

UNIVERSIDAD NACIONAL DE INGENIERÍA
FACULTAD DE INGENIERÍA CIVIL



TESIS

**“DETECCIÓN Y RECONOCIMIENTO AUTOMATIZADO DE
SEÑALES PREVENTIVAS EN INVENTARIOS VIALES”**
PARA OBTENER EL TÍTULO PROFESIONAL DE INGENIERO CIVIL

ELABORADO POR

MARZIA VALERIA RAMOS ZÚÑIGA

ASESOR

MSc. LEONARDO FLORES GONZALEZ

LIMA- PERÚ

2021

Dedicatoria:

Esta tesis está dedicada a mis queridos padres, Juan y Luzmila, por todas sus enseñanzas, su gran esfuerzo y apoyo incondicional para lograr cumplir mis objetivos personales y profesionales.

ÍNDICE

RESUMEN	4
ABSTRACT.....	5
PRÓLOGO.....	6
LISTA DE CUADROS	7
LISTA DE TABLAS	8
LISTA DE FIGURAS.....	9
LISTA DE SÍMBOLOS Y SIGLAS	13
CAPÍTULO I. INTRODUCCIÓN	15
1.1 GENERALIDADES.....	15
1.2 PLANTEAMIENTO DE LA REALIDAD PROBLEMÁTICA	16
1.3 OBJETIVOS	16
1.3.1 Objetivo General.....	16
1.3.2 Objetivos Específicos	16
1.4 HIPÓTESIS	17
CAPÍTULO II. MARCO TEÓRICO Y CONCEPTUAL	18
2.1 SEÑALIZACIÓN VERTICAL.....	18
2.2 INVENTARIO VIALES	20
2.2.1 Inventario vial básico.....	21
2.3 VISIÓN POR COMPUTADOR.....	25
2.3.2 Pixel	25
2.3.3 Espacios de color	25
2.4 SEGMENTACIÓN DE IMAGEN	27
2.5 OPERACIONES MORFOLÓGICAS.....	29
2.5.4 Dilatación:.....	29
2.5.5 Erosión.....	29
2.5.6 Apertura	30
2.5.7 Cierre	31
2.6 DESCRIPTOR: HISTOGRAMA DE ORIENTACIÓN DE LA GRADIENTE 31	
2.7 SOPORTE DE MÁQUINAS DE VECTORES.....	33

CAPÍTULO III. ESTADO DE ARTE	40
3.1 NORMATIVA	40
3.1.1 Manual de dispositivos de control de tránsito.....	40
3.1.2 Manual de inventarios viales.....	40
3.2 TECNOLOGÍAS USADAS PARA INVENTARIOS VIALES	41
3.2.3 Métodos de detección	41
3.2.4 Métodos de Reconocimiento.....	47
3.3 BASE DE DATOS DE SEÑALES DE TRÁNSITO	51
CAPÍTULO IV. ESTRATEGIA DE SOLUCIÓN	55
4.1 METODOLOGÍA	55
4.2 INVENTARIO DE SEÑALES	57
4.2.1 Información de la vía en estudio	57
4.2.2 Campo de estudio del inventario vial	58
4.3 RECOLECCIÓN DE DATA.....	58
4.4 PROCESAMIENTO DE VIDEOS Y CREACIÓN DE BASE DE DATOS ..	63
4.5 PREPROCESADO DE DATA PARA ENTRENAMIENTO	66
4.6 ALGORITMO DE DETECCIÓN	67
4.6.3 Filtro por color.....	68
4.6.4 Filtro de forma	71
4.7 ALGORITMO DE RECONOCIMIENTO.....	75
4.7.5 Descriptor	76
4.8 ALGORITMO DISTANCIA DE OBJETOS.....	79
4.9 Mapa de señales.....	81
CAPÍTULO V. VALIDACIÓN DE LA HIPÓTESIS	82
5.1 ANÁLISIS DE RESULTADOS	82
5.1.1 Imágenes detectadas por usando el filtro de color HSV	82
5.1.2 Imágenes detectadas por usando el filtro de color normRGB	85
5.1.3 Tamaño de la imagen.....	88
5.1.4 Análisis métodos de filtros de color	88
5.1.5 Tiempo computacional	89
5.1.6 Tiempo de procesamiento.....	90
5.1.7 Porcentaje de éxito.....	90
5.1.8 Rendimiento del clasificador	91

CONCLUSIONES	95
RECOMENDACIONES.....	96
BIBLIOGRAFÍA	98
REFERENCIAS BIBLIOGRÁFICAS	102

RESUMEN

La presente investigación tiene por objetivo principal proponer herramientas que permitan automatizar el proceso del inventario vial. Para ello, se desarrolló un algoritmo para detectar y reconocer las señales de tránsito peruanas del tipo preventivas en la Av. Eduardo Habich ubicada en Lima Metropolitana en el distrito de San Martín de Porres. El tramo seleccionado para el estudio se encuentra entre la Av. Túpac Amaru y vía Panamericana Norte y posee una longitud aproximada de 1km.

La investigación está basada en el uso de algoritmos de procesamiento de imágenes. Se propuso una metodología que se divide en dos partes: la primera parte consiste en la recolección de datos en campo para la base de datos; y la segunda parte consiste en el procesamiento de la información en gabinete. Para la primera parte se realizó la recolección de 2026 imágenes de señales preventivas de vías de la ciudad de Lima. Estas imágenes fueron capturadas con cámaras de distintas resoluciones. A partir de la consolidación de la base de datos, en la segunda parte se plantea el algoritmo de detección y reconocimiento. La etapa de detección consiste en el uso de filtros de color y forma. Para ello se realizó una comparación entre dos filtros de color: HSV y RGB normalizado que permitió distinguir cual tenía mejor performance frente al algoritmo planteado. La etapa de reconocimiento consiste en el uso de herramientas de clasificación supervisada con el algoritmo llamado máquinas de vectores de soporte.

Finalmente, con el desarrollo de esta investigación se logró obtener un algoritmo que permite la detección de las señales de tránsito con un porcentaje de reconocimiento de 62.5% y una base de datos sólida que puede ser retroalimentada y dar pie a futuras investigaciones en la materia de estudio.

ABSTRACT

The main goal of this research is to propose tools which help to automate the process in the road inventory. For this reason, an algorithm was developed to detect and recognize Peruvian warning traffic signs on Eduardo Habich avenue located in Metropolitan Lima in San Martín de Porres district. The section selected for the study is between Tupac Amaru avenue and Panamericana Norte road and has 1 km approximately.

The research is based on the use of image processing algorithms. A methodology that is divided into two parts was proposed: First part is data collection and the second part proposes an algorithm. For the first part, the collection of 2026 images of Peruvian preventive type traffic signs of Lima, which were captured with different types of camera resolution. From the consolidation of the database, the second part presents the detection and recognition algorithm. The detection stage consists in the use of color filters and shape filters; so it makes a comparison between two color model HSV and RGB normalization to compare their performance. The recognition stage consists of the use of supervised classification tools with support vector machines algorithm.

Finally, the research achieved to detect and recognize preventive signals with a supervised classification algorithm with 62.5% and give an important Peruvian database, which can be used in future researches referred to this topic.

PRÓLOGO

Esta tesis presenta una herramienta de apoyo para mejorar el proceso de inventarios viales usando procesamiento de imágenes y algoritmos de clasificación supervisada, en relación con señales de tránsito preventivas peruanas.

Se ha dividido en dos etapas generales una etapa de campo para la recolección de información y creación de una base de datos propia, y segunda etapa la etapa de gabinete donde se prueba el performance del algoritmo propuesto. Finalmente se presentan resultados óptimos y recomendaciones de mejora para el algoritmo.

Se espera que esta tesis sirva para que los inventarios viales que se realizan actualmente sean de manera automatizada, en este proyecto se enfoca en un tipo de elemento de la vía sin embargo se espera que sirva como base de futuras investigaciones para realizar un inventario completo.

MSc. Leonardo Flores Gonzalez

LISTA DE CUADROS

Cuadro 1 Cronograma de actividades para la recaudación de información	59
Cuadro 2 Descripción de características técnicas de los equipos.....	60
Cuadro 3 Equivalencias de códigos	65

LISTA DE TABLAS

Tabla 1 Ficha de reconocimiento de la vía.....	23
Tabla 2 Umbrales de color.....	70
Tabla 3 Tiempo computacional etapa de Detección.....	89
Tabla 4 Número de objetos detectados.....	90
Tabla 5 Elementos detectados según tipo	90

LISTA DE FIGURAS

Figura 1 Ejemplo de señales reguladoras	18
Figura 2 Ejemplo de señales preventivas	18
Figura 3 Ejemplo de señales informativas	19
Figura 4 Diagrama del flujo explicativo de un inventario vial	21
Figura 5 Reconocimiento de campo	24
Figura 6 Modelo RGB	26
Figura 7 modelo HSV	27
Figura 8 Dilatación del elemento A, con el elemento estructurante B	29
Figura 9 Erosión	30
Figura 10 Ejemplo de eliminación de ruido de la figura	30
Figura 11 Descripción gráfica de elementos (Estructura de celda y bloques)	32
Figura 12 Histograma de los bloques de la imagen	32
Figura 13 Representación de histograma concatenado	33
Figura 14 HOG de señal peatonal	33
Figura 15 Ejemplo de hiperplano de separación	35
Figura 16 Margen de un hiperplano de separación	35
Figura 17 Detección de color	44
Figura 18 Ejemplos de detección de señales	47
Figura 19 Detección y reconocimiento de imágenes	48
Figura 20 Ejemplos de detección y reconocimiento	50
Figura 21 Tipos de señales de GTSRB	51
Figura 22 Base de datos	52
Figura 23 Base de datos de Belgium TS	52
Figura 24 Imágenes de la misma señal a diferentes distancias	53

Figura 25 Imágenes con diferentes características de luz y clima	53
Figura 26 Estructura de la base de datos LIS	54
Figura 27 Diagrama de metodología propuesta.....	56
Figura 28 Tramo en estudio, Av. Eduardo Habich	57
Figura 29 Instalación del equipo.....	61
Figura 30 Detalle de la información	61
Figura 31 Visualización en la laptop	62
Figura 32 Frame del video	62
Figura 33 Detalles del video	63
Figura 34 Herramienta para recorte	64
Figura 35 Base de datos de señales preventivas	66
Figura 36 Ejemplo aplicativo del uso de Medfilt2	67
Figura 37 Ejemplo aplicativo del uso Imgaussfilt	67
Figura 38 Diagrama Cromático CIE 1931(a).....	68
Figura 39 Diagrama Cromático CIE 1931 (b).....	69
Figura 40 Concentración de valores HSV para color amarillo	69
Figura 41 Aplicación de operaciones morfológicas.....	70
Figura 42 Ejemplos de imágenes a contraluz	71
Figura 43 Ejemplo de diagramación de la señal tipo curva la derecha.....	72
Figura 44 Características geométricas de la señal	74
Figura 45 Ejemplo de extracción de la imagen	74
Figura 46 Etapa de detección del algoritmo propuesto.....	75
Figura 47 Distancia para el borde (Distance to border)	77
Figura 48 Regresión lineal para descriptor DtBs	77
Figura 49 Máscara de la señal preventiva	78

Figura 50 HOG	78
Figura 51 Video de salida	79
Figura 52 Imagen para el análisis.....	80
Figura 53 Data de latitud.....	80
Figura 54 información de la imagen extraída	81
Figura 55 Datos de la señal de tránsito	81
Figura 56 Aplicación del sistema de detección y reconocimiento a la señal peatonal 1 (P-48) con código de proyecto 00014 (HSV)	82
Figura 57 Aplicación del sistema de detección y reconocimiento a la señal peatonal 2 (P-48) con código proyecto 00014 (HSV)	83
Figura 58 Aplicación del sistema de detección y reconocimiento a la señal intersección rotatoria (P-15) con código proyecto 00008 (HSV)	83
Figura 59 Aplicación del sistema de detección y reconocimiento a la señal resalto 1 (P-33) con código proyecto 00011 (HSV)	83
Figura 60 Aplicación del sistema de detección y reconocimiento a la señal peatonal 3 (P-48) código proyecto 00014(HSV)	84
Figura 61 Aplicación del sistema de detección y reconocimiento a la señal Proximidad a cruce escolar (P-49 A) con código proyecto 00015 (HSV).....	84
Figura 62 Aplicación del sistema de detección y reconocimiento a la señal Ciclovía y señal de resalto 2 (P-33) (HSV).....	84
Figura 63 Aplicación del sistema de detección y reconocimiento a la señal peatonal 1 (P-48) con código proyecto 00014 (normRGB).....	85
Figura 64 Aplicación del sistema de detección y reconocimiento a la señal peatonal 2 (P-48) con código proyecto 00014 (normRGB).....	85
Figura 65 Aplicación del sistema de detección y reconocimiento a la señal intersección rotatoria (P-15) con código proyecto 00008 (normRGB).....	86
Figura 66 Aplicación del sistema de detección y reconocimiento a la señal resalto 1 (P-33) con código proyecto 00011 (normRGB).....	86

Figura 67 Aplicación del sistema de detección y reconocimiento a la señal peatonal 3 (P-48) con código proyecto 00014(normRGB).....	86
Figura 68 Aplicación del sistema de detección y reconocimiento a la señal de Proximidad a cruce escolar 1 (P-49) con código proyecto 00015(normRGB)	87
Figura 69 Aplicación del sistema de detección y reconocimiento a la señal resalto (P-33) con código proyecto 00011	87
Figura 70 Aplicación del sistema de detección y reconocimiento a la señal (P-2A) con código proyecto 00017	88
Figura 71 Detección usando modelo HSV	88
Figura 72 Detección usando modelo RGB normalizado	89
Figura 73 Detalles e información del clasificador	92
Figura 74 Error del clasificador para celdas de 4x4	92
Figura 75 Detalles e información del clasificador	93
Figura 76 Error del clasificador para celdas de 8x8	93
Figura 77 Tiempo de procesamiento de 6800 frames	94

LISTA DE SÍMBOLOS Y SIGLAS

MTC	Ministerio de transporte y comunicaciones
SINAC	Sistema nacional de carreteras
UNI	Universidad Nacional de Ingeniería
IIFIC	Instituto de investigación de la facultad de ingeniería civil
GTSRB	German Traffic Sign Benchmark
ASTM	American Society of Testing Material
DVR	Digital Video Recorder o gravador de video digital
SIB	Señal de inventario básico
MUTCD	Manual uniform traffic control device
UTM	Universal Transverse Mercator
STP	Señales de tránsito preventivas
BB	Bounding box
FP	Falsos positivos
ROIs	Region of interest o regiones de interés
DtBs	Distance to borders
HOG	Histogram oriented gradient o histograma de orientación gradiente
SVM	Support vector machine o máquinas de vector soporte
CNN	Redes neuronales convolucionales
ANN	Red neuronal artificial
GPS	Sistema de Posicionamiento Global
RGB	Red-Green-Blue
HSV	Hue-Saturation-Value
MSE	Error medio cuadrático
MATLAB	Lenguaje de Programación

D	Magnitud de orientación del gradiente
θ	Orientación de orientación del gradiente
Km/s	Kilómetro por segundo
m	Metro

CAPÍTULO I. INTRODUCCIÓN

1.1 GENERALIDADES

En los últimos años el desarrollo de la tecnología ha ayudado a reducir el tiempo de ejecución de las actividades que realizamos, es así que la detección de rostros, conteo de vehículos, detección de letras y otras actividades de este tipo ha sido estudiada mediante la visión por computador (Velez,2003; Gonzalez,2008). La bibliografía revisada con referencia a identificación de señales de tránsito es dirigida a la aplicación en vehículos inteligentes o en asistentes de conductor donde se brinda al conductor conocimientos a priori sobre las señales de tránsito para que este pueda tomar sus precauciones y así evitar accidentes (Greenhalgh,2012; Maldonado, 2007). Las investigaciones del procesamiento de imágenes se han dividido por lo general en dos etapas detección y reconocimiento. La primera donde la imagen es procesada, mejorada y segmentada, es decir que a partir de una imagen dada se pueda extraer únicamente el objeto de interés para reducir el campo de estudio (Saadna, 2017). Para extraer la región de interés será necesario estudiar los modelos de color como RGB, CIElab, Ycbcr, HSV y otros, ya que la extracción basada en el color es una de las más utilizadas en esta etapa, otra herramienta es la extracción por la forma. Los colores característicos de las señales de tránsito (rojo, azul, amarillo, blanco) son la característica principal de las señales; muchas investigaciones han usado un espacio de color normalizado o la creación canales basados en la información RGB, o usos de un filtro basado en la información del histograma de la imagen (Maldonado, 2007; Liu,2002; King et al.,2009). Según la extracción basada en la forma del elemento, estas pueden ir desde lo más simple considerando las propiedades geométricas del elemento como área y perímetro (Fang,2003) hasta metodologías más avanzadas como Histogramas de orientación de gradiente (HOG) (Greenhalgh,2012).

La segunda etapa es el reconocimiento para ello se usan diversos métodos algunos con más costo computacional que otros como redes neuronales, matching of Chamfer, algoritmos genéticos, y máquinas de vectores soporte (Maldonado, 2014; Greenhalgh,2012). Multilayer Feed forward es supervisada por una red neuronal de retropropagación ha sido usada por su fácil implementación (Lorsakul,2007; Mahammad,2014). Uno de los métodos utilizados en los últimos años es el reconocimiento mediante redes neuronales convolucionales

(Pérez,2016). Para optimización de redes neuronales se han usado los algoritmos genéticos que ayudan a reducir el tiempo de análisis, pero estos no influyen en el resultado de la etapa de reconocimiento de las señales (Karungaru,2013).

1.2 PLANTEAMIENTO DE LA REALIDAD PROBLEMÁTICA

El uso de tecnologías aplicado a la ingeniería civil es cada vez mayor, y uno de los objetivos principales es la optimización de procesos los cuales implican tiempo y costos (Schmitt,2015). Las señales de tránsito verticales nos brindan información acerca de lugares, advierten de peligros, ordenan conductas de seguridad, y si se encontraran en mal estado puede ser una de las causas de accidentes viales, por lo tanto, su buen mantenimiento ayuda en la mejora de seguridad vial de peatones y conductores. Una de las herramientas de gestión en infraestructura vial que permite optimizar la asignación de recursos humanos, económicos y financieros disponibles o permite implementar metodologías para el continuo mantenimiento son los inventarios viales, y para ello es necesario crear una base de datos continua (MTC, 2016). Sin embargo, los inventarios viales pueden requerir de tiempo e inversión constante.

Por lo indicado en el párrafo anterior se formula la siguiente pregunta: ¿cómo reconocer e identificar una señal de tránsito de tipo vertical para un inventario vial básico optimizando tiempo?

1.3 OBJETIVOS

1.3.1 Objetivo General

- Automatizar el proceso de recolección de datos referidos a las señales de tránsito preventivas en el inventario vial básico.

1.3.2 Objetivos Específicos

- Revisar información en artículos científicos, tesis sobre detección y reconocimiento de señales de tránsito para evaluar las distintas técnicas de procesado y arquitecturas de clasificación.
- Recopilar y crear una base de datos de señales de tránsito a partir de secuencias de imágenes proporcionada por un sistema de cámaras y en aplicaciones web como google Street.
- Construir algoritmos eficientes y fiables para reconocimiento de señales de tránsito.

1.4 HIPÓTESIS

Las señales de tránsito serán detectadas e identificadas mediante el uso de algoritmos de segmentación basados en la extracción de características propias de las señales de tránsito y el uso de algoritmos de clasificación supervisada.

CAPÍTULO II. MARCO TEÓRICO Y CONCEPTUAL

2.1 SEÑALIZACIÓN VERTICAL

“Son dispositivos instalados al costado o sobre el camino, tienen por finalidad, reglamentar el tránsito, prevenir e informar.” (MTC, 2018)

Teniendo en cuenta el Manual de Dispositivos de Control de tránsito tenemos 3 tipos de señales verticales reguladoras, prevención e información.

Señales reguladoras que tienen la finalidad informar a los usuarios de las prohibiciones, prioridades, restricciones, obligaciones y autorizaciones existentes, no cumplir o pasar por alto estas señales constituye una falta que puede acarrear un delito.



Figura 1 Ejemplo de señales reguladoras

(Fuente: MTC, 2018)

Señales preventivas tienen el fin es prevenir a los conductores sobre algún riesgo que pueda haber en la naturaleza o situaciones imprevistas, por lo general se da un fondo color amarillo y símbolo de color negro.

