aporta no sólo al campo de la ingeniería de tránsito y transporte, sino que además se cumple con una cuota de responsabilidad social, pues aunque tiene alcances en el ámbito privado, su aplicación está proyectada para ser una herramienta utilizada en el mejoramiento de la calidad de vida de los habitantes de Neiva, más aún cuando la ciudad está sumergiéndose en grandes proyectos con altísimas inversiones económicas para el mejoramiento de la movilidad en la capital huilense.

Del material fílmico analizado en todo el conjunto, se destacó la importancia del cruce escogido para el desarrollo de este proyecto (Av 26 con Carrera 7ª) pues se evidenció que es efectivamente un punto de convergencia de los cuatro puntos cardinales de la ciudad, donde por ejemplo una gran cantidad de los vehículos que se desplazan por la avenida 26 de Sur a Norte, toman el desvío hacia el oriente de

la ciudad (Avenida Calle 27), contrario a lo que se esperaba. De igual forma, la mayoría de los automotores que arriban a este cruce por la avenida 7^a, terminan tomando la Av. 26 hacia el Norte. Mientras que los que llegan de este punto cardinal por la misma avenida, terminan girando a la izquierda por la 7^a hacia el centro y sur de la Capital huilense.

Por último vale la pena destacar el alcance de este proyecto, no sólo en la cuantificación de la movilidad en la ciudad, para efectos estadísticos, sino en aplicaciones más específicas como foto-multas, cobros automáticos en peajes, control de circulación para locaciones privadas, hasta el desarrollo de sistemas automatizados para semaforización, parqueaderos, entre otros.

5. RECOMENDACIONES

De la experiencia y los resultados obtenidos en este proyecto no cabe la menor duda que al tener en cuenta las siguientes recomendaciones, se puede alcanzar, e incluso superar un 95% de efectividad. Estas recomendaciones se hacen siguiendo 3 ítems: Sensor, estructura y condiciones ambientales.

Del sensor, resultan tres recomendaciones claves, la primera obedece a la referencia del sensor, ya que la WebCam Logitech C920 fue la que mejor imágenes capturó, se recomienda entonces el uso de este sensor, pues además cuenta con el mejor sistema para contrarrestar la escasa iluminación en la escena; la segunda es la velocidad, siendo 30 fps, la velocidad recomendada; terminando con la resolución ideal para aplicaciones de este tipo, que contuviera suficiente información de la escena idóneo y un bajo coste computacional en el proceso, es pues la resolución sugerida 640x480.

De la estructura se recomienda que sea lo suficientemente robusta y estable, sobre todo en la parte superior, con una altura no menor a 6 metros sobre el nivel de la vía, y con el soporte para el sensor justo en el punto medio de la calzada que se va a estudiar. Además, la estructura deberá estar separada 20m como mínimo de la esquina más cercana, y en lo posible lejos de salidas de estacionamientos o establecimientos comerciales, destacando que debe contar con una acometida eléctrica y de ser posible un sistema de respaldo, para los casos en que el sistema eléctrico falle.

En cuanto a los factores ambientales, se recomienda escoger el lugar de estudio cuya superficie de la vía sea lo más pareja posible, libre de huecos, y en lo posible que no existan sombras; Para las horas de la noche, es necesario el uso de dos reflectores tipo led de 150W, con un sistema de encendido automático, una vez la iluminación natural descienda.

Dentro de las situaciones que se deben evitar, se recomienda asegurar que los vehículos no estacionen en el área de estudio ni a 50 metros detrás o frente al sensor. Por otro lado, a este proyecto se puede agregar el modelado matemático del flujo vehicular, y hacer la interfaz con los controladores del sistema de semaforización, logrando que sea 100% automatizado

BIBLIOGRAFIA

BALLESTEROS V, G. (2004). Verificación de vehículos mediante técnicas de visión artificial. Universidad autónoma de Madrid. España.

BRADSKI, GARY Y KAEHLER, ADRIAN. (2008). Learning OpenCV. 1ª Edición. O'Reilly Media Inc. EE. UU.

CAL, RAFAEL Y REYES, M. Y CÁRDENAS, JAMES. (1994). Ingeniería de tránsito, Fundamentos y aplicaciones. 7ª Ed. Alfaomega. México.

FENALCO, ANDI Y COMITÉ AUTOMOTOR COLOMBIANO. Informe del Sector Automotor a diciembre de 2014. Bogotá D.C, 2015. Recurso disponible en internet: <http://www.andi.com.co/Documents/INFORME%20DEL%20SECTOR%20AUTOMOTOR%20A%20DICIEMBRE%202014%20cierre%20a%C3%B1o.pdf>

GÓMEZ G, E. (2006). Guía básica de conceptos de óptica geométrica. Universidad de Sevilla. España.

GONZÁLEZ, A; MARTÍNEZ, F; PERNÍA, V; ALBA, F; CASTEJÓN, M; ORDIERES, J & VERGARA, E. (2006). Técnicas y algoritmos básicos de visión artificial. Universidad de la Rioja. España.

PLATERO, C. (2009). Apuntes de visión artificial. Cap. 1. Dpto. Electrónica, Automática e Informática Industrial.

TAPIA A, J & VEIZAGA B, R. (2006). Apoyo didáctico para la enseñanza y aprendizaje de la asignatura de ingeniería de tráfico. Universidad Mayor de San Simón. Cochabamba – Bolivia.

MAINI, RAMAN & DR. AGGARWAL, HIMANSHU. Study and Comparison of Various Image Edge Detection Techniques. Penn State University.

Gonzalez Z. Andrés Felipe. Medición de Flujo vehicular basado en visión artificial. Julio de 2006, Quindío Colombia, Universidad del Quindío.