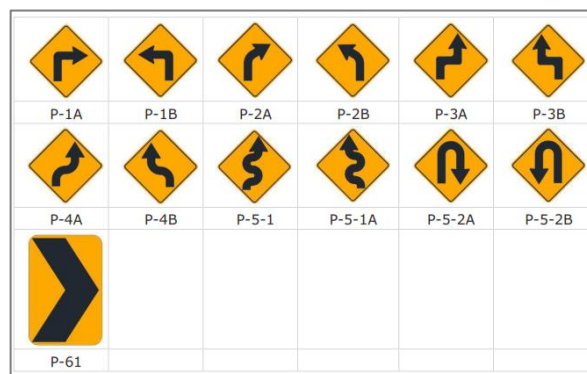


Figura 2 Ejemplo de señales preventivas

(Fuente: MTC, 2018)

Señales informativas tienen el fin de informar a los conductores y brindarles información acerca de ciudades, kilometrajes de rutas, nombres de calles, lugares turísticos, y otros



Figura 3 Ejemplo de señales informativas

Fuente: MTC, 2018

Para el desarrollo de esta investigación se tiene como objeto de estudio las señales verticales preventivas motivo por el cual se describe con detalle la información brindada en el manual.

Las señales preventivas tienen como característica principal su forma rectangular o romboide, solo se tienen tres tipos de señales con forma rectangular que son la de cruce ferroviario, prohibido adelantar y delineador de curva horizontal. Son de color amarillo y amarillo limón fluorescente con símbolos, letras y/o números de color negro. Sin embargo, también tiene algunas excepciones.

La ubicación de estas señales debe permitir que los conductores tengan tiempo para una respuesta rápida y adecuada de tal manera que tome una decisión y la ejecute de manera segura. La distancia desde la señal preventiva al lugar de peligro está en función a la velocidad límite, distancia de reducción de velocidad y cambio de carril.

Las señales de tipo preventivas se clasifican según el manual antes mencionado teniendo en cuenta lo siguiente:

- ✓ Características Geométricas de la vía:
 - Curvatura horizontal.
 - Pendiente longitudinal.

- ✓ Características de la superficie de rodadura.
- ✓ Restricciones físicas de la vía.
- ✓ Intersecciones con otras vías.
- ✓ Características operativas de la vía.
- ✓ Emergencias y situaciones especiales.

2.2 INVENTARIO VIALES

El inventario vial es una herramienta de planificación y gestión de carreteras, que sirve como guía para la toma de decisiones; este conjunto de documentos nos brinda información técnica de la evaluación, calificación de distintas obras y elementos que forman parte de la vía como puentes, pavimento, alcantarillas, señalización. (MTC, 2016)

Un inventario vial se constituye de dos tipos principales: inventario vial básico que es un documento donde se identifican y registran la ubicación de los elementos principales de la vía y sus características básicas; e inventario vial calificado, que es un documento en el cual se registran datos del trazo geométrico, características estructurales del pavimento y los estados de operación de la vía. (MTC, 2016)

Dentro de una red es importante la evaluación, calificación y planificación para realizar un análisis de los recursos que se deben destinar a las obras dentro de una vía o un conjunto de ellas para su buen funcionamiento, y también implementar un sistema de gestión de infraestructura vial.

El manual de inventarios viales del Ministerio de Transporte y Comunicaciones cuenta con un diagrama de flujo que explica el funcionamiento de un inventario vial, en la Figura 4 podemos observar que tiene dos divisiones en cada una enumera las distintas actividades que se deben realizar para cada etapa; en el inventario básico se obtendrá una base de datos y finalmente en el inventario calificado es necesario la reactualización y la creación de bases SIG u otras para la correcta gestión de información.

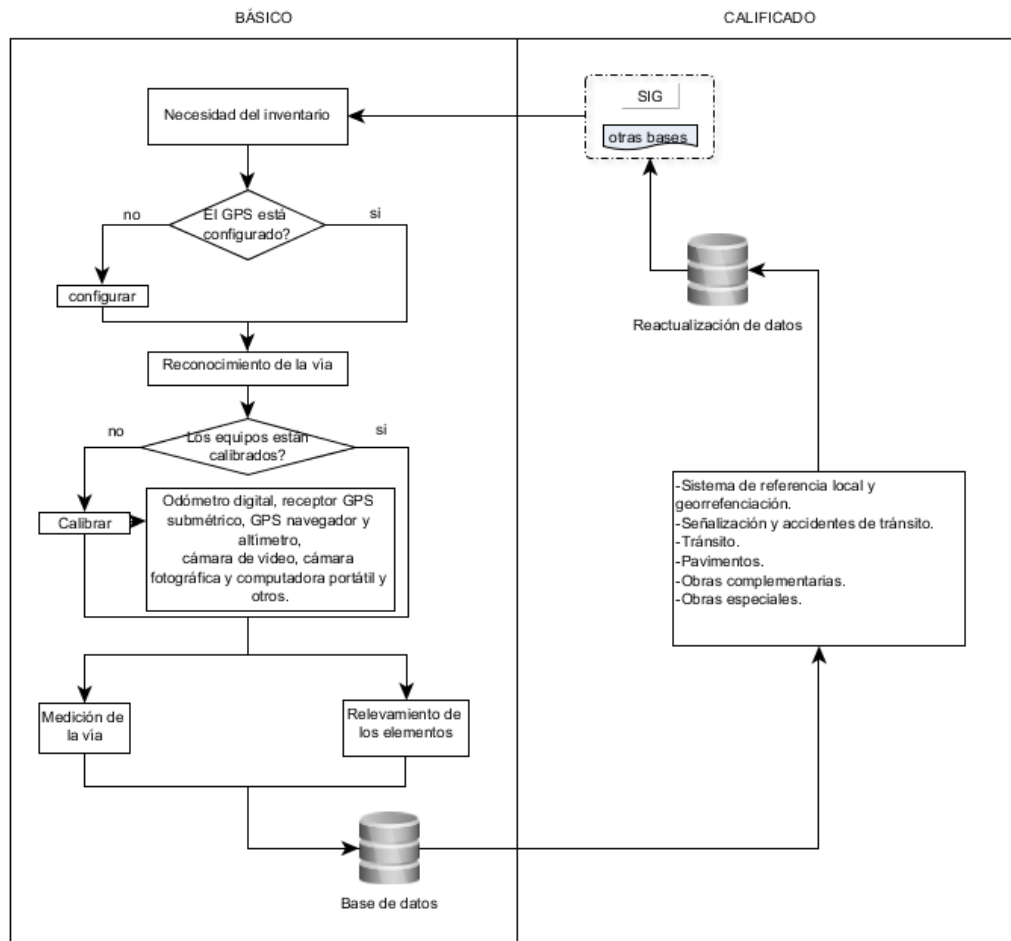


Figura 4 Diagrama del flujo explicativo de un inventario vial

Fuente: MTC, 2016

2.2.1 Inventario vial básico

Este acápite describe los equipos utilizados, y los trabajos realizados en campo y gabinete.

2.2.1.1 Equipos del inventario

Dentro del manual de inventario vial se describe el ítem de equipos del inventario vial básico, siendo uno de los más importantes el navegador GPS, es una herramienta para gestionar el almacenamiento de información georreferenciada de los elementos. A continuación, se presenta una lista con los equipos utilizados.

- ✓ Computadora portátil
- ✓ Receptor de georeferenciación GPS
- ✓ GPS submétrico
- ✓ GPS Navegador

- ✓ Cámara de video
- ✓ Odómetro digital
- ✓ Altimetro
- ✓ Equipos de almacenamiento de información digital
- ✓ Equipo de comunicación
- ✓ Camioneta 4x4
- ✓ Herramientas y accesorios

2.2.1.2 Trabajo de campo

En el manual de inventario viales se describe una metodología para la recolección de información en campo que consta de tres etapas:

- Reconocimiento de la ruta,
- Medición de la vía, y
- Relevamiento.

El diagrama general del trabajo en campo se muestra en la Figura 5. A continuación, se describe las 3 etapas.

La primera etapa abarca las actividades relacionadas con el estudio preliminar de la vía o inspección, identificación de características principales, organización del equipo de trabajo y las actividades que se desarrollan. El manual menciona que esta etapa es optativa, es decir dependerá de la entidad competente. La Tabla 1 muestra los recursos necesarios e información de la vía que debe ser registrada para esta etapa.

Para la segunda etapa es necesaria la calibración de los equipos, el objetivo principal será determinar la medición de la longitud total de la vía, el procedimiento para la georeferenciación, ubicar el punto de inicio, determinar la trayectoria. Se realizan las actividades para la toma de información para ello las brigadas recorren el tramo en estudio y con ayuda de los equipos antes mencionados inician con la recopilación de datos. El formato para la medición de la vía, es el entregable en esta etapa.

En la tercera etapa el manual menciona que se levanta la información de los elementos principales de la vía como puentes y señales, se describe su estado de conservación y el georeferenciación de estos elementos. Finalmente se elabora una ficha resumen, y el entregable final es un formato.

Tabla 1 Ficha de reconocimiento de la vía

UNIDAD DE MEDIDA	Km
PERSONAL	1 ingeniero 1 conductor
EQUIPO	Camioneta 4x4 (máximo tres años de antigüedad) Receptor
MATERIALES	Pintura Clavos o estacas
TRABAJOS A REALIZAR	<ol style="list-style-type: none">0. Configuración de equipos1. Determinación del punto inicial2. Ubicación de los puntos notables3. Ubicación de elementos fijos de control4. Determinación de la trayectoria en vías de Evitamiento5. Determinación de la trayectoria en bifurcaciones6. Determinación del punto final.
CONFIGURACIÓN DE GPS	La configuración del GPS navegador se realiza de acuerdo al manual del equipo suministrado por el fabricante y con la Norma Técnica de Levantamiento Geodésicos del IGN.
INDICADOR DE COMPROBACIÓN	Ficha de reconocimiento del punto inicial y punto final
FRECUENCIA	La frecuencia cada 3 años a partir de la finalización del ultimo inventario vial

Fuente: MTC,2016.

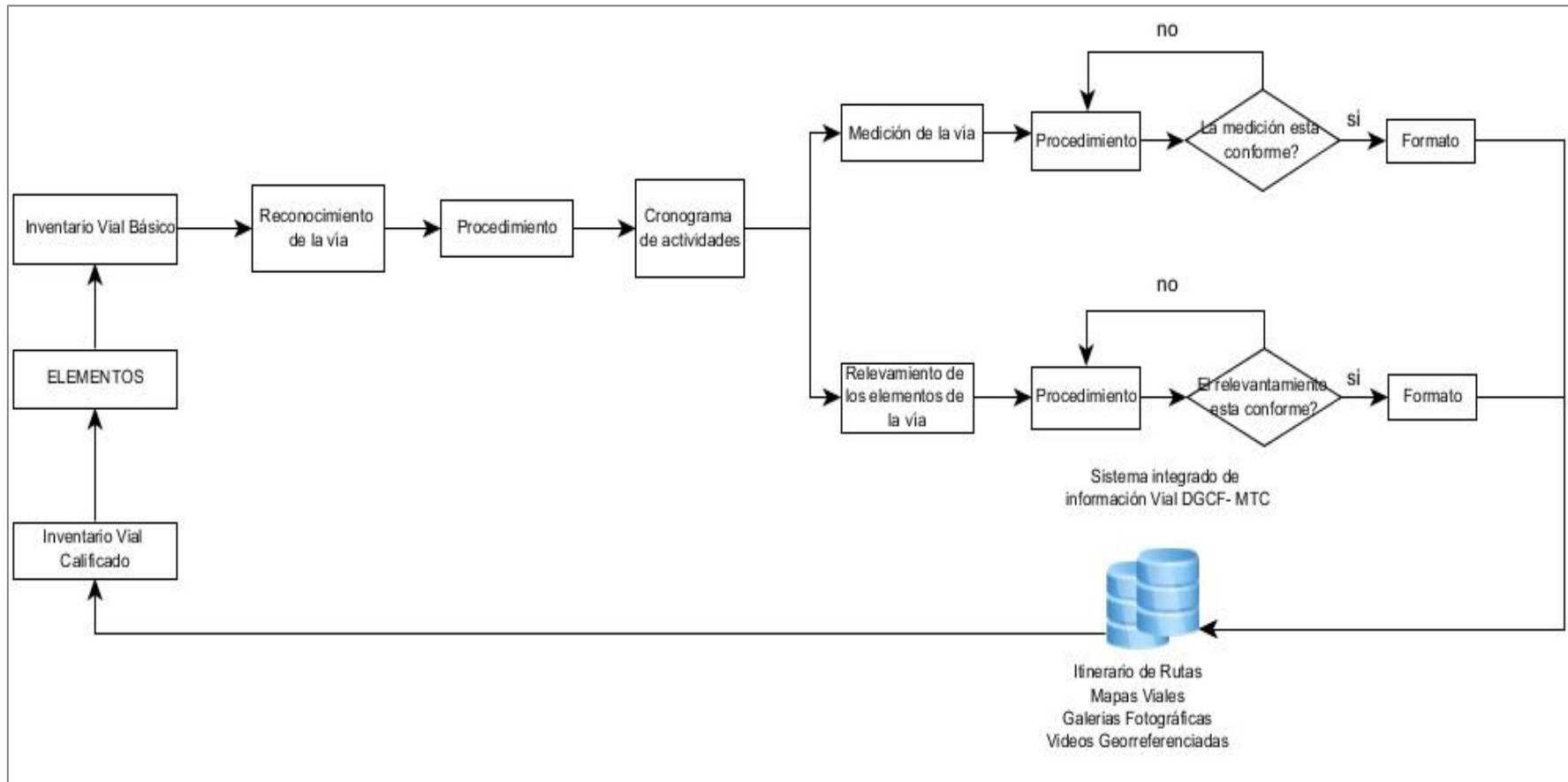


Figura 5 Reconocimiento de campo

Fuente: MTC, 2016

2.2.1.3 Trabajo de gabinete

En el trabajo de gabinete se debe realizar un análisis de las características principales del inventario. Se describirá el resumen del trabajo realizado, la situación actual y los entregables. Para este trabajo de investigación solo se describe la señalización que es el objetivo principal de la tesis.

El manual de inventarios viales enumera una serie de pasos a seguir para realizar el inventario de señalización:

- ✓ Ubicar la antena GPS submétrico en el centro del techo del vehículo para iniciar el recorrido.
- ✓ Determinar el lado de la ubicación de la señal.
- ✓ Indicar el tipo de señal.
- ✓ Reconocer el texto o símbolo escrito dentro de la señal.
- ✓ Estacionar el vehículo en el punto más cercano del eje de la vía respecto al elemento.
- ✓ Registrar las coordenadas geográficas con GPS y altitud.

2.3 VISIÓN POR COMPUTADOR

Para entender los capítulos siguientes será necesario entender algunos de los conceptos principales de la visión por computador que incluye el uso de imágenes.

2.3.2 Pixel

El pixel es el elemento de la imagen viene de la abreviación en inglés “picture element”, cada pixel de una imagen ofrece información sobre determinada región.

2.3.3 Espacios de color

Dentro del campo de procesamiento de imágenes se encuentran diferentes modelos de color. En la siguiente sección mostraremos algunos de los modelos más utilizados en el procesamiento de imágenes

RGB: El modelo RGB es uno de los más conocidos nos brinda información de 3 canales cada uno contiene la intensidad de la luz del color rojo, verde y azul. Este modelo está basado en la percepción del ojo humano y es muy común en la vida diaria es usado por las cámaras digitales de video, video juegos u otros.

Se representa en un sistema cartesiano de coordenadas como se muestra en la Figura 6, donde la línea entre el origen y el vértice contrario pertenece a las

imágenes en escala de grises, esta línea tiene su inicio en el valor y final en el color blanco. Además, los diferentes colores dentro del cubo tienen su representación en un punto y cada valor de la coordenada representa al valor de cada canal respectivamente. El rango de la intensidad de pixeles depende de la cantidad de bits el más usual es de 8 bits donde la luminosidad varía entre 0-255 en la escala de grises.

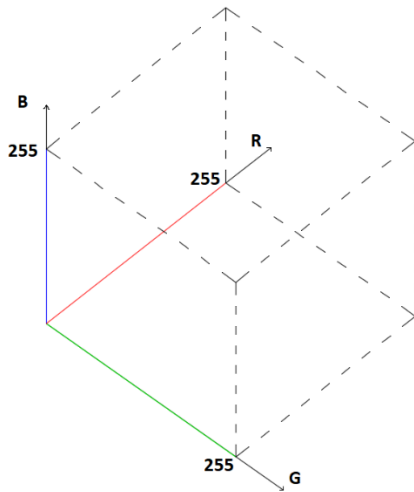


Figura 6 Modelo RGB

Fuente: Elaboración propia

HSV: El modelo HSV (Hue: Matiz, Saturation: Saturación y Value: Luminosidad) creado por Alvy Ray Smith, es una transformación no lineal del modelo de color RGB, como se observa en la Ecuación (1), (2) y (3). En la Figura 7 se aprecia un cono invertido donde el matiz o tono (H) varía entre un ángulo de 0 y 360° cada ángulo corresponde a color diferente, la saturación(S) representado como el radio de la sección circular varía de 0-1 y la luminosidad(V) representado como las aristas de un cono que varía de 0-1 es la cantidad de luz que tiene el objeto o que tan brillante es el objeto.

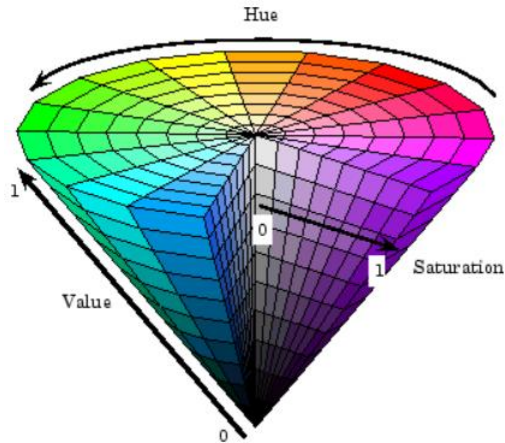


Figura 7 modelo HSV

Fuente: Mathworks

La Ecuación (1), (2) y (3) son ecuaciones de transformación para cada canal, estas ecuaciones no son lineales:

$$H = \begin{cases} \text{no definido} & \text{si } MAX = MIN \\ 60 \times \frac{G - B}{MAX - MIN} + 0 & \text{si } MAX = R \text{ y } G \geq B \\ 60 \times \frac{G - B}{MAX - MIN} + 360 & \text{si } MAX = R \text{ y } G < B \\ 60 \times \frac{G - B}{MAX - MIN} + 120 & \text{si } MAX = G \\ 60 \times \frac{G - B}{MAX - MIN} + 240 & \text{si } MAX = B \end{cases} \quad (1)$$

$$S = \begin{cases} 0 & \text{si } MAX = G \\ 1 - \frac{MIN}{MAX} & \text{en otro caso} \end{cases} \quad (2)$$

$$V = MAX \quad (3)$$

Donde:

MAX: El valor máximo de las componentes RGB

MIN: el valor mínimo de las componentes RGB

2.4 SEGMENTACIÓN DE IMAGEN

La segmentación es una técnica que permite extraer elementos de interés de una imagen por alguna característica en especial. Se conocen muchos métodos entre

ellos segmentación por contorno, textura o color. Esta técnica permite extraer un objeto sin consumir mucha memoria computacional sin embargo uno de los principales problemas que presenta es el ruido generados por elementos con las mismas características dentro de la imagen, por la resolución de la cámara, iluminación o clima. En los capítulos posteriores se indican algunas técnicas que permiten reducir o eliminar el ruido en la imagen.

Asimismo, hay diversas maneras o métodos de segmentación que depende del tipo de característica del objeto que se desea extraer.

Segmentación de contornos: El contorno de un elemento es un conjunto de pixeles que lo separan del fondo o también conocido como background, en esta técnica se basa en cambios de matiz, brillo, o cambios de patrón dentro de la imagen.

Segmentación por umbralización: Se analiza el histograma de la imagen para hallar el umbral sin embargo dos imágenes pueden tener el mismo histograma, este es un problema de la umbralización fija se da mediante las siguientes condiciones, la imagen final es una:

$$I(i, j) = \begin{cases} i, & \text{si } I(i, j) \geq U \\ j, & \text{si } I(i, j) < U \end{cases} \quad (4)$$

Umbral Generalizado: Para este caso teniendo en cuenta que las características locales de los pixeles pueden cambiar ya sea por efectos de iluminación, lo que hace variar las intensidades de los colores en la imagen es así que un objeto de color amarillo puede tornarse más claro de lo que es.

$$I(i, j) = \begin{cases} 1, & \text{si } I(i, j) \in \mathbb{R} \\ 0, & \text{en otro caso} \end{cases} \quad (5)$$

Segmentación basada en el color: La característica principal para extraer dentro de la imagen es un color en específico, la imagen a color se divide en varios canales dependiendo el modelo de color, para RGB la imagen será dividida en 3 canales, y dentro de cada uno de los canales el color tendrá una intensidad definida por cada canal. Sin embargo, dependiendo de la incidencia de la luz al objeto la intensidad variará dentro de un rango, en algunos casos la variación será muy grande por ejemplo cuando el objeto se encuentra a contra luz en este caso la intensidad dentro de cada canal será cercana a 0, en otras palabras, el objeto

no se distingue. Este tipo de segmentación realiza una búsqueda en las imágenes del color que se encuentra en los rangos definidos inicialmente.

2.5 OPERACIONES MORFOLÓGICAS

Las operaciones morfológicas son técnicas simples que permiten reducir o mejorar las características de la imagen. Se detallarán dos operaciones morfológicas muy conocidas, las cuales son las operaciones básicas:

2.5.4 Dilatación:

Se considerará dos elementos A y B. El elemento B es el elemento que dilata al elemento A por medio de una operación de convolución, y el elemento B se conoce como elemento estructurante de la dilatación (Vélez, 2003).

Esta operación produce un efecto de dilatar el aspecto del elemento A. Como se observa en la Figura 8 el elemento estructurante se desplaza por los elementos del objeto a dilatar.

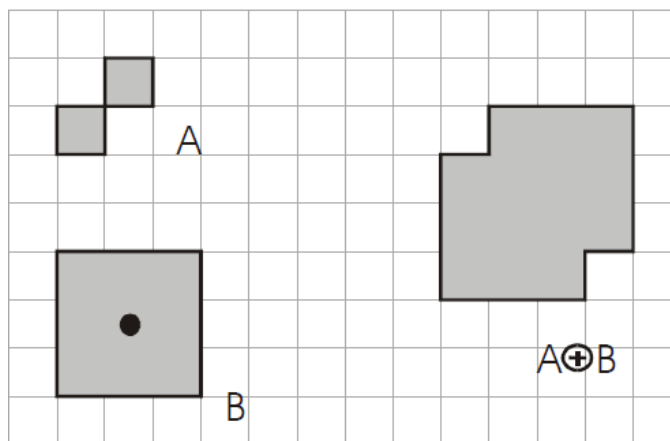


Figura 8 Dilatación del elemento A, con el elemento estructurante B

Fuente: Vélez, 2003

2.5.5 Erosión

La erosión es la operación contraria a la dilatación, A es el elemento a erosionar y B el elemento estructurante. En este caso la erosión adelgaza el elemento sobre la que se aplica el elemento estructurante (Vélez, 2003). El proceso de erosión coloca el punto negro del elemento estructurante por todos los píxeles de borde, siendo eliminados los píxeles que coinciden con ambos elementos.

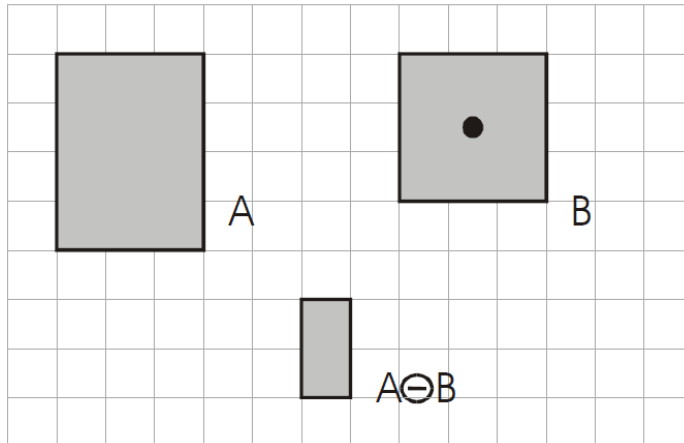


Figura 9 Erosión

Fuente: Vélez, 2003

2.5.6 Apertura

La apertura es una combinación de operaciones morfológicas, primero aplicando la operación de erosión y luego la dilatación de A con B.

La apertura, amplía las zonas de pixeles inactivos presentes en una zona de pixeles activos. Este proceso ayuda a reducir ruido conocido como sal y pimienta. para mejorar la imagen segmentada o binaria, la Figura 10 (a) es un claro ejemplo de una imagen con ruido y es eliminado mediante la apertura usando un elemento estructural por ejemplo de 3x3, es decir una matriz de 3x3, el resultado se observa en la Figura 10 (b).

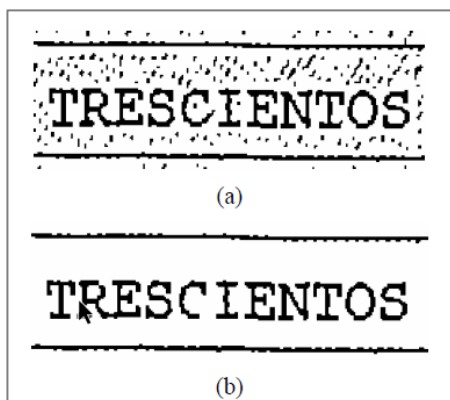


Figura 10 Ejemplo de eliminación de ruido de la figura

Fuente: Web Google

2.5.7 Cierre

De igual forma el cierre es la combinación de las operaciones mencionadas anteriormente, sin embargo, en diferente orden primero se aplica la operación de dilatación y luego la de erosión.

El cierre elimina zonas de pixeles inactivos presentes en el interior de una zona de pixeles activos.

2.6 DESCRIPTOR: HISTOGRAMA DE ORIENTACIÓN DE LA GRADIENTE

El histograma de orientaciones de gradiente (HOG de sus siglas en inglés Histogram Oriented gradient) es un descriptor que tiene como característica principal describir un objeto por su forma. El cambio direccional en la intensidad de la imagen se da en las dos direcciones (horizontal y vertical).

El histograma de orientaciones gradiente fue propuesto por Dalal y Triggs en 2005 para su aplicación en detección de personas. La idea principal es que la apariencia de un objeto y forma pueden ser bien caracterizados a través de la distribución local de gradientes o la dirección del contorno. Este método calcula los gradientes de la imagen de cada pixel, los gradientes miden el cambio relativo entre dos puntos, es decir la variación entre los pixeles adyacentes en una misma dirección. Para ello se muestran ecuaciones para el cálculo de la gradiente en la vertical; variación entre el pixel de arriba y de abajo (Ecuación (1)); y en la horizontal entre el pixel de la izquierda y el de la derecha (Ecuación (2)).

$$d_y = I_{(x,y+1)} - I_{(x,y-1)} \quad \text{Ecuación (1)}$$

$$d_x = I_{(x+1,y)} - I_{(x-1,y)} \quad \text{Ecuación (2)}$$

Donde:

d_y, d_x : variación entre pixeles en la dirección Y y X respectivamente

$I_{(x,y)}$: Coordenada de un pixel

Con el cálculo anterior el gradiente de la imagen es representado con la magnitud y la orientación de la siguiente manera:

$$\text{Magnitud } D = \sqrt{d_x^2 + d_y^2} \quad \text{Ecuación (3)}$$

$$\text{Orientación } \theta = \arctan\left(\frac{d_y}{d_x}\right) \quad \text{Ecuación (4)}$$

Para el cálculo del histograma es necesario la agrupación de los vectores (D, θ), un bloque está conformado por celdas y estas a su vez por píxeles Figura 11 las celdas son los rectángulos de color verde. El rango de los intervalos de la orientación (θ) puede ser entre 0-360 considerando el signo o 0-180 grados sin considerar el signo, una configuración común para el segundo rango es dividirlo en 9 intervalos de 20 grados cada uno, la magnitud (D) de cada intervalo del histograma será la suma de las magnitudes de los vectores en el bloque que tengan su orientación dentro del mismo intervalo Figura 12. Finalmente se obtendrá un histograma concatenado Figura 13.

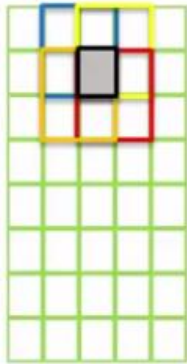


Figura 11 Descripción gráfica de elementos (Estructura de celda y bloques)

Fuente: Curso UAB, Coursera

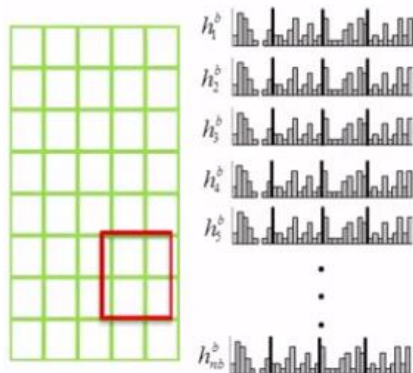


Figura 12 Histograma de los bloques de la imagen

Fuente: Curso UAB, Coursera

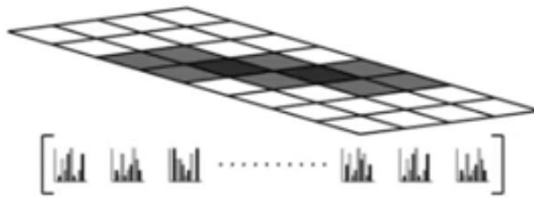


Figura 13 Representación de histograma concatenado

Fuente: Dalal and Triggs, 2005

En la Figura 14 (izquierda) es una señal vertical peatonal de 64x64 y en la Figura 14 (derecha) el descriptor HOG de la imagen se visualiza en histogramas normalizados de 9x1, es decir 9 intervalos de la orientación de celdas de 8x8, se observa de la dirección dominante del histograma captura la forma de la señal, especialmente en los bordes.



Figura 14 HOG de señal peatonal

Fuente: Elaboración propia

Dentro del procedimiento anterior, en algunos casos ciertos gradientes muy similares pueden encontrarse dentro de intervalos diferentes para este problema, se recomienda multiplicar un peso proporcional a la distancia de la orientación al centro de cada intervalo, este procedimiento permite que un gradiente contribuya no solo a un intervalo sino a ambos intervalos.

2.7 SOPORTE DE MÁQUINAS DE VECTORES

El soporte de máquina vectoriales (SVM, por sus siglas en inglés Support vector machine) es un algoritmo el cual puede ser usado para problemas de clasificación o regresión. Este algoritmo fue propuesto por Vapnik y sus colaboradores (Cortes & Vapnik, 1995; Boser et al., 1992) es un clasificador lineal ya que separa dos grupos por un hiperplano que cumple con ciertas restricciones, existen diferentes tipos de datos: datos separables que es el caso más sencillo; sin embargo, este es un caso ideal ya que siempre encontraremos datos con ruido incluido, el segundo

tipo son los datos cuasiseparables; y el tercer tipo datos donde el hiperplano tiene altas dimensiones, estos datos no son linealmente separables en el espacio original llamado espacio de características.

La SVM se fundamenta en la búsqueda de un hiperplano que separe convenientemente dos grupos distintos de elementos, este hiperplano cumple con equidistar de los elementos de ambos grupos.

Carmona (2013) describe y explica de manera sencilla como hallar el hiperplano de separación para clasificar correctamente dos grupos de datos separable; teniendo una muestra de datos $S = \{(x_1, y_1), \dots, (x_n, y_n)\}$ considerados como datos separables $x_i \in \mathbb{R}^d$ e $y_i \in \{1, -1\}$, donde y_i representa una etiqueta para distintas clases, se puede definir un hiperplano de separación como se observa en la Figura 15 (a). La siguiente función define el hiperplano de separación.

$$D(x) = (w_1x_1 + w_2x_2 + \dots + w_dx_d) + b = \langle w, x \rangle + b \quad (6)$$

Donde w y b son coeficientes reales. Además, el hiperplano cumplirá con las siguientes restricciones para x_i del conjunto:

$$\langle w, x_i \rangle + b \geq 0 \quad \text{si } y_i = +1 \quad (7)$$

$$\langle w, x_i \rangle + b \leq 0 \quad \text{si } y_i = -1, i = 1, \dots, n \quad (8)$$

o también,

$$y_i(\langle w, x_i \rangle + b) \geq 0 \quad , i = 1, \dots, n \quad (9)$$

De forma más reducida:

$$y_i D(x_i) \geq 0 \quad , i = 1, \dots, n \quad (10)$$

Sin embargo, existen infinitos hiperplanos que cumplen con la formulación anterior como se observa en la Figura 15 (b). El objetivo es hallar un hiperplano óptimo que tenga el mínimo error al separar los datos, para ello Vapnik definió el concepto de margen (τ) que es la mínima distancia del hiperplano al elemento de cualquier grupo de los datos analizados (Figura 16 (a)); los elementos de cada grupo que definirán el hiperplano reciben el nombre de vector soporte. Una vez encontrado

los vectores soporte se obtendrá el hiperplano óptimo solo si su margen es de tamaño máximo, se debe tener en cuenta que este equidista del dato más cercano de cada grupo como se observa en la Figura 16 (b).

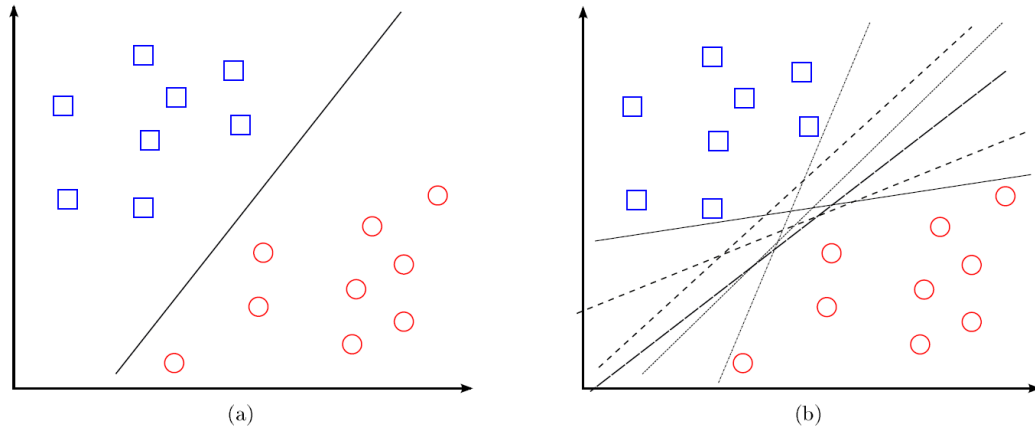


Figura 15 Ejemplo de hiperplano de separación
Fuente: Carmona,2013

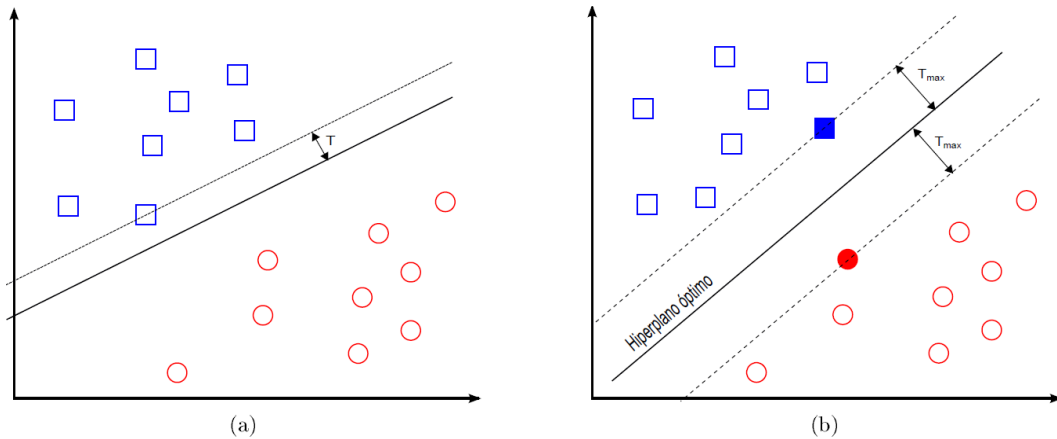


Figura 16 Margen de un hiperplano de separación
Fuente: Carmona,2013

La distancia de un elemento x_i a un hiperplano denominado $D(x')$ se define en la Ecuación (11) como el margen y además tiene la propiedad de ser perpendicular al hiperplano

$$Margen = \frac{|D(x')|}{\|w\|} \quad (11)$$

$|D(x')|$: Valor absoluto de hiperplano

$\|w\|$: valor de la norma del vector w

De la Ecuación(10) y (11), los datos entrenados cumplirán la expresión siguiente:

$$\frac{y_i D(x_i)}{\|w\|} \geq \tau \quad , i = 1, \dots, n \quad (12)$$

Entonces para encontrar el hiperplano óptimo es necesario encontrar el valor de w que maximiza el margen según la Ecuación (12). Sin embargo existen infinitas soluciones, como ejemplo Carmona menciona que las funciones lineales $\lambda(\langle w, x_i \rangle + b)$ con $\lambda \in \mathbb{R}$ cumplen con la expresión. La Ecuación (12) también puede ser expresada como

$$y_i D(x_i) \geq \tau \|w\| \quad , i = 1, \dots, n \quad (13)$$

El producto escalar de τ y la norma de w se fija de forma arbitraria en la unidad, es decir:

$$\tau \|w\| = 1$$

Por tanto el τ máximo se da cuando la norma del vector w es mínima y está sujeta a la restricción dada por

$$y_i D(x_i) \geq 1 \quad , i = 1, \dots, n \quad (14)$$

Del tutorial de Carmona 2013 nos brinda una metodología para la búsqueda del hiperplano. Para ello es necesario encontrar el valor de w y b que minimice la función $f(w) = \|w\|$ sujeto a la restricción dada en mencionada

$$\min f(w) = \frac{1}{2} \|w\|^2 = \frac{1}{2} \langle w, w \rangle \quad (15)$$

$$s. a. y_i (\langle w, x_i \rangle + b) - 1 \geq 0, i = 1, \dots, n$$

El problema de optimización con restricciones es un problema de optimización cuadrática y puede ser resuelto bajo la teoría de optimización. En la teoría menciona que un problema primal, tiene la forma dual si la función objetivo y las funciones de restricción son estrictamente convexas, solo en este caso solo es necesario resolver el problema dual para obtener la solución 'del problema primal.

1er paso: Conseguir la función lagrangiana, esta función nos permite incluir la función objetivo y función de restricción en una única función como se muestra en la ecuación

$$L(w, b, \alpha) = \frac{1}{2} \langle w, w \rangle - \sum_{i=1}^n \alpha_i [y_i (\langle w, x_i \rangle + b) - 1] \quad (16)$$

Donde $\alpha_i \geq 0$ son denominados multiplicadores de Lagrange.

2do paso: Aplicar las condiciones de Karush- Kuhn-Tucker (Obtenido de Carmona, 2013).

$$\frac{\partial L(w^*, b^*, \alpha)}{\partial w} \equiv w^* - \sum_{i=1}^n \alpha_i y_i x_i = 0 \quad , i = 1, \dots, n \quad (17)$$

$$\frac{\partial L(w^*, b^*, \alpha)}{\partial w} \equiv \sum_{i=1}^n \alpha_i y_i = 0 \quad , i = 1, \dots, n \quad (18)$$

$$\alpha_i [1 - y_i (\langle w^*, x_i \rangle + b^*)] = 0, i = 1, \dots, n \quad (19)$$

Paso 3:

Las ecuaciones (17) y (18) son el resultados de aplicar la condición de Karush- Kuhn- Tucker y la ecuación (19) es el resultado de la segunda condición.

Las primeras permitirán expresar los parámetros w y b en términos de α_i :

$$w^* \equiv \sum_{i=1}^n \alpha_i y_i x_i \quad , i = 1, \dots, n \quad (20)$$

Paso 4: Establecer restricción para los coeficientes α_i

$$\sum_{i=1}^n \alpha_i y_i \quad , i = 1, \dots, n \quad (21)$$

Paso 5: Con las relaciones obtenidas en el paso 3, se construirá un problema dual para usar la función lagrangiana en función de α_i , se reemplazará las relaciones halladas en la ecuación (20) en la ecuación (22). Y además se observa que el tercer término de la ecuación (22) es nulo

$$L(w, b, \alpha) = \frac{1}{2} \langle w, w \rangle - \sum_{i=1}^n \alpha_i y_i \langle w, x_i \rangle - b \sum_{i=1}^n \alpha_i y_i + \sum_{i=1}^n \alpha_i \quad (22)$$

$$\begin{aligned} L(\alpha) &= \frac{1}{2} \left(\sum_{i=1}^n \alpha_i y_i x_i \right) \left(\sum_{i=1}^n \alpha_j y_j x_j \right) - \left(\sum_{i=1}^n \alpha_i y_i x_i \right) \left(\sum_{i=1}^n \alpha_j y_j x_j \right) + \sum_{i=1}^n \alpha_i \\ L(\alpha) &= \sum_{i=1}^n \alpha_i - \frac{1}{2} \left(\sum_{i=1}^n \alpha_i y_i x_i \right) \left(\sum_{i=1}^n \alpha_j y_j x_j \right) \\ L(\alpha) &= \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle \end{aligned} \quad (23)$$

Finalmente se debe maximizar (22) sujeto a las restricciones del paso 4 Ecuación (40), junto a las asociadas a los multiplicadores de Lagrange

$$\begin{aligned} \max L(\alpha) &= \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle \\ & \text{s. a } \sum_{i=1}^n \alpha_i y_i = 0 \\ & \alpha_i \geq 0, \quad i = 1, \dots, n \end{aligned} \quad (24)$$

La solución del problema dual, α^* , permite obtener la solución del problema primal

$$D(x) = \sum_{i=1}^n \alpha_i^* y_i \langle x, x_i \rangle + b^* \quad (25)$$

$$y_i (\langle w^*, x_i \rangle + b^*) = 1 \quad (26)$$

Para resolver la ecuación (25) es necesario determinar el valor de b^* . Despejando b^* de la ecuación (44):

$$b^* = y_{sv} - \langle w^*, x_{sv} \rangle \quad (27)$$

Donde (x_{sv}, y_{sv}) representa la información de los vectores soportes. Carmona 2013 menciona que en la práctica una solución más robusta es usar el valor de b^* promediado de todos los vectores soportes utilizados, N_{sv} . La expresión se transforma en la siguiente ecuación (28):

$$b^* = \frac{1}{N_{sv}} \sum_{i=1}^{N_{sv}} y_{sv} - \langle w^*, x_{sv} \rangle \quad (28)$$

Se debe tener en cuenta que los vectores soporte, es decir los ejemplos de cada clase que serán los únicos a considerar para la construcción del hiperplano.

Usando la expresión de la ecuación (20) en la ecuación (27) o (28) permitirá calcular el valor de b^* en función de la solución del problema dual.

La teoría y formulaciones presentadas en este capítulo se usan para la clasificación binaria, no obstante, actualmente se encuentran varias herramientas implementadas en diferentes softwares como en Matlab para el uso de este algoritmo para la multclasificación.

CAPÍTULO III. ESTADO DE ARTE

3.1 NORMATIVA

3.1.1 Manual de dispositivos de control de tránsito

El manual de dispositivos de control de tránsito automotor para calles y carreteras se aprobó el 13 de octubre de 1993 mediante la resolución Ministerial N° 413-93-TCC/15.01. Luego el 03 de mayo del 2000 se aprobó la actualización mediante resolución N° 210-2000-MTC/15.02. En los siguientes años 2000,2008,2014,2016 se hicieron modificaciones.

Este manual debe ser usado obligatoriamente para los distintos niveles del gobierno nacional, regional y local, el manual brinda los requerimientos necesarios como su modo de uso por su funcionalidad, color, tamaños, formas, clasificación y otros. El manual indica que los dispositivos mencionados dentro de este serán usados en la movilidad y regulación de tránsito como para el diseño, rehabilitación, mantenimiento, construcción y tránsito temporal en zonas de trabajo.

El manual está dividido en 6 capítulos que se muestran a continuación:

- Capítulo 1 Generalidades
- Capítulo 2 Señales verticales
- Capítulo 3 Marcas en el pavimento o demarcaciones
- Capítulo 4 Dispositivos de control del tránsito de casos especiales
- Capítulo 5 Dispositivos de control de tránsito en zonas de trabajo
- Capítulo 6 Semáforos

3.1.2 Manual de inventarios viales

El manual de Inventarios viales se aprobó el 3 de abril del 2014 con resolución directoral N° 09-2014-MTC/14 e incorporación de la parte IV con dirección directoral N° 022-2015-MTC/14. Dicho manual, de conformidad con el artículo 18° del Reglamento Nacional de Gestión de Infraestructura Vial, constituye un documento de carácter normativo y de cumplimiento obligatorio. Contiene un conjunto de documentos oficiales de información técnica.

El objetivo principal del inventario es brindar y establecer una metodología para desarrollar y evaluar los trabajos del inventario vial de los elementos que tiene el

sistema nacional de carreteras (SINAC), tanto del inventario vial básico como el inventario vial calificado. La normativa sigue las siguientes divisiones:

- ✓ Generalidades
- ✓ Inventario vial Básico
- ✓ Inventario Vial Calificado
- ✓ Apendice

3.2 TECNOLOGÍAS USADAS PARA INVENTARIOS VIALES

Dentro de la etapa de búsqueda de información se revisó información de investigaciones destacadas con referencia a la detección y reconocimiento de señales de tránsito y de los avances que se muestran actualmente.

3.2.3 Métodos de detección

A continuación, se describirán tipos de detección, los distintos de métodos utilizados actualmente en diferentes investigaciones.

3.2.3.1 Basados en color

Investigaciones usando el Modelo RGB

En esta investigación se realizan ciertas transformaciones a cada canal de color, King et al. (2009) prefiere un modelo $r'g'b'$ o modelo RGB transformado, para ello normaliza los canales RGB por su intensidad, finalmente crea un canal adicional que es del color amarillo. En las siguientes formulas se transforma los canales del modelo RGB al modelo $r'g'b'$.

$$r' = \frac{R}{R + G + B} * 3 \quad (29)$$

$$g' = \frac{G}{R + G + B} * 3 \quad (30)$$

$$b' = \frac{B}{R + G + B} * 3 \quad (31)$$

Finalmente, para obtener los cuatro canales RGBY se reemplaza los valores $r'g'b'$ en las siguientes formulas:

$$R = r' - \frac{g' + b'}{2} \quad (32)$$

$$G = g' - \frac{r' + b'}{2} \quad (33)$$

$$B = b' - \frac{r' + g'}{2} \quad (34)$$

$$Y = \frac{r' + g'}{2} - \frac{|r' - g'|}{2} - b' \quad (35)$$

Esta investigación parte de usar 4 canales en lugar de 3, para segmentar las imágenes será necesario encontrar los rangos de color que debe ser extraído de la imagen y el uso de operaciones morfológicas para remover los pixeles no deseados.

Investigaciones usando el Modelo HSV

Algunos investigadores consideran que no es posible detectar las señales de tránsito en imágenes reales por las variaciones de luz, y es los filtros o rangos de un color en la imagen varían de acuerdo a distintos factores externos, sin embargo existen modelos de color como el HSV que son robustos o invariantes a ciertos factores externos como la variación de iluminación. Existen diferentes autores y artículos de investigación que usan el modelo HSV en sus metodologías. Yakimov (2015) usa un método experimental para determinar los filtros para el color rojo que se presenta en las siguientes expresiones

$$350 < H < 360 \quad (36)$$

$$0.85 < S \leq 1 \quad (37)$$

$$0.85 < V \leq 1 \quad (38)$$

Luego de la segmentación, usa diferentes algoritmos para reducir el ruido en la imagen.

Wang et al. (2013) elige el modelo HSV, en esta investigación encuentran que con los filtros clásicos se encuentran buenos resultados, sin embargo, estos no funcionan en imágenes con condiciones bajas de luz, por lo tanto, se propone una nueva metodología para filtros de color rojo usando la información de los pixeles vecinos, se calculan diferentes parámetros. Primero el grado o nivel de rojo es calculado para cada pixel o punto denominado c , llamado como $f_R(c)$, usando la Ecuación (39):

$$f_R(c) \begin{cases} S(c) \frac{\sin(H(c) - 300^\circ)}{\sin(60^\circ)} & \text{if } H(c) \in [300^\circ, 360^\circ] \\ S(c) \frac{\sin(60^\circ) - H(c)}{\sin(60^\circ)} & \text{if } H(c) \in [0^\circ, 60^\circ] \\ 0 & \text{otro} \end{cases} \quad (39)$$

Segundo, el canal de rojo normalizado $f_{NR}(x)$ es calculado como sigue:

$$f_{NR}(x) = \frac{(f_R(x) - \mu_R(\omega_x))}{(\sigma_R(\omega_x))} \quad (40)$$

Donde $\mu_R(\omega_x)$ y $\sigma_R(\omega_x)$ son la media y la varianza de las intensidades de los pixeles rojos en la ventana ω_x con centro en x .

Tercero, la intensidad normalizada $f_{NI}(x)$ es calculada con la siguiente ecuación:

$$f_{NI}(x) = \frac{(f_I(x) - \mu_I(\omega_x))}{(\sigma_I(\omega_x))} \quad (41)$$

Donde $f_I(x)$ es la intensidad del pixel x ; $\mu_I(\omega_x)$ y $\sigma_I(\omega_x)$ son la media y la desviación estándar de las intensidades de los pixeles en la ventana ω_x . Finalmente, el mapa de bit para el color rojo será calculado con la Ecuación (42):

$$B(x) = \begin{cases} 1, & \text{if } f_{NR}(x) > \max(THR1, f_{NI} + THR2) \\ 0 & \text{otro} \end{cases} \quad (42)$$

Investigaciones usando el Modelo HSI

Este modelo es elegido por la invariancia en los cambios de luminancia.

Escalera et al (2003) ha elegido el modelo HSI para detectar las señales de tránsito. Solo se usa la capa H y S para evitar los problemas de variaciones de luz. Ellos construyen dos look-up tables (LTUs), son tablas que contienen valores pre calculados sobretodo de expresiones altamente no lineales con el objetivo de reducir los altos costos computacionales, uno para la componente Hue y el otro para la componente S una vez calculados ambos son multiplicados y normalizados por el máximo valor 255. La idea es que cada LUT compense al otro, es decir, si una componente tiene bajos valores la otra componente pueda compensar o corregirla.



Figura 17 Detección de color

Fuente: Escalera et al. (2003)

Fang et al. (2004) elige el modelo HSI, la luz y sombras influyen en los valores de saturación e intensidad y las señales de tránsito son difíciles de detectar bajo condiciones pobres de clima por la variación constante de la iluminación. Por otro lado, el Matiz (Hue) es invariante para cada cambio en la luz diurna. La Ecuación (43) permite calcular el matiz con los valores de la imagen RGB.

$$h = \begin{cases} \frac{180}{\pi} \cos^{-1} \left\{ \frac{0.5[(r - g) + (r - b)]}{[(r - g)^2 + (r - b)(g - b)]^{1/2}} \right\} & \text{if } (g - b) \geq 0, \\ -\frac{180}{\pi} \cos^{-1} \left\{ \frac{0.5[(r - g) + (r - b)]}{[(r - g)^2 + (r - b)(g - b)]^{1/2}} \right\} & \text{if } (g - b) < 0, \end{cases} \quad (43)$$

donde r,g,b son los canales de rojo, verde y azul de los pixeles. Para cada señal será calculada el vector que contiene los todos los valores de hue $\{h_1, h_2, \dots, h_q\}$ y se asume que tiene una distribución gaussiana con varianza σ^2 , luego z será el grado de similitud siendo el máximo de los valores encontrados en la definido por la Ecuación (48)

$$z = \max(k = 1, q) \quad (44)$$

$$z_k = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{(h - h_k)^2}{2\sigma^2}\right) \quad (45)$$

La diferencia con las anteriores investigaciones es que no se obtiene una imagen segmentada si no que cada pixel representa la similitud entre pixeles de color.

Investigaciones usando el Modelo YUV

El modelo YUV tienen tres componentes basados en la separación de la luminancia e información del color como se muestra a continuación:

$$Y = 0.299R + 0.587G + 0.114B \quad (46)$$

$$U = 0.493(B - Y) \quad (47)$$

$$V = 0.877(R - Y) \quad (48)$$

En la investigación de Miura(2002) los filtros para YUV son hechos con una proyección de la gradiente horizontal y vertical. El rendimiento del algoritmo solo funciona para algunos ejemplos.

Gomez et. al. (2010) compara diferentes métodos de segmentación para encontrar el mejor método para reconocer las señales de tránsito. En esta investigación se clasifico los métodos de segmentación en tres categorías:

- Segmentación con binarización
- Descomposición del color
- Detección de forma

Luego propone un nuevo método de segmentación el cual combina SVM con LUT (look up table)

3.2.3.2 Basados en la forma

Los métodos basados en la forma son generalmente más robustos que los basados en color ya que procesan la información de la imagen en escala de grises y usan la información de sus gradientes. Sin embargo, estos tienen mayor costo computacional ya que analiza varios sectores de la imagen, es por ello que algunos autores usan métodos basados en color para adquirir o extraer las regiones de interés y luego usan los métodos basados en la forma para detectar la posición de las señales y reconocer la geometría.

Vitabile et al. (2001) usa todos los pixeles para reconocer una señal de tránsito, usa imágenes de 36x36 pixeles, luego de detectarla dentro de la imagen la región de interés es redimensionada a la misma escala de la imagen binaria para el

cálculo de la similitud usando el coeficiente de Tanimoto donde el valor del coeficiente es normalizado si es cercano a uno la región de interés es similar al modelo. El coeficiente de Tanimoto S es definido de la siguiente manera:

$$S = \frac{|x \cap y|}{|x \cup y|} \quad (49)$$

Donde X y Y son definidos como las imágenes binarias que se van a comparar.

Otra técnica utilizada es el uso de template matching o comparación de plantillas para eliminar las regiones donde no se encuentra una señal. Para esta técnica una ventana o plantilla recorre toda la imagen para buscar regiones similares y se utiliza para la función de similitud el error medio cuadrático (MSE).

$$MSE = \frac{1}{MN} \sum_{y=1}^N [T(x, y) - I(x, y)]^2 \quad (50)$$

Donde $T(x, y)$ es el valor de intensidad de la plantilla en la posición (x, y) ; $I(x, y)$ es el valor de la intensidad de la imagen de entrada en la posición (x, y) ; M y N es el ancho y altura de la imagen respectivamente.

Las transformadas de Hough son algoritmos usados para detectar los bordes de las señales y seleccionar los contornos cerrados, lo que lo hace muy sensible al ruido y oclusiones. Dentro de la investigación de García (2010) es usado la transformada de Hough. Es detectado el 97% de 435 señales de límite de velocidad y 94% de 312 señales de peligro en un tiempo que se encuentra en el rango de 20-200 ms/imagen.

Youssef et al. (2016) usa un descriptor HOG para ventanas de 40x40, bloques de 10x10 y celdas de 2x2. La base de datos usada es GTSRB para la etapa de entrenamiento y DITS es usada para la segunda etapa de entrenamiento y ambas son usadas en la tercera etapa. Para reducir los falsos positivos y el tiempo de procesamiento, es usado segmentación por color con una mejora para el modelo HSV.

Fang et. al. (2003), realiza un algoritmo en forma de cascada, considera que un color solo puede tener formas específicas, tiene en cuenta que los bordes son elementos fundamentales para la forma. Para la detección usa redes neuronales,

siendo la capa de entrada la forma de la señal como un detector de borde. La investigación detecta las siguientes formas como el círculo, rectángulo, octógono y triángulo equilátero.

Qin et al. (2010), usa un vector de distancia para el borde: la cual es la distancia entre el contorno del elemento y el bounding box. Finalmente se obtiene un vector que tiene información del margen (izquierdo, derecho, arriba y abajo).

Cardarelli et. al. (2009) presenta un método para detectar señales triangulares y rectangulares; la detección de borde está basado en el uso del operador sobel(Gupta, 2013). El operador es robusto ante la rotación, oclusión y deformación, sin embargo, tiene un aumento en el costo computacional de 10 % comparado con el detector pattern matching. La Figura 18 se observa primero una señal de peligro que será analizada, la siguiente imagen es la transformación en escala de grises, en la tercera imagen es aplicada una máscara de 5x5 obteniendo la imagen sobel normalizada y la ultima la fase sobel definido con $\arctg(Gy/Gx)$.

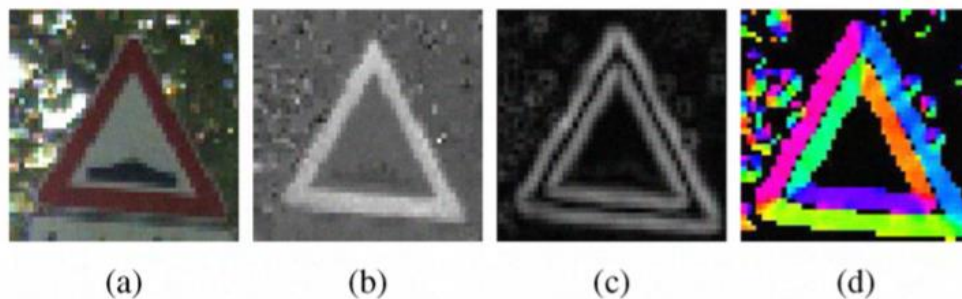


Figura 18 Ejemplos de detección de señales

Fuente: Cardelli, 2009

3.2.4 Métodos de Reconocimiento

Actualmente se cuenta con una variedad de métodos aplicado en investigaciones en el campo del reconocimiento. El problema común de los distintos algoritmos de reconocimiento es principalmente la base de datos. A continuación, se citará un resumen de la información revisada:

Investigaciones basadas en aprendizaje profundo o Deep

Sermanet & LeCun (2011), usaron redes neuronales convolucionales (CNN), ellos obtuvieron una precisión de 98.97%, el cual está por encima del reconocimiento personas de 98.81%. Para este método no es necesario el uso de información del

color. Teniendo una precisión de 99.17%. El problema del método es la amplia y compleja data que necesita. Wu et al. (2013) también aplica redes CNN, la elección de la arquitectura de las capas afecta la eficiencia de las CNN. Está constituida por las capas de filtro, Pooling layers, y finalmente la normalización.

Z. Zhu (2016) recolecta información de 100K imágenes de vistas panorámicas de las calles, las cuales contienen 300 k señales de tránsito. Para entrenar dos redes con la base de datos, una trata a todas las señales como una sola categoría y sirve para la detección de imágenes y la otra puede detectar y clasificar simultáneamente las señales de tránsito; esta etapa tiene una exactitud de 88%.

En la Figura 19 se observa que primero las señales son detectadas y encerradas en un cajetín luego es encontrada el contorno de la señal y finalmente son clasificadas dándole una etiqueta.



Figura 19 Detección y reconocimiento de imágenes

Fuente: Zhu, 2016c

En el Perú, Ayuque (2016) propone un sistema de clasificación de señales de tránsito con redes neuronales convolucionales, para ello utilizó una base de datos con señales de tránsito alemanas para el reconocimiento de señales de tránsito peruanas. El fin de esta investigación es usar el reconocimiento para el monitoreo de infracciones de señales de tránsito vehicular; Ayuque implementa 6 diseños diferentes usando TensorFlow, la red clasifica bien 6 de 7 imágenes, que equivale a 85% de porcentaje de aciertos, analiza señales reglamentarias: señal de prohibido estacionar y la señal de Pare. Concluye que la técnica de pre-procesamiento puede mejorar los resultados de la CNN. Para Ciresan et al. (2012) tiene un porcentaje de acierto de 99.46% y para Sermanet & LeCun (2011) de 99.17%

En el LACCEI 2010 realizado en la ciudad de Arequipa de Perú, Rocha et al. 2010 presenta un sistema de reconocimiento de señales a partir de las redes neuronales. La metodología consta en 4 fases inicio, elaboración, construcción y transición

En la etapa de reconocimiento, se usan momentos invariantes básicos, los momentos invariantes difusos y los momentos invariantes afines de cada una de las imágenes de prueba para entrenar la red. La red utilizada es de tipo back propagation, se utiliza este tipo de red porque reduce el error correlacionando con los campos a clasificar. En esta investigación se tomaron 73 imágenes con diferentes posiciones, distancias, obstáculos y momentos del día. Este sistema propuesto tiene la ventaja de reducir el ruido producido por los objetos del mismo color de las señales, de las dos redes propuestas en la etapa de reconocimiento, la segunda red tiene un mayor número de capas y tiene mejor reconocimiento de señales.

Abedin (2016) usan SURF con una red neuronal artificial (ANN), usan una base propia que contiene 200 imágenes capturadas en las carreteras de Bangladesh con diferentes climas y condiciones de iluminación. Tiene un porcentaje de 97% con 3% de falsos positivos sin embargo ellos tienen buenos resultados.

Redes neuronales son usadas en Alam (2015), karul (2007), Karungaru et al. (2013).

También se utiliza el método de SVM, Maldonado (2007) desarrolla un sistema de detección y reconocimiento automático basado en SVM capaz de detectar señales circulares, rectangulares, triangulares y octogonales. El sistema consta en tres etapas la etapa de segmentación por color; luego la detección usa clasificación por forma usando SVM; y finalmente reconocimiento basado en kernel gausiano de SVM, La ventaja del sistema propuesto es indiferente a la oclusión y rotación. Tiene una probabilidad de éxito de 9.24%, 67.85% y 44.90% para las máscaras pequeñas, medianas y grandes respectivamente.

La Figura 20 se observa que la primera fila la señal no es reconocida en la segunda y en la tercera captura debido a la neblina, en la segunda fila todas las señales con reconocidas.



Figura 20 Ejemplos de detección y reconocimiento

Fuente: Maldonado, 2007

Gómez (2010) propone un sistema que clasifica por el color de las señales, detección por forma y descomposición cromática y acromática. El SVM para el método de segmentación y para mejorar la velocidad usa lookup table (LUT). Obtiene un porcentaje de reconocimiento de 69.49% para el test y 78.29% para la validación.

También existen otros métodos basados en características del objeto, Fatin Zaklouta et al. (2011) que usan diferentes tamaños de histograma orientados de gradiente (HOG) como descriptor y distancia transformada para evaluar el rendimiento del árbol k-d que es la abreviatura de k- dimensional y los bosques aleatorios o en inglés "random forest" que son métodos más robustos a variaciones, el porcentaje de clasificación para "random forest" es 97.2% con HOG como descriptor.

Ellahyani et al. (2016) calcula las características de HOG de la imagen en el espacio de color HSI y luego lo combina con características de similitud local (LSS). El autor prefiere el uso de "random forest" como clasificador. El porcentaje de éxito en el reconocimiento es de 97.43% de la base de datos alemana GTSDb y 94.21 % para un sistema de 8-10 fotogramas/s

SIFT o Scale-invariant feature transform por sus siglas en inglés el cual es un algoritmo para extraer características relevantes de las imágenes para utilizarlo luego en las etapas de reconocimiento, fue usada por Sathish et al.(2016) en 8 videos grabados en la india, el rango del porcentaje de reconocimiento o clasificación fue de 75-100% con falsos positivos que no exceden el 2%. Hua et al.(2010) también uso SIFT combinado luego clasificarlo con SVM. LA base de

datos utilizada consta de 130 imágenes de 50x50 píxeles y hubo un porcentaje de precisión del 93% con un tiempo de ejecución de 0.098 ms por imagen

3.3 BASE DE DATOS DE SEÑALES DE TRÁNSITO

Hay una variedad de bases de datos de señales de tránsito a nivel mundial que brindan universidades, institutos o investigadores para libre descarga para el uso académico:

German Traffic Sign Benchmark (GTSRB)

Es una base de datos de Alemania del Instituto Neuroinformático para el uso en reconocimiento tiene alrededor de 50000 y más de 40 clases. Fue creada para la International Joint Conference on Neural Networks (IJCNN) del 2011. (<http://benchmark.ini.rub.de/?section=gtsrb&subsection=dataset>).



Figura 21 Tipos de señales de GTSRB

Swedish University Traffic Sign Dataset (STS Data set)

Esta base de datos pertenece a la Universidad de Linköping, la cual fue publicada en el 2011, contiene más de 20000 imágenes, aproximadamente 3488 señales de tránsito, para la recolección de esta información se recorrieron 350 km carreteras de Suecia, puede ser descargada desde la siguiente página (<http://www.cvl.isy.liu.se/research/datasets/traffic-signs-dataset/>), a diferencia de la base de datos anterior las imágenes no muestran solamente la señal de tránsito sino todo la escena recorrida.



Figura 22 Base de datos

KUL Belgium Traffic Sign Dataset (KUL Data set)

La base de datos pertenece a la Universidad Católica de Lovaina, tiene 62 clases de señales de las carreteras de Bélgica. Con un conjunto de entrenamiento de 4591 imágenes y de test de 2534 muestras.

(http://www.vision.ee.ethz.ch/~timofter/traffic_signs/)



Figura 23 Base de datos de Belgium TS

MASTIF dataset

Esta base de datos con señales croatas es totalmente gratuita para el uso académico, se creó para el Proyecto de investigación MASTIF. Cuenta con tres bases de datos (TS2009, TS2010, y TS2011) cada una con el año el cual se tomó la información cada señal fue tomada 4 -5 veces en diferentes distancias desde el auto. TS2009 contiene alrededor de 6000 señales recortadas, TS2010 alrededor de 3000 señales esta base de datos tiene la imagen completa más anotaciones

del tipo de señal que se encuentra en la imagen y el video y TS2011 tiene 1000 señales, cada base de datos tiene información adicional ya procesada.

(<http://www.zemris.fer.hr/~ssegvic/mastif/datasets.shtml>)



Figura 24 Imágenes de la misma señal a diferentes distancias

Data Set of Italian Traffic Signs (DITS data)

La siguiente base de datos es de la universidad Sapienza de Roma, esta database presenta dos innovaciones con respecto a otra base de datos, se divide en dos grupos y que han sido capturadas en la noche y otra contiene secuencia de imágenes con niebla. Está compuesta por 8048 imágenes para el entrenamiento y 1206 para el test. Con un total de 58 clases de señales con diferentes tamaños.

(<http://www.dis.uniroma1.it/~bloisi/ds/dits.html>)



Figura 25 Imágenes con diferentes características de luz y clima

LISA dataset

Es una base de datos creada por el laboratorio de automóviles seguros e inteligentes, en ella está disponible videos y frames con anotaciones de las señales de tránsito de carreteras de Estados Unidos, estas señales de tránsito tienen similitud a la señalización peruana.

En la Figura 26 muestra la estructura de la información que contiene la base de datos LISA, cada Track contiene la anotación de la señal, además de cada Track

son extraídos 30 frames y todos las señales en estos frames tienen información de la posición, tipo y alguna información adicional de metadata.

Contiene 47 señales de tránsito americana, 7855 anotaciones en 6610 frames, señales de tamaño 6x6 pixeles, la información fue adquirida de diferentes tipos de cámara y diferentes tamaño de imágenes, algunas en a color y otras a escala de grises. (<http://cvrr.ucsd.edu/LISA/lisa-traffic-sign-dataset.html>)

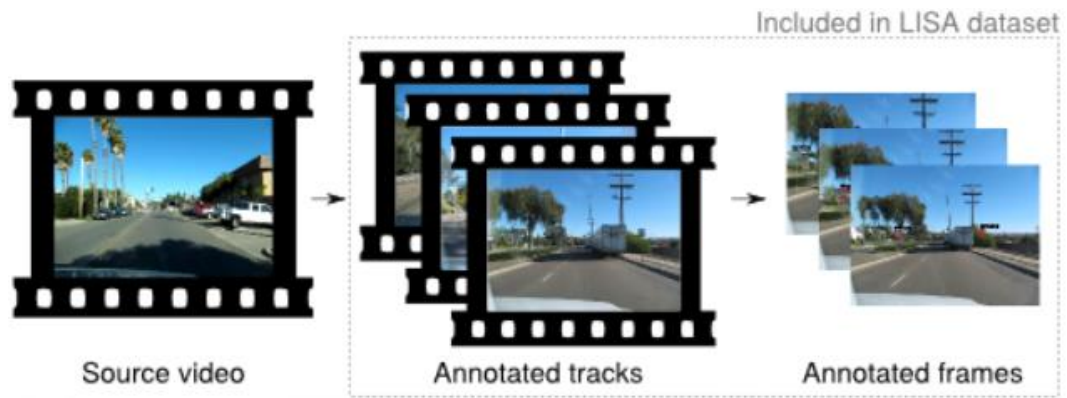


Figura 26 Estructura de la base de datos LIS

CAPÍTULO IV. ESTRATEGIA DE SOLUCIÓN

4.1 METODOLOGÍA

En la Figura 27 muestra un diagrama de flujo de la metodología desarrollada en la investigación, en este capítulo se describirá detalladamente cada uno de los pasos realizados en el mapa conceptual mostrado.

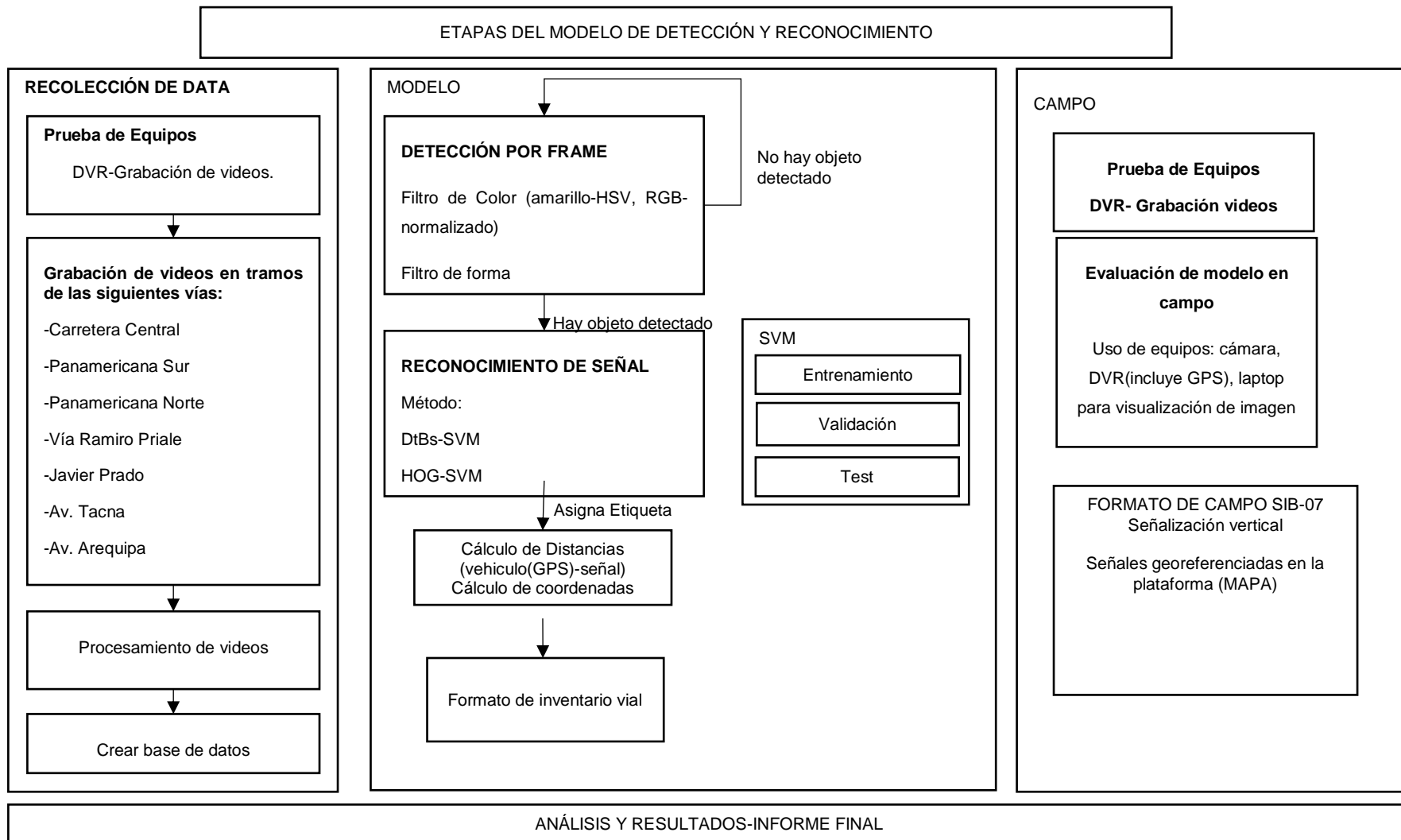


Figura 27 Diagrama de metodología propuesta
Fuente: elaboración propia

4.2 INVENTARIO DE SEÑALES

Para la etapa de validación del modelo del proyecto se realizó el inventario manual, para ello se conformó un equipo. En los siguientes apartados se describe la información de la vía como las características geométricas y la información recolectada en el inventario vial.

4.2.1 Información de la vía en estudio

Departamento/Región : Lima

Provincia : Lima

Distrito : San Martín de Porres

Longitud : 1 Km

Descripción : Esta vía se encuentra ubicada en el departamento de Lima, distrito de San Martín de Porres. El tramo en estudio es una vía urbana con denominación de vía local, con 3 carriles para cada sentido y un separador central, conexión entre la Av. Túpac Amaru con la Panamericana norte.



Figura 28 Tramo en estudio, Av. Eduardo Habich

Fuente: Google Earth

4.2.2 Campo de estudio del inventario vial

El campo de estudio del proyecto son las señales verticales preventivas, para ello se conformó una comisión de 2 personas con el objetivo de realizar el inventario del tramo seleccionado con fecha en agosto del 2017, fue necesario el uso de un GPS navegador para la medición de coordenadas y formatos que se encuentran en el Manual de Inventarios Viales del Ministerio de Transporte y comunicaciones para el inventario vial básico con código SIB-07, los formatos rellenos se encuentran en el anexo de campo. (Anexo 1)

En total fueron inventariadas 08 señales de tránsito preventivas a lo largo del tramo y de 4 categorías diferentes, las cuales se encuentran en buen estado.

4.3 RECOLECCIÓN DE DATA

El capítulo anterior enumera diversas bases de datos de señales de tránsito a nivel mundial que han sido utilizadas en diferentes investigaciones y se encuentran disponible en la red de manera gratuita con fines educativos, las bases de datos contienen información de las señales de tránsito según las normas europeas con excepción de la base LISAT la cual contiene información de señales de tránsito según la normativa norteamericana MUTCD (Manual Uniform Traffic Control Device por sus siglas en ingles), esta norma es usada como referencias para el Manual de Dispositivos de Tránsito del MTC. Sin embargo, para este proyecto se recolectó y creó una base de datos propia de acuerdo a los requerimientos del proyecto, y para usarla en futuras investigaciones. Primero se recolectó información de señales de tránsito externas al campo de estudio con cámaras de distintas características e información de aplicaciones de google como google Street View, para mayor diversidad en los datos. Las características que varían en las cámaras son la resolución de la imagen, profundidad de color, estabilizador de imagen, y distancia focal.

Un factor importante que se debe tener en cuenta para la recolección de información es la variación de color que depende de diversos factores, uno de ellos es la incidencia de luz en la señal, es decir el color captado por la cámara en la mañana no será el mismo el color captado en la tarde , por ejemplo la incidencia de la luz en la adquisición de videos; si las señales son grabados o captadas a contraluz los colores se verán afectados, esto sucede al inicio de la mañana y a final de la tarde.

La mayor parte de la información recolectada son imágenes extraídas de los videos obtenidos en la salida de campo programada con un vehículo y con una cámara de 2.4 MP, esta información se complementó con imágenes de captadas con el celular de 3MP, y cámara digital de 8 MP, fuera de las fechas indicadas.

A continuación, se brinda el cronograma de las vías que se recorrieron y el horario; en esta se mencionan todos los trayectos recorridos:

- Carretera Central km 0 hasta Km. 94 Matucana
- Panamericana Sur hasta Km. 31.5 peaje San Pedro
- Panamericana Norte hasta
- Vía Ramiro Priale
- Vía Evitamiento
- Javier Prado
- Av. Tacna
- Av. Arequipa
- Av. Angamos
- Av. Túpac Amaru
- Av. Paseo la República

Fue necesario solicitar el préstamo de una camioneta de la Facultad de Ingeniería Civil de la Universidad Nacional de Ingeniería para los recorridos, el siguiente cuadro se muestra el cronograma.

Cuadro 1 Cronograma de actividades para la recaudación de información

Salida	Horario
Jueves 11 de enero del 2018	13:00-17:00
Viernes 12 de enero del 2018	13:00 -17:00
Sábado 13 de enero del 2018	9:00 -13:00
Lunes 15 de enero del 2018	13:00 -17:00
Martes 16 de enero del 2018	9:00 -17:00

Fuente: elaboración propia

Los equipos usados en este proyecto fueron adquiridos gracias al financiamiento de proyectos de investigación del Instituto de Investigación de la Facultad de Ingeniería Civil (IIFIC), en el Cuadro 2 se describen las características de cada equipo:

Cuadro 2 Descripción de características técnicas de los equipos

Descripción de Equipos	
<p>DOMO MÓVIL CMOS ½.8" 2.4 MP 1080P 3.6mm IP67 & IK10 Audio IR:20m Resolución: HD 1080P • NTSC: 1984(H) x 1225(V) • Lente: 3.6mm • AGC, BLC, 3D-DNR, HLC, dWDR • Smart IR: 20m • Iluminación: 0.05Lux@F1.2, 0Lux IR On • 12VDC • Consumo: 3.3 Watt. • IP67 • IK10 • No fuente • Temp: -30° a +60°C • Angulo: 88° • Micrófono incorporado</p>	
<p>DVR MÓVIL 4CH (Análogo 960H / HDCVI) 04 Audio H.264 120 fps 720p / 1080p VGA 1 HDD 4G-WiFi-GPS • Velocidad de Grabación: 120fps@720P 60fps@1080P • Capacidad 1 Disco Duro SATA 2.5" (hasta 1TB) • Función Pentaplex (En vivo, graba, reproduce, backup, transmite) • Visualización por Smarthphone (iPhone, Android, Windows Phone) • 01 Entrada de red 10/100Mbps • Es inmune a Vibraciones</p>	
<p>DISCO DURO SATA 1TB - ESPECIAL PARA DVR/NVR</p>	
<p>WORKSTATION</p>	
<p>LAPTOP</p> <p>Laptop Asus N551JX Intel Core i7-4720HQ 2.6 GHz, RAM 16GB, HDD 1TB, Video NVidia GTX 950M 4GB, BluRay,15.6"Full HD, Win 10 Home</p>	

Fuente: elaboración propia

Para la salida de campo solo fue necesario realizar grabaciones sin información de coordenadas por ello solo se instaló el DVR y la cámara como se muestra en la Figura 29. La cámara es colocada en la parte superior del vehículo.



Figura 29 Instalación del equipo

Fuente: elaboración propia

En la Figura 30 se muestra los cables de energía para el funcionamiento de los equipos antes mencionados, el cable de red se utiliza para ver la información a tiempo real a través de un ordenador como se muestra en la Figura 31, y por último el cable de video para recolectar la información de la cámara y almacenarla en el DVR.

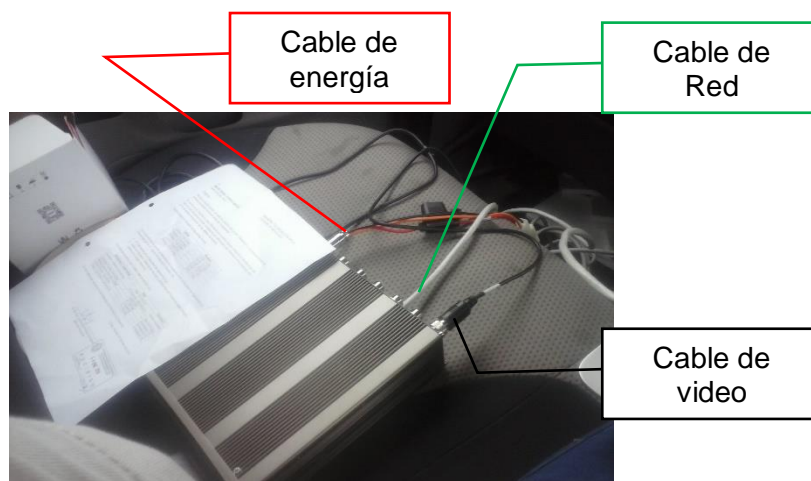


Figura 30 Detalle de la información

Fuente: elaboración propia



Figura 31 Visualización en la laptop

Fuente: elaboración propia

En la Figura 32 se observa una señal preventiva ubicada en la Av. Túpac Amaru cerca a la intersección con la Av. Honorio Delgado adquirida por la cámara de 2.4 M, en la parte inferior derecha se brinda información obtenida por el GPS sin embargo no fue necesario adquirir información de la posición ni velocidades, ya que no son requeridos para esta etapa del proyecto, adicional a esta información el DVR adquiere información de la fecha y hora del video.

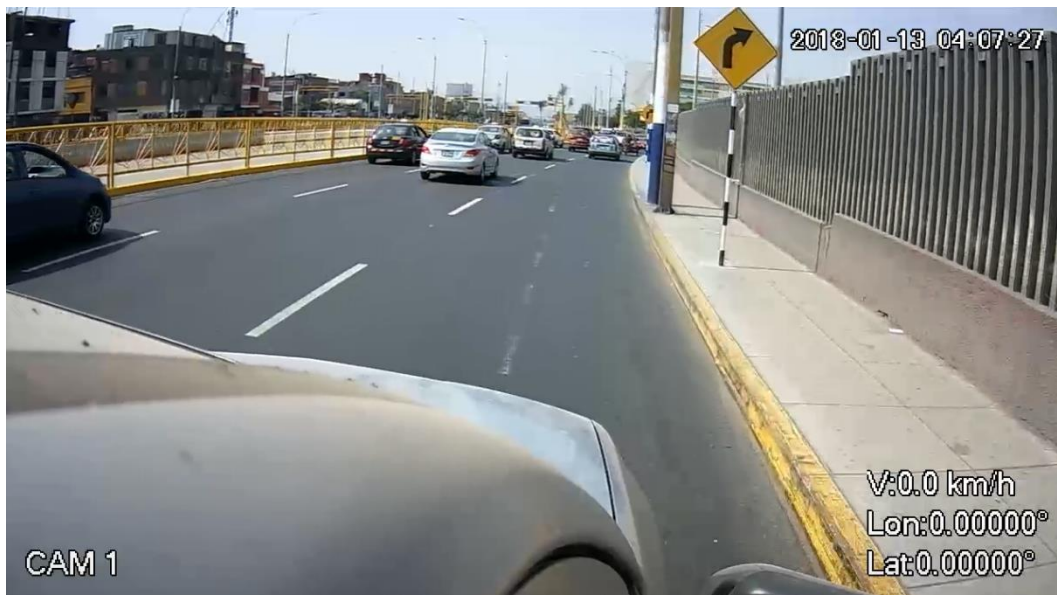


Figura 32 Frame del video

Fuente: elaboración propia

4.4 PROCESAMIENTO DE VIDEOS Y CREACIÓN DE BASE DE DATOS

La base de datos creada está conformada por imágenes de señales de tránsito preventivas fuera del área de estudio, la etapa de procesamiento de datos consta en extraer únicamente la región de interés de cada frame. En este acápite se describirá el procedimiento y las condiciones asumidas:

Primero es necesaria la extracción de los frames de cada video, como se puede observar en la Figura 33 por cada segundo de grabación se obtienen 30 frames o fotogramas, para el proyecto solo se extraerán 5 frames/seg teniendo en cuenta que la velocidad del automóvil es en promedio 50 km/h-80 km/h en vías como la panamericana sur, Ramiro Prialé y Vía Expresa, es decir cada 36-58 m será captada una imagen por la cámara .

Luego de la obtención de los frames o fotogramas, hubo frames que no contaban con información de señales de tránsito, por ello se realiza una inspección rápida y se descartan.

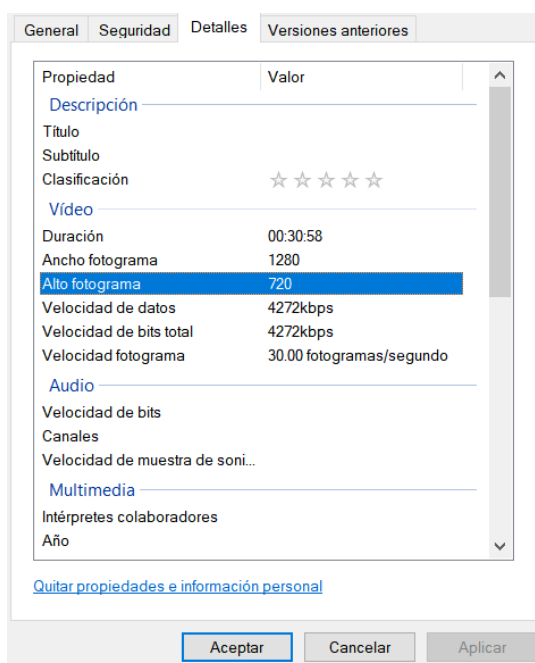


Figura 33 Detalles del video

Fuente: elaboración propia

El siguiente paso es la extracción manual de las regiones de interés, se creó una herramienta de recorte elaborada en el software MATLAB 2016, esta herramienta

simplifica el proceso se debe tener en cuenta que si la base de datos es amplia y diversa se mejora el proceso de reconocimiento. En este proceso se recolectaron 2062 imágenes de señales preventivas.

Recorte de imágenes

```
DirImag=uigetdir
Imag = datastore(DirImag, 'IncludeSubfolders', true, 'FileExtensions',...
    '.png','Type', 'image','LabelSource', 'foldernames');

for i=1:length(Imag.Files)
    temp1=readimage(Imag,i);
    w = waitforbuttonpress;
    imshow(temp1)
    if w == 0
        temp2=imcrop(temp1);
        imwrite(temp2,strcat(...
            'D:\FIC_UNI\2019-III\IIFIC\05_Data\Recorte/',num2str(i),'.png'))
    else
        continue
    end
end
end
```

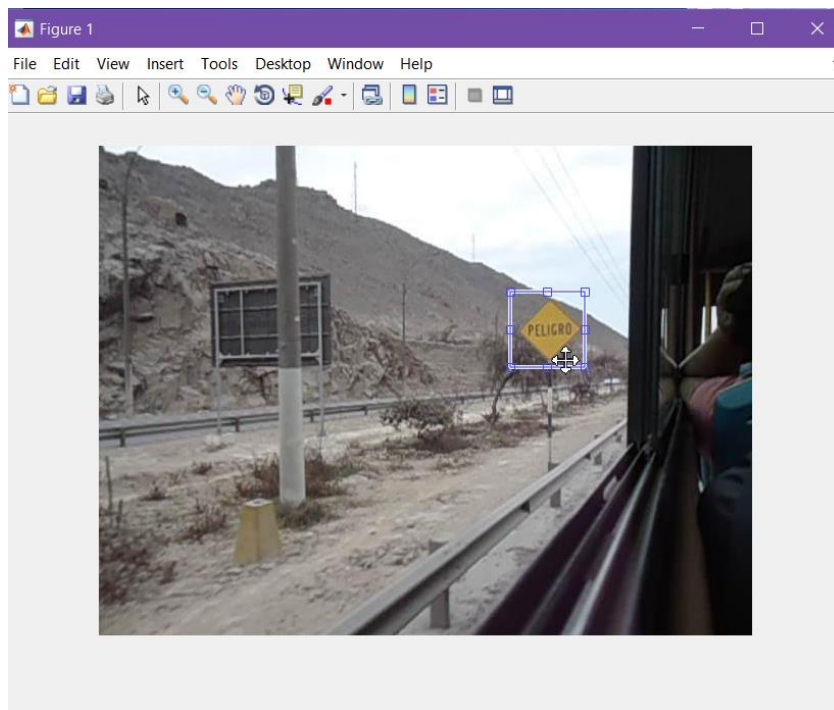


Figura 34 Herramienta para recorte

Fuente: Elaboración propia

Terminado el proceso de recorte de regiones de interés se categoriza y etiqueta cada una de las señales según su tipo, en el Cuadro 3 se muestra una lista de equivalencia entre el código utilizado en el proyecto y el código de señal preventiva según el manual de dispositivos de control de tránsito automotor para calles y carreteras del Ministerio de Transporte y Comunicaciones, también se coloca la cantidad de señales por tipo, estas STP en algunos casos son las mismas señales que varían en tamaño y en orientación.

Cuadro 3 Equivalencias de códigos

N	CÓDIGO	CÓDIGO DATA	CANTIDAD	NOMBRE
1	-	00001	1	Ferrocarril
2	P-15	00008	90	Señal intersección rotatoria
3	P-33	00011	373	Señal proximidad reductor de velocidad tipo resalto
4	P-46	00013	64	Señal "ciclovía" señal ciclistas en la vía
5	P-48	00014	414	Señal zona de presencia de peatones
6	P-49	00015	101	Señal zona escolar
7	-	00016	52	Despacio
8	P-2A	00017	274	Señal curva a la derecha
9	-	00018	1	Peligro
10	P-16A	00019	7	Señal incorporación de tránsito a la derecha
11	P-4A	00020	32	Señal curva y contra-curva a la derecha
12	-	00035	6	Zona de trabajo
13	P-16B	00070	36	Señal incorporación de tránsito a la izquierda
14	P-17C	00071	31	Señal reducción de calzada a lado izquierdo
15	P-2B	00072	248	Curva a la izquierda
16	-	00073	15	
17	P-18A, P17B	00074	5	Reducción del carril externo al lado derecho
18	P-10A	00075	19	Señal empalme en ángulo agudo a la derecha
19	-	00076	38	Peligro derrumbe
20	P-35	00077	32	Señal fuerte pendiente en descenso
21	P-41	00078	13	Señal túnel
22	P-4B	00079	13	Curva y contra-curva a la izquierda
23	P-5-1	00080	9	Camino sinuoso a la derecha
24	P-5-1A	00081	7	Camino sinuoso a la izquierda
25	P-5-2B	00082	7	Curva en "u" a la izquierda
26	P-5-2A	00083	8	Curva en "u" a la derecha
27	P-4B	00084	6	Señal curva y contra-curva a la izquierda

28	P-56	00085	24	Señal zona urbana
29	P-1B	00086	8	Señal curva pronunciada a la izquierda
30	P-10B	00087	4	Señal empalme en ángulo agudo a la izquierda
31	P-38	00088	3	Señal altura máxima permitida

Fuente: elaboración propia

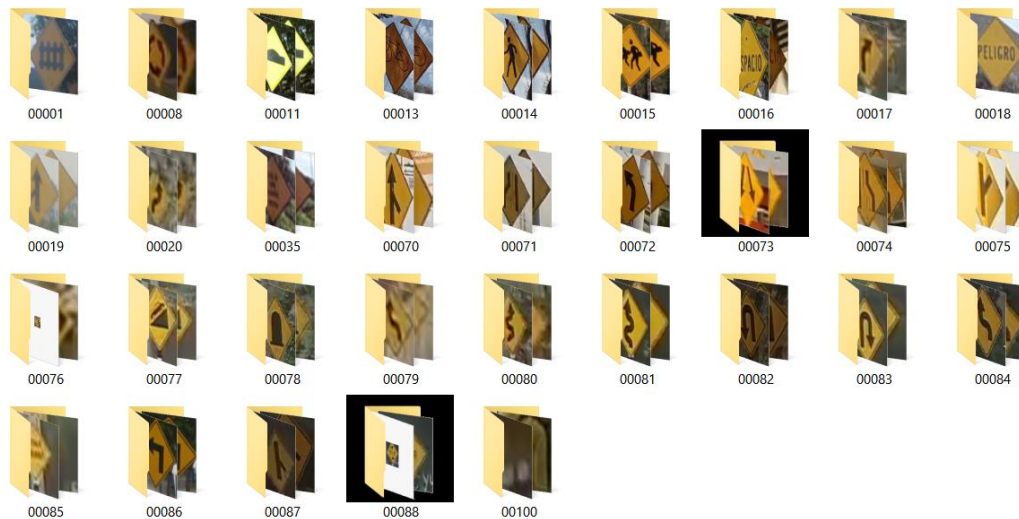


Figura 35 Base de datos de señales preventivas

Fuente: elaboración propia

Una vez ordenada la base de datos según código, la información está lista para para la etapa de reconocimiento, según la metodología mostrada.

4.5 PREPROCESADO DE DATA PARA ENTRENAMIENTO

Será necesario el pre procesamiento de las imágenes recortadas, a continuación, se explican algunos comandos utilizados.

- Medfilt2 (Matlab): realiza un filtrado medio de la imagen en dos dimensiones. Cada píxel de salida contiene el valor medio en un vecindario de 3 por 3 alrededor del píxel correspondiente de la imagen de entrada. Los valores de mediana para los puntos dentro de la mitad de la anchura del vecindario ($(m\ n)/2$) de los bordes pueden aparecer distorsionados. Nos sirve para eliminar ruido de sal y pimienta. En la Figura 36 la imagen de la izquierda tiene ruido sal y pimienta y en la imagen izquierda se aplica el comando Medfilt2.

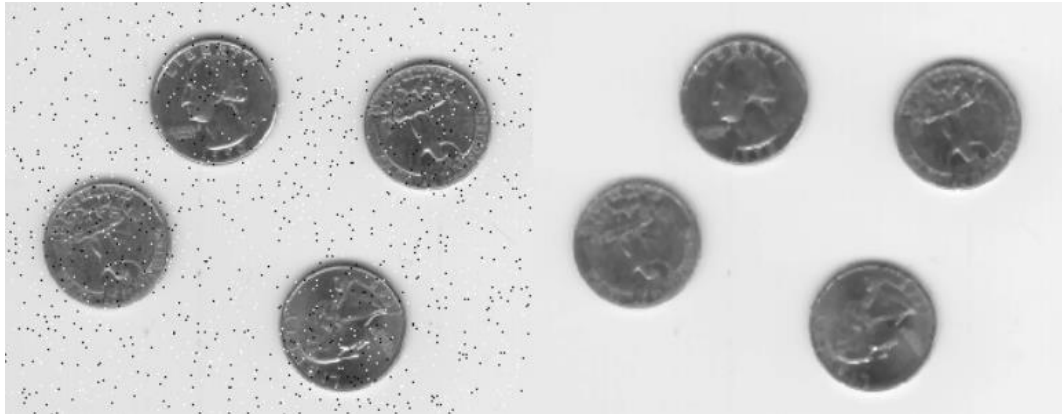


Figura 36 Ejemplo aplicativo del uso de Medfilt2

Fuente: mathworks

- `imgaussfilt` (Matlab) Filtrar la imagen con un filtro de Gauss con desviación estándar de 2. La imagen de la izquierda de la Figura 37 se muestra la imagen original y la imagen de la derecha se aplica el comando `imgaussfilt`.



Figura 37 Ejemplo aplicativo del uso `imgaussfilt`

Fuente: mathworks

4.6 ALGORITMO DE DETECCIÓN

La detección es la etapa donde se extrae los objetos de interés dentro de la imagen de manera automatizada que cumplen con los requerimiento o características que se indican dentro del proyecto, uno de los grandes problemas en esta etapa es obtener elementos que cumplen con las características, pero no son señales de tránsito a estos elementos se le llaman falsos positivos (FP). Para extraer las regiones de interés (ROI-Region of interest) se deben usar filtros

que pueden ser algoritmos basados en detección de forma o color, tener en cuenta que a menor ROIs detectados el costo computacional en la etapa de reconocimiento será menor y por tanto se optimizará el proceso. En esta investigación se consideró los filtros de color y forma debido a que las señales preventivas tienen como características principal el color amarillo y la forma de romboide dentro de los acápites siguiente se describe a detalle los filtros utilizados.

4.6.3 Filtro por color

Las señales de tránsito tienen un color característico indicado en el manual de dispositivos de control de tránsito automotor para calles y carreteras.

“Amarillo. Se utilizará como fondo para las señales de prevención”. Los colores especificados en el manual se definen en la base de coordenadas cromáticas –X e Y aplicadas sobre el diagrama cromático CIE 1931 mostrada en la Figura 38 que se encuentran dentro de la norma ASTM D-4956, vigente.

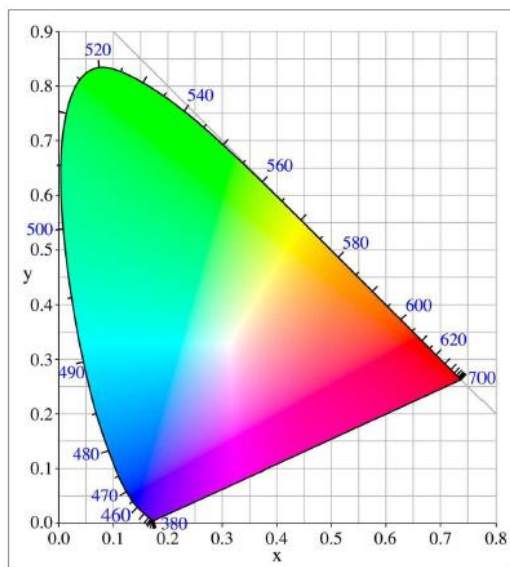


Figura 38 Diagrama Cromático CIE 1931(a)

Fuente: MTC, 2018

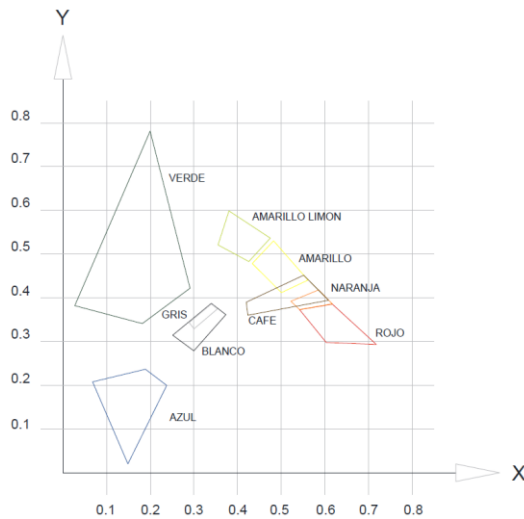


Figura 39 Diagrama Cromático CIE 1931 (b)

Fuente: MTC, 2018

Para la metodología propuesta en la tesis se compara el desempeño de los filtros usando el modelo de color el RGB normalizado y HSV. Para la obtención de los umbrales o rangos se extrae información de los pixeles y es elegido analizando los histogramas para cada canal en ambos modelos. En la Figura 40 muestra la concentración de valores del color característico de la ST en el modelo HSV, los valores para cada canal se encuentran concentrado en ciertos rangos que se muestra en la Tabla 2.

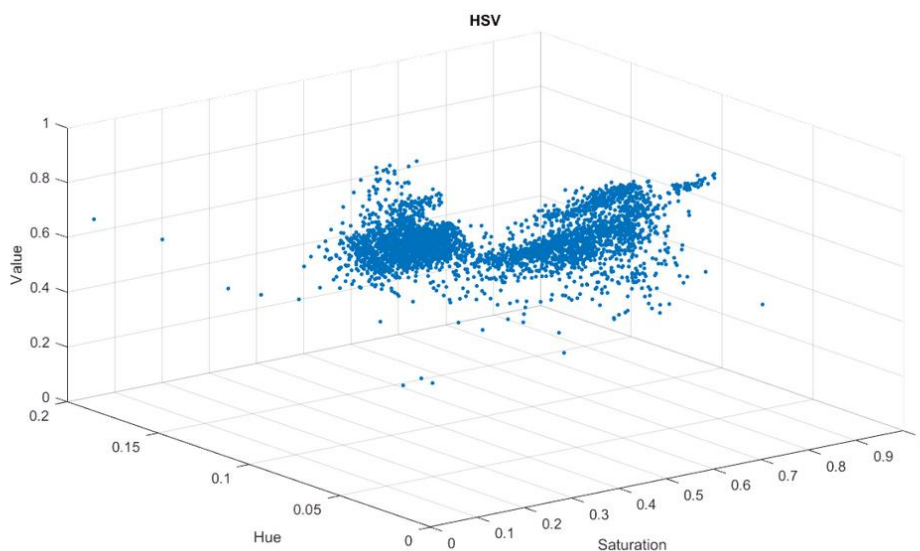


Figura 40 Concentración de valores HSV para color amarillo

Fuente: Elaboración propia

En la Tabla 2 se encuentran los valores de los rangos obtenidos para cada modelo de color.

Tabla 2 Umbrales de color

Descripción	RGB normalizado		Descripción	HSV	
	Min	Max		Min	Max
R'	0.6	0.85	Hue	0.05	0.17
G'	0.5	0.65	Saturation	0.25	-
B'	0	0.5	Value	0.25	-

Fuente: elaboración propia

Para reducir el ruido de algunas imágenes se realizan operaciones morfológicas con un elemento estructurante cuadrado para rellenar pequeños espacios y filtrar elementos muy pequeños conocidos como sal y pimienta en Figura 41 en la imagen superior se encuentra la imagen segmentada original y en la imagen inferior la imagen eliminando el ruido de tipo sal y pimienta. Como se observa se han rellenado los huecos dentro de los elementos.

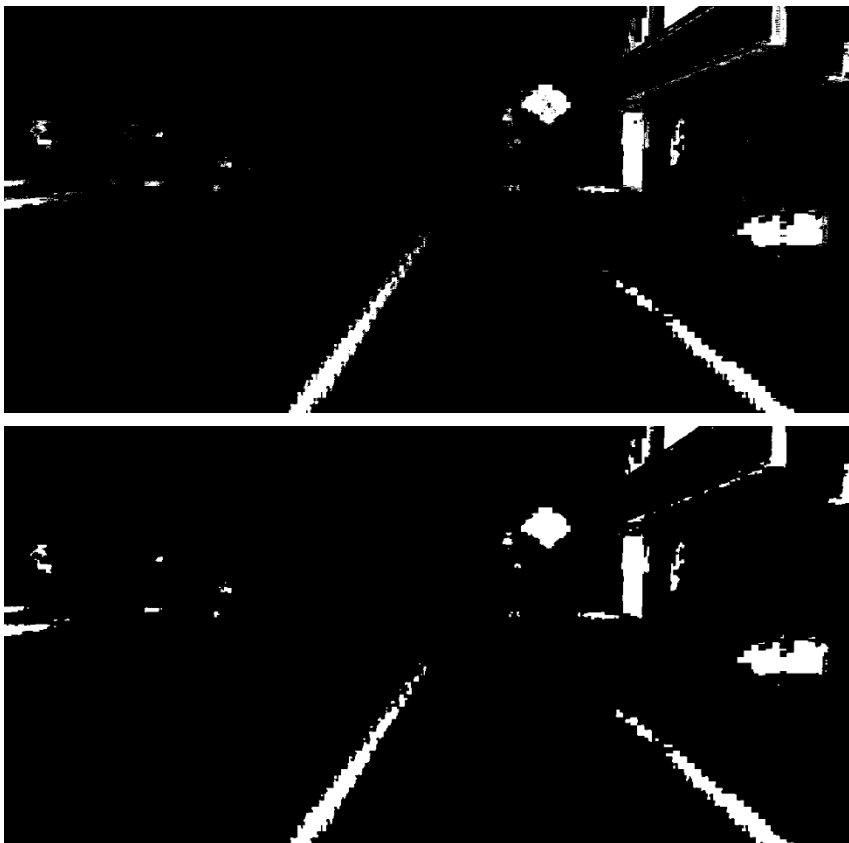


Figura 41 Aplicación de operaciones morfológicas.

Se debe tener en cuenta que el umbral de la Tabla 2 es para imágenes con el color característico de la STP, por ejemplo, la toma de datos debe realizarse con la luz diurna, no funciona para imágenes donde la señal se encuentre a contra luz como se observa en la Figura 42, es decir que el sol se encuentre justo detrás de la señal, en estos casos los pixeles no se encuentran en el umbral analizado.



Figura 42 Ejemplos de imágenes a contraluz

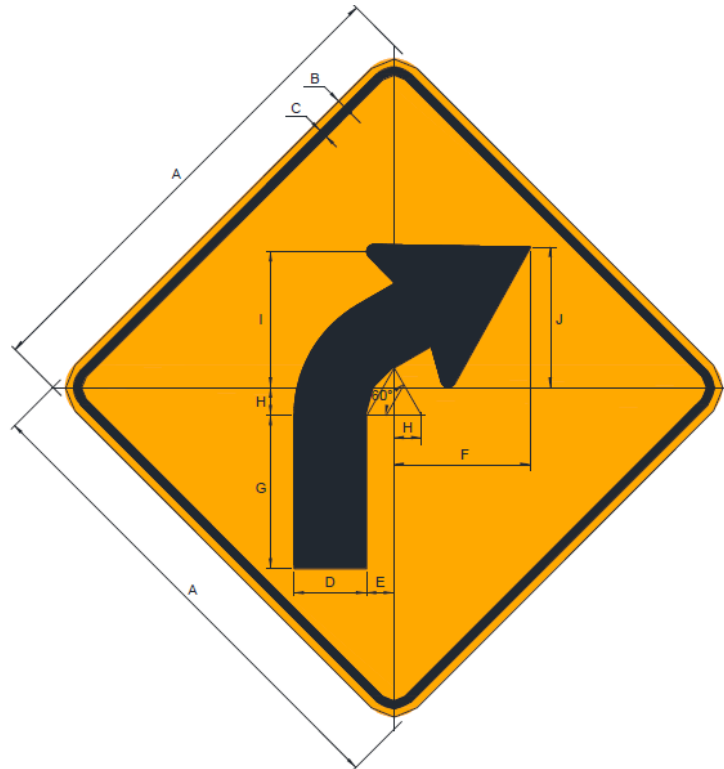
Fuente: elaboración propia

4.6.4 Filtro de forma

El manual del MTC describe las características geométricas de las señales y estas son determinadas a partir de la velocidad máxima a la que operan las vías, la información de la velocidad determina cual es la distancia mínima a la que debe ser vista y leída considerando 4 rangos de velocidades:

- ✓ Menor o igual a 50 km/h
- ✓ 60-70 km/h
- ✓ 80-90 km/h
- ✓ 100 a mayor km/h

En el Sección 2.13 del manual se encuentran las dimensiones de la señal considerando los rangos de velocidad mencionado anteriormente. En la Figura 43 se muestra la información de la señal curva a la derecha.



P-2A	VELOCIDAD (Km/h)	DIMENSIONES (milímetros)									
		A	B	C	D	E	F	G	H	I	J
450 x 450	Ciclovia	450.0	7.5	7.5	67.5	25.5	129.0	142.5	25.5	129.0	132.0
600 x 600	50 o menor	600.0	10.0	10.0	90.0	34.0	172.0	190.0	34.0	172.0	176.0
800 x 800	60 - 70	800.0	13.3	13.3	120.0	45.3	229.3	253.3	45.3	229.3	234.7
1000 x 1000	80 - 90	1000.0	16.7	16.7	150.0	56.7	286.7	316.7	56.7	286.7	293.3
1200 x 1200	100 o mayor	1200.0	20.0	20.0	180.0	68.0	344.0	380.0	68.0	344.0	352.0

Figura 43 Ejemplo de diagramación de la señal tipo curva la derecha

Fuente: MTC, 2018

En el paso anterior, segmentación con filtros de color, se extrae información tanto de señales como de objetos que son FP como postes de semáforo, barandas, bermas, líneas en el pavimento, carteles publicitario, con el mismo color que las STP, para evitar que los FP ingresen a la etapa de reconocimiento y en consecuencia aumente el tiempo de procesamiento de la metodología usada es necesario descartarlos en la primera etapa (detección), el siguiente paso será etiquetar los componentes encontrados para obtener las características geométricas tales como área y perímetro, para luego descartar los FP, se analiza las características geométricas de cada uno de los componentes, los parámetros descritos a continuación como criterios de filtrado se determinan empíricamente del conjunto de datos de imágenes.

- ✓ Elementos demasiado grandes

- ✓ Elementos muy pequeños
- ✓ Elementos en donde el área es mayor a 0.45 el área del boundingbox
- ✓ Donde la relación entre altura y el ancho sea mayor a 0.7 y menor a 1.3
- ✓ Donde el perímetro entre el área sea menor a 0.15 A continuación, se detalla de donde y porque se utiliza estos criterios:

Para el primer ítem se considera que el vehículo que va recorriendo la vía a cierta distancia de la berma y no se toma directamente la señal, al encontrar la señal a una distancia considerable el objeto en la imagen su área no conforma gran parte del frame.

Para el segundo ítem se usa la misma suposición del primer ítem.

Para el tercer ítem debemos tener en cuenta que la señal es un romboide y el boundingbox (BB) es la caja que encierra dicho elemento como se observa en la Figura 44, considerando el plano de la señal sin ninguna rotación del eje, el área del romboide simétrico es el 50% del área del BB que lo contiene. Se aplica un error del 5%, ya que al segmentar por color inicialmente no necesariamente se tomarán todos los pixeles que son parte de la señal de tránsito y el área será mucho menor.

Para el cuarto ítem debe tener en cuenta que no se mira la señal perpendicularmente a su plano, entonces la forma de la señal varía según el punto de visualización en su horizontal, si la visual es perpendicular al plano la relación sería de 1

Para el quinto ítem se realiza la operación matemática teniendo en cuenta que la señal tiene la forma romboide, el perímetro es $4L$, donde L es la longitud de uno de sus lados, el área es $D*d/2$ como el área va variando de acuerdo a la perspectiva de la cámara, es por ello que se tendrá una variación de la información del área, para ello de las imágenes recolectadas se toman varias muestras para hallar el rango que es 0.15.

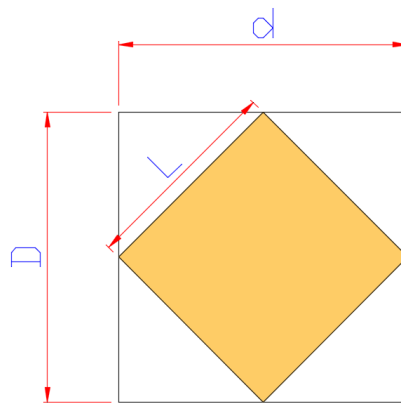


Figura 44 Características geométricas de la señal

Fuente: elaboración propia

Después del proceso de segmentación el área de la señal se extrae utilizando los parámetros del Bounding Box que son obtenidos y el área extraída pasa a la etapa de verificación por forma y luego a la etapa de reconocimiento considerando los vectores de característica o descriptores: el DtBs y HOG.



Figura 45 Ejemplo de extracción de la imagen

Fuente: elaboración propia

En la Figura 46 la imagen superior izquierda se observa el frame analizado, superior-derecha se encuentra el frame transformado al modelo de color HSV, Inferior-izquierda se aplicó la segmentación por color, la cual detecta varios FP dentro de la imagen como los bordes de la vereda, letreros, muros, etc. Luego de

aplicar el filtro de forma descrito líneas arriba se reduce en gran medida los FP por lo cual queda solo dos objetos dentro de la imagen inferior derecha.

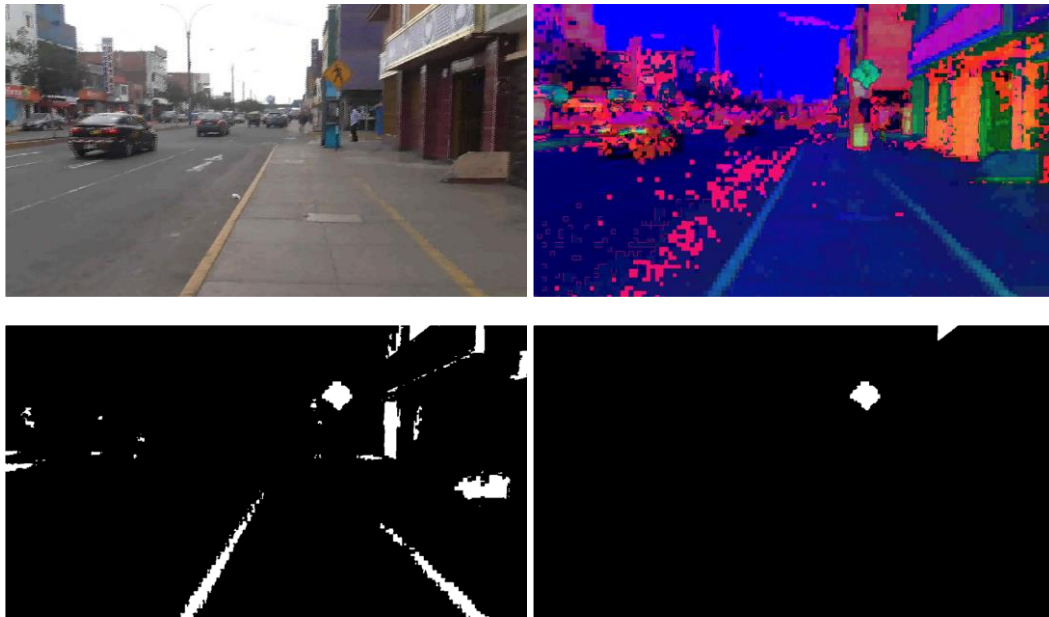


Figura 46 Etapa de detección del algoritmo propuesto

Fuente: elaboración propia

4.7 ALGORITMO DE RECONOCIMIENTO

La etapa de reconocimiento o clasificación es la etapa donde a los verdaderos positivos (VP) se les brindará una etiqueta para reconocer a qué tipo de señal pertenece, por ejemplo, señal peatonal, giro a la derecha, vuelta en U u otra señal. El modelo utilizado en este proyecto es la máquina de vectores soporte (SVM, por sus siglas en inglés Support Vector Machine) que es un clasificador binario, para la aplicación de este modelo es necesario contar con un descriptor, el cuál brinda información de las características de la imagen como cantidad de píxeles u otros, para la tesis se utilizó el descriptor llamado histograma de orientación gradiente (HOG, por sus siglas en inglés Histogram Orientated gradient). El código para la etapa de entrenamiento de la clasificación se encuentra Anexo 2, se realizó en el software de programación de Matlab (2016) usando las herramientas de matching Learning incluidas en la biblioteca de Matlab y el código base fue descargado de la librería de git-hub, acondicionando el código base a las características del proyecto, señales de tránsito preventivas peruanas.

4.7.5 Descriptor

Un descriptor es un identificador de características de un elemento, la idea principal es codificar la información del elemento en la menor cantidad de características representativas de tal manera que sea robusta ante problemas causados por los entornos naturales, cambio de iluminación, deterioro del objeto y oclusiones parciales.

Dentro de la metodología se describirán dos descriptores: ambos son descriptores de forma, el primero que clasifica los objetos por geometría, para el caso de señales, el segundo clasifica las señales por tipo de señal es decir analiza el símbolo dentro de la señal.

El primer descriptor utilizado es distancia para bordes DtBs (Maldonado,2009), es un método robusto contra la rotación y escala, extrae la distancia desde el bounding box hasta el borde externo del elemento. Las distancias de cada lado del bounding box se miden independientemente obteniendo 4 vectores en la Figura 47 (a) muestra la señal analizada, (b) se observa los objetos que cumple con los filtros de color o que tienen un color similar al de la señal, en la (c) se observa los objetos que cumplen con las características geométricas y en la (d) se observan las distancias mencionadas anteriormente donde D1, D2, D3 y D4 que son la parte inferior, superior, izquierda y derecha respectivamente luego se concadenan en un solo vector. Los datos utilizados para la etapa de entrenamiento se realiza una etapa adicional para eliminar señales que no tienen la forma adecuada se extrae el vector DtBs y cada 20 elementos se analiza la linealidad del vector, se considerará el error por mínimos cuadrados donde se considera que el coeficiente de regresión lineal R^2 como medida para evaluar lo antes expuesto para un vector si el R es menor a 0.7 no se considerará dentro de la base para la etapa de entrenamiento y la imagen es descartada, cada vector para el coeficiente de regresión lineal mayor a 0.7 si se considera, para finalmente usar SVM para la clasificación, clasificador simple, esta etapa es usada para reducir el tiempo de procesamiento en la etapa de reconocimiento del símbolo de la señal de tránsito. El descriptor sirve como filtro para los elementos que cumplan con las características de forma especificada (rombo). El clasificador es un clasificador binario, con dos etiquetas "1" para los elementos que cumplen con la forma y "-1" para elementos que no cumplan, la regresión lineal solo se usa para la etapa de entrenamiento del clasificador.

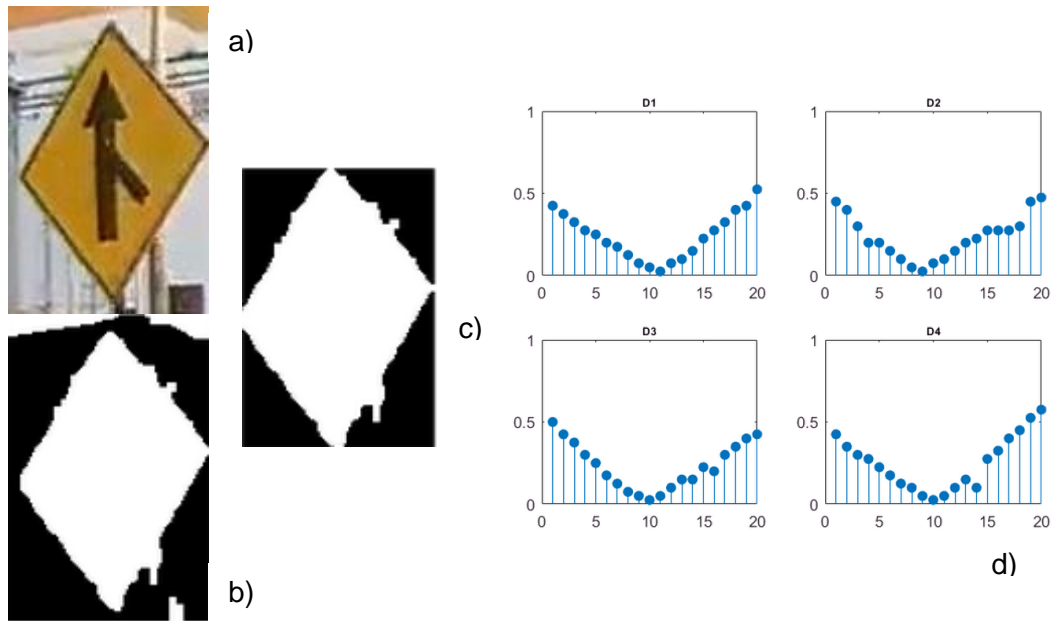


Figura 47 Distancia para el borde (Distance to border)

Fuente: elaboración propia

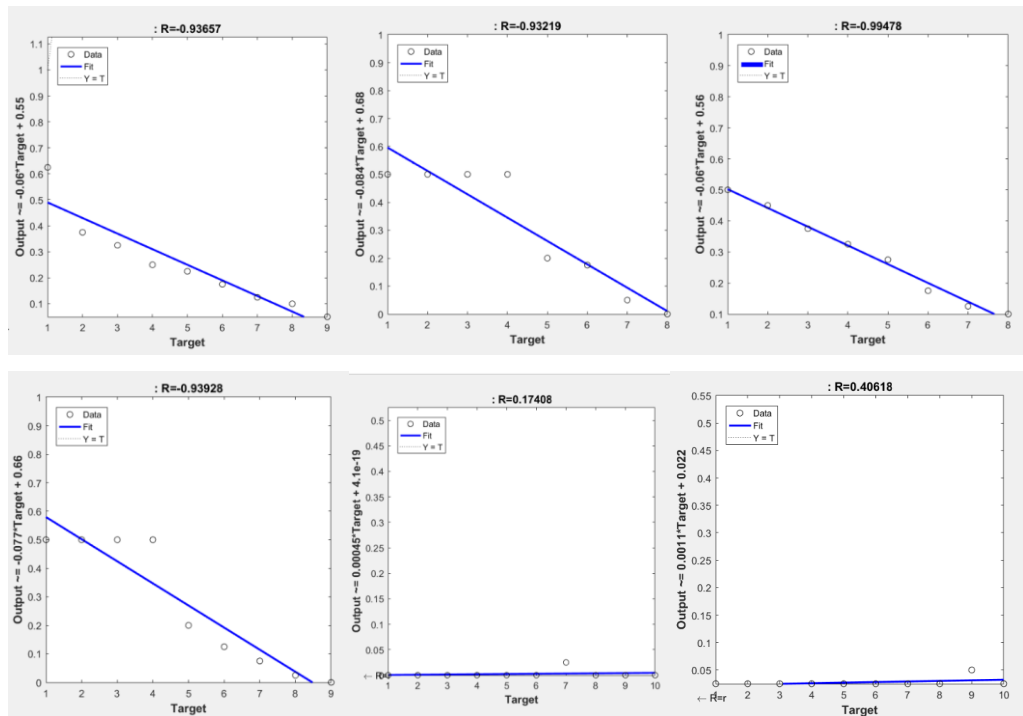


Figura 48 Regresión lineal para descriptor DtBs

Fuente: elaboración propia

Una vez realizada la verificación de forma mencionada anteriormente, se aplicará una máscara creada, para la eliminar el fondo y concentrar el análisis de

clasificación en el símbolo de la señal. La máscara que se muestra en la Figura 49, tiene dimensiones de 64x64 píxeles por lo que es necesario redimensionar el ROIs a la dimensión antes mencionada. El código para la clasificación de forma está en el Anexo 2 del presente documento.



Figura 49 Máscara de la señal preventiva

Fuente: elaboración propia

El segundo descriptor basado en la orientación de gradiente brinda información acerca de la forma del objeto que estamos estudiando y sobre todo el símbolo de la señal de tránsito, que es el objeto de interés para determinar el tipo de la señal, este descriptor es estable por su invariancia a los cambios de iluminación y color. Para el modelo se consideraron ciertas condiciones imágenes de 64x64 píxeles, celdas de 4x4 y bloques de 4x4 resultando un conjunto de 8100 atributos por imagen.

En la Figura 50 se puede observar gráficamente la orientación del gradiente, que describe la forma del letrero y del símbolo de la señal. El software Matlab tiene implementado este descriptor por lo que es necesario incluir las condiciones antes descritas.

```
[aHog,hogVisualization] = extractHOGFeatures(maskprev(aPatch{x,1},remask),'CellSize',[4 4]);
```



Figura 50 HOG

Fuente: elaboración propia

Una vez obtenido el descriptor que se encuentra en forma de vector de longitud, será necesario ingresar la información para la etapa de aprendizaje del descriptor, se consideraron los labels o etiquetas mostrados en el Cuadro 3. El código para la prueba del algoritmo se encuentra en el Anexo 2. La Figura 51 muestra la salida final.

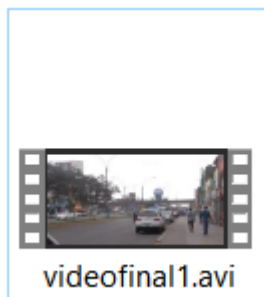


Figura 51 Video de salida

Fuente: elaboración propia

4.8 ALGORITMO DISTANCIA DE OBJETOS

El algoritmo utilizado sirve para adquirir la información de coordenadas geográficas de las señales y estimar una distancia aproximada entre la señal y el vehículo.

$$F = \frac{Px D}{W} \quad (51)$$

Donde:

F : Distancia focal

P : Ancho aparente

W : Ancho real del objeto

D : Distancia del objeto a la cámara

Primero es importante calcular la distancia focal de la cámara, ingresando datos de una distancia conocida, el P será el ancho en pixeles del objeto en la imagen.

Para el cálculo de la distancia D' :

$$D' = \frac{W}{Px F} \quad (52)$$

Donde:

D' : Distancia estimada de la cámara al objeto, distancia de la STP a la cámara

El W será considerado teniendo en cuenta el tipo de vía que será inventariada, por ejemplo para una de Velocidad de diseño (V_d) de 50 km/h a menos el ancho del elemento será de 600 mm como se observa en la Figura 43.

Para la recolección o levantamiento de información de las señales de tránsito se utilizará la cámara de video, y la información de las imágenes serán almacenadas en el DVR, el cual cuenta con un GPS.

Se utilizará la función [Velocidad Longitud Latitud]=Coordenadas(imagen) en el Anexo 2 - **Función de lectura de coordenadas** para extraer la información de las coordenadas geográficas y velocidad. A continuación, se muestra un ejemplo:



Figura 52 Imagen para el análisis

Fuente: elaboración propia

En la siguiente figura se muestra el extracto de la imagen que contiene datos de la latitud, de igual manera se extrae la información de la velocidad y longitud.

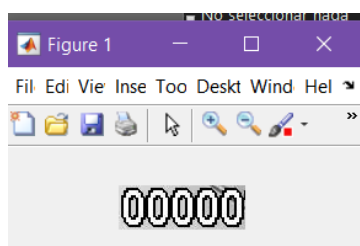


Figura 53 Data de latitud

Fuente: elaboración propia

Finalmente se extrae la información de la imagen y se guarda en variables

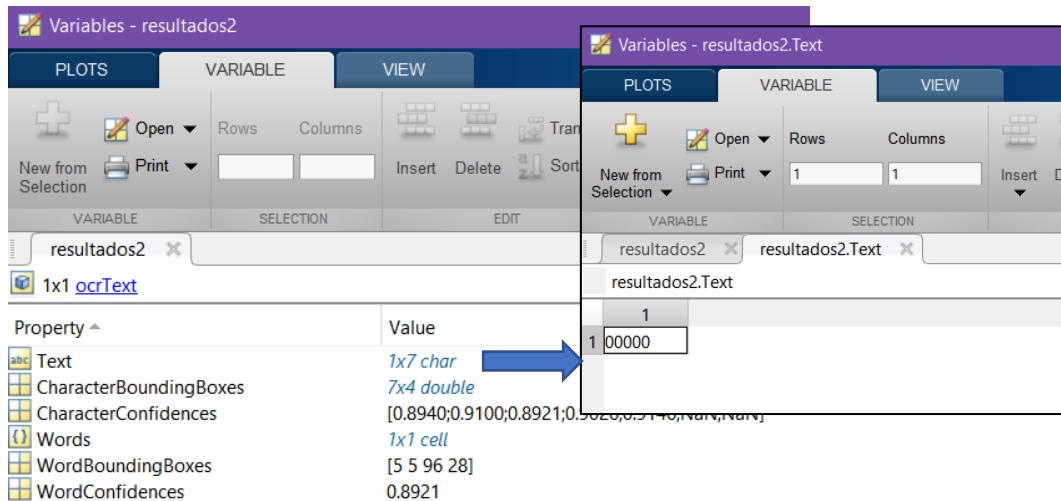


Figura 54 información de la imagen extraída

Fuente: elaboración propia

4.9 Mapa de señales

Finalmente, para una mejor visualización se presentará el mapa de señales detectadas y reconocidas de la Av. Eduardo de Habich. En el Anexo 3 se muestra el mapa de señales de tránsito. Cada señal brinda información del tipo, Calzada y material como se muestra en la Figura 55.

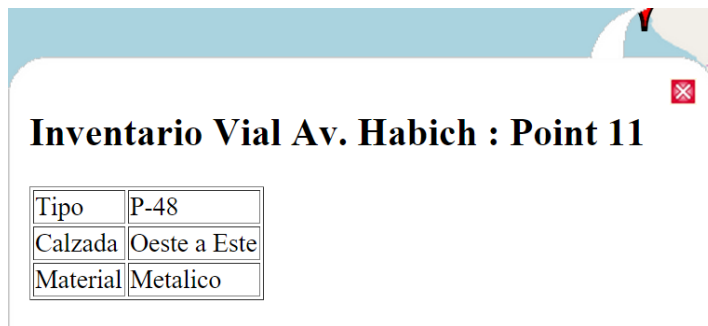


Figura 55 Datos de la señal de tránsito

Fuente: elaboración propia

CAPÍTULO V. VALIDACIÓN DE LA HIPÓTESIS

El código de Matlab es utilizado en fotogramas o frame como se describió en la metodología, en este capítulo se describe y se dan a conocer los resultados de la investigación

5.1 ANÁLISIS DE RESULTADOS

Como se describió anteriormente, se utilizaron dos filtros de color: HSV y normRGB para medir el performance de cada uno. El filtro de forma usado fue el mismo para ambos casos. El algoritmo es capaz de imprimir el código de proyecto de la señal en la imagen como se observa en las Figura 56 a la Figura 70 y también de extraer un video finalizado el proceso como se presenta en el siguiente acápite.

5.1.1 Imágenes detectadas por usando el filtro de color HSV

De la Figura 56 hasta Figura 62 se observa ejemplos de la aplicación del sistema de detección y reconocimiento para el modelo de color HSV. En la Figura 62 no se reconoce la señal de Ciclovía y señal de sobresalto P-33; y en la Figura 61 no es reconocida la señal P-49 A.



Figura 56 Aplicación del sistema de detección y reconocimiento a la señal peatonal 1 (P-48) con código de proyecto 00014 (HSV)

Fuente: elaboración propia



Figura 57 Aplicación del sistema de detección y reconocimiento a la señal peatonal 2 (P-48) con código proyecto 00014 (HSV)

Fuente: elaboración propia



Figura 58 Aplicación del sistema de detección y reconocimiento a la señal intersección rotatoria (P-15) con código proyecto 00008 (HSV)

Fuente: elaboración propia



Figura 59 Aplicación del sistema de detección y reconocimiento a la señal resalto 1 (P-33) con código proyecto 00011 (HSV)

Fuente: elaboración propia



Figura 60 Aplicación del sistema de detección y reconocimiento a la señal peatonal 3 (P-48) código proyecto 00014(HSV)

Fuente: elaboración propia



Figura 61 Aplicación del sistema de detección y reconocimiento a la señal Proximidad a cruce escolar (P-49 A) con código proyecto 00015 (HSV)

Fuente: elaboración propia

La Figura 62 muestra la señal de Ciclovía y STP P-33 que no fueron detectadas en el proceso.



Figura 62 Aplicación del sistema de detección y reconocimiento a la señal Ciclovía y señal de resalto 2 (P-33) (HSV)

Fuente: elaboración propia

5.1.2 Imágenes detectadas por usando el filtro de color normRGB

De la Figura 63 hasta Figura 70 se observa ejemplos de la aplicación del sistema de detección y reconocimiento para el modelo de color normRGB. La señal peatonal 1 P-48 (Figura 63), la señal Ciclovía y señal P-33 no fueron reconocidas.



Figura 63 Aplicación del sistema de detección y reconocimiento a la señal peatonal 1 (P-48) con código proyecto 00014 (normRGB)

Fuente: elaboración propia

Las demás SPTs si fueron reconocidas en el algoritmo considerando el filtro normRGB, se puede observar desde la Figura 64 hasta la Figura 68.



Figura 64 Aplicación del sistema de detección y reconocimiento a la señal peatonal 2 (P-48) con código proyecto 00014 (normRGB)

Fuente: elaboración propia



Figura 65 Aplicación del sistema de detección y reconocimiento a la señal intersección rotatoria (P-15) con código proyecto 00008 (normRGB)

Fuente: elaboración propia



Figura 66 Aplicación del sistema de detección y reconocimiento a la señal resalto 1 (P-33) con código proyecto 00011 (normRGB)

Fuente: elaboración propia



Figura 67 Aplicación del sistema de detección y reconocimiento a la señal peatonal 3 (P-48) con código proyecto 00014(normRGB)

Fuente: elaboración propia



Figura 68 Aplicación del sistema de detección y reconocimiento a la señal de Proximidad a cruce escolar 1 (P-49) con código proyecto 00015(normRGB)

Fuente: elaboración propia

Otros ejemplos fuera del campo de estudio, el algoritmo detecta perfectamente en otras condiciones como en un día soleado. La siguiente imagen pertenece a la carretera central.



Figura 69 Aplicación del sistema de detección y reconocimiento a la señal resalto (P-33) con código proyecto 00011

Fuente: elaboración propia



Figura 70 Aplicación del sistema de detección y reconocimiento a la señal (P-2A) con código proyecto 00017

Fuente: elaboración propia

5.1.3 Tamaño de la imagen

Las señales se encuentran por encima de la mitad del fotograma como se observa en las imágenes mostradas por lo que no sería necesaria la captura total de la imagen. Lo que reduce la cantidad de memoria de almacenamiento en el DVR.

5.1.4 Análisis métodos de filtros de color

En este proyecto analizamos los dos filtros de color seleccionados RGB normalizado y HSV. Como observamos en las figuras siguientes. Se analizó un frame, y se observa que el filtro HSV, detecta un FP que se encuentra dentro del rango utilizado. Tiene un tiempo de procesamiento $T=0.4115$.



Figura 71 Detección usando modelo HSV

Fuente: elaboración propia

Para el filtro de color RGB normalizado, solo se detectó la señal. Se observa que el tiempo de procesamiento es $T=0.3730$ s menor al filtro de color HSV.

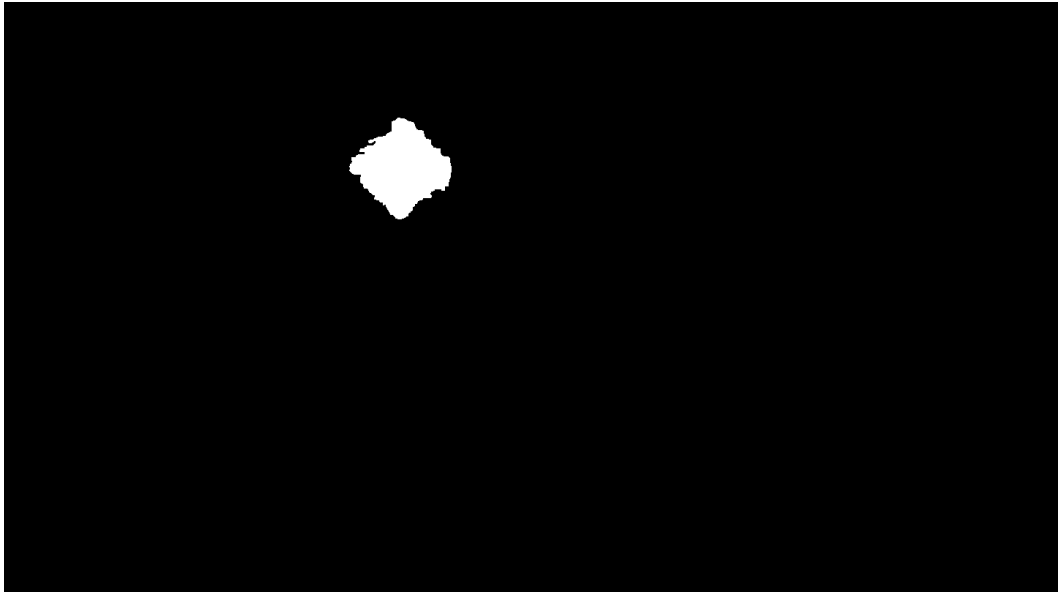


Figura 72 Detección usando modelo RGB normalizado

Fuente: elaboración propia

En el siguiente ítem se muestra detalladamente los tiempos utilizados para cada modelo de color.

5.1.5 Tiempo computacional

El tiempo que se demora en procesar la imagen es muy importante porque de ello depende si el algoritmo puede ser utilizado en tiempo real, en la siguiente tabla detallamos los tiempos antes mencionados, se observa que el modelo HSV tiene un mejor desempeño hasta la etapa de conversión donde el tiempo acumulado es menor, en las operaciones morfológicas los tiempos son similares. Sin embargo, en las siguientes etapas el tiempo acumulado del HSV sube considerablemente esto se debe a que en el filtro HSV los elementos analizados son dos a diferencia del RGB normalizado. En la tabla 3 se observa la importancia de eliminar los FP para mejorar el desempeño en el tiempo de procesamiento.

Tabla 3 Tiempo computacional etapa de Detección

Descripción	normRGB	HSV
Conversión a nuevo modelo de color	0.1345	0.0376

Segmentación de color	0.1489	0.1616
Operaciones Morfológico	0.2276	0.2777
Etiquetado	0.3391	0.3726
Segmentación de forma	0.3611	0.4001
Total	0.3730	0.4115

Fuente: Elaboración propia

5.1.6 Tiempo de procesamiento

Se han analizado alrededor de 6500 frames, el tiempo de procesamiento total varía en su mayoría entre 0-0.55 segundos como se observa en la Figura 77, solo para el modelo de color HSV, en los puntos más altos es donde se encuentra mayor cantidad de objetos detectados.

5.1.7 Porcentaje de éxito

Para ambos casos se observa una cantidad de similar de objetos detectados. Se logró detectar para cada caso 5 señales de tránsito de tipo preventivas siendo un porcentaje de éxito de 62.5%.

Tabla 4 Número de objetos detectados

Modelo de color	Objetos detectados	Objetos no detectados
normRGB	5	3
HSV	5	3

Fuente: elaboración propia

En la siguiente tabla se muestra que tipos de STP fueron detectadas y reconocidas según cada filtro de color.

Tabla 5 Elementos detectados según tipo

CÓDIGO DE SEÑAL SEGÚN MTC	CÓDIGO DE SEÑAL DEL PROYECTO	RGB	HVS
P-48	00014	No	Sí
P-48	00014	Sí	Sí
P-48	00014	Sí	Sí

CÓDIGO DE SEÑAL SEGÚN MTC	CÓDIGO DE SEÑAL DEL PROYECTO	RGB	HVS
P-49	00015	Sí	No
P-33	00011	Sí	Sí
P-33	00011	No	No
P-15	00008	Sí	Sí
Ciclovía	s/c	No	No

Fuente: elaboración propia

En ambos casos no se detectó ni se reconoció la STP señal de Ciclovía ni la segunda STP de código P-33 estas se encuentran lejos de la cámara.

5.1.8 Rendimiento del clasificador

El descriptor utilizado es el histograma de orientación gradiente, la cantidad de elementos predictores utilizados en el vector depende de la cantidad de pixeles utilizados en las celdas, para el proyecto se utilizó celdas de 4x4 y 8x8, lo que nos da un vector de 1764 y 8100 respectivamente. Además, se evaluó el rendimiento de cada predictor con el método cross validation o validación cruzada, la cual es una técnica para evaluar modelos de machine learning.

A continuación, se mostrará el rendimiento para el descriptor HOG, para ambos casos antes mencionados.

Para ambos se usa el método SVM uno a uno. Para el descriptor de celdas 8x8 el error del clasificador es ligeramente menor.

Property	Value
BinaryY	2062x528 double
CodingName	'onevsone'
Y	2062x1 categorical
X	2062x8100 single
RowsUsed	[]
W	2062x1 double
ModelParameters	1x1 ECOCParams
NumObservations	2062
PredictorNames	1x8100 cell
CategoricalPredictors	[]
ResponseName	'Y'
ExpandedPredictorNames	1x8100 cell
ClassNames	33x1 categorical
Prior	1x33 double
Cost	33x33 double
ScoreTransform	'none'
BinaryLearners	528x1 cell
BinaryLoss	'hinge'
CodingMatrix	33x528 double
LearnerWeights	1x528 double

Figura 73 Detalles e información del clasificador

Fuente: Elaboración propia(Matlab)

```
PREVSVMModel =
  classreg.learning.partition.ClassificationPartitionedECOC
  CrossValidatedModel: 'ECOC'
  PredictorNames: {1x8100 cell}
  ResponseName: 'Y'
  NumObservations: 2062
  KFold: 10
  Partition: [1x1 cvpartition]
  ClassNames: [1x33 categorical]
  ScoreTransform: 'none'

  Properties, Methods

EPrev =
  0.0824
```

Figura 74 Error del clasificador para celdas de 4x4

Fuente: Elaboración propia(Matlab)

Property	Value
BinaryY	2062x528 double
CodingName	'onevsone'
Y	2062x1 categorical
X	2062x8100 single
RowsUsed	[]
W	2062x1 double
ModelParameters	1x1 ECOCParams
NumObservations	2062
PredictorNames	1x8100 cell
CategoricalPredictors	[]
ResponseName	'Y'
ExpandedPredictorNames	1x8100 cell
ClassNames	33x1 categorical
Prior	1x33 double
Cost	33x33 double
ScoreTransform	'none'
BinaryLearners	528x1 cell
BinaryLoss	'hinge'
CodingMatrix	33x528 double
LearnerWeights	1x528 double

Figura 75 Detalles e información del clasificador

Fuente: Elaboración propia(Matlab)

```
PREVSVMModel =
classreg.learning.partition.ClassificationPartitionedECOC
  CrossValidatedModel: 'ECOC'
  PredictorNames: {1x1764 cell}
  ResponseName: 'Y'
  NumObservations: 2062
  KFold: 10
  Partition: [1x1 cvpartition]
  ClassNames: [1x33 categorical]
  ScoreTransform: 'none'

  Properties, Methods

EPrev =
    0.0616
```

Figura 76 Error del clasificador para celdas de 8x8

Fuente: Elaboración propia(Matlab)

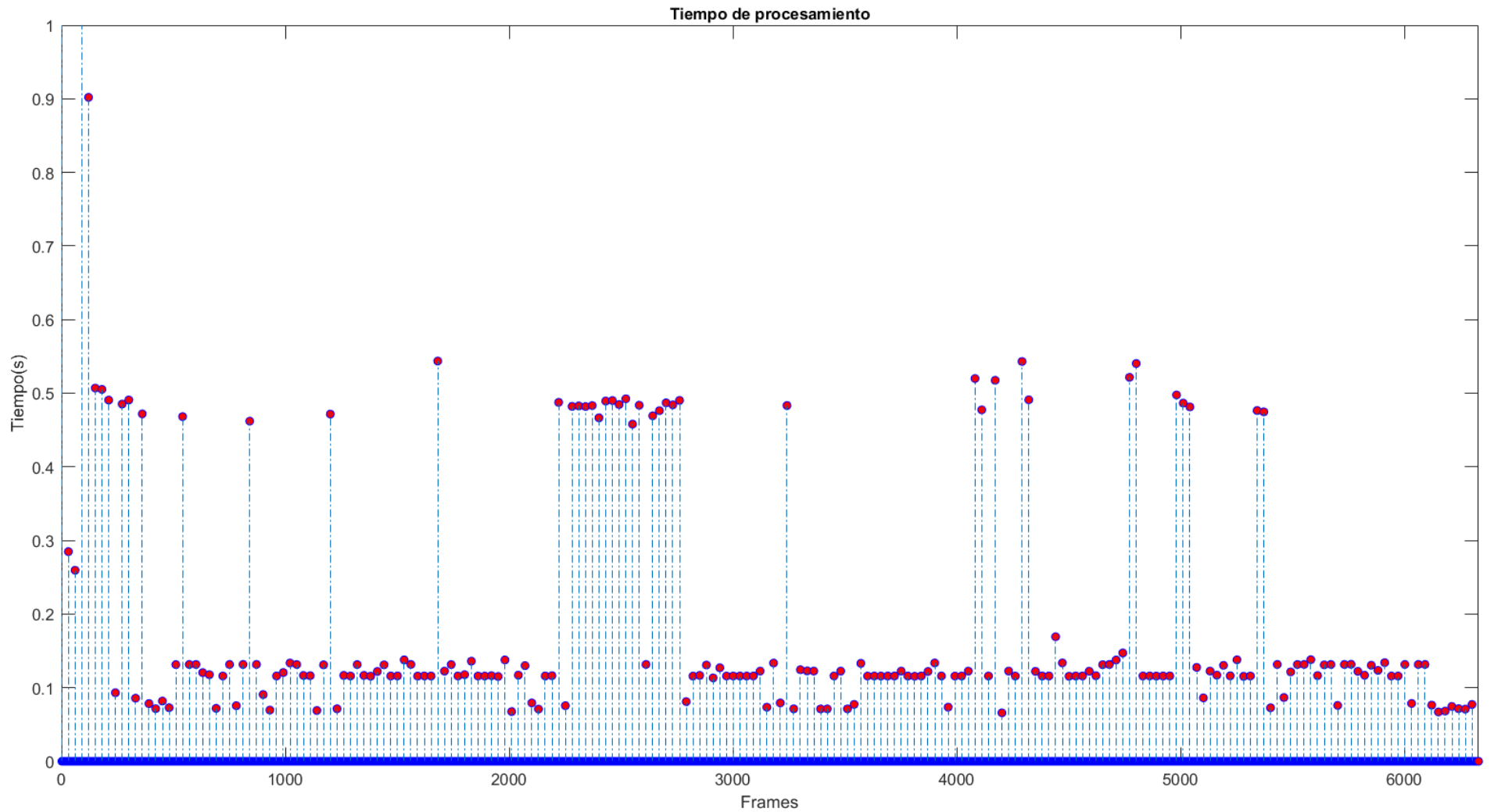


Figura 77 Tiempo de procesamiento de 6800 frames

CONCLUSIONES

- En esta tesis se crea una nueva base de datos para detectar y clasificar señales de tránsito. Comparado con otras bases de datos es una base de señales de tránsito peruanas de diversos tamaños que están ordenados según un código, se recolectó 31 clases diferentes de vías urbanas de la ciudad de Lima y carreteras de la provincia de Huarochirí.
- Las señales de tránsito son elementos con características similares como el color y forma predeterminadas lo que permite el uso de filtros de color y forma para la detección de las regiones de interés. Sin embargo, el algoritmo propuesto encuentra dificultades para detectar señales en mal estado como deterioro de la pintura, ya que no se encuentra en el rango especificado de intensidad del color.
- El procedimiento para cumplir con el objetivo del proyecto fue el uso de máquinas de soporte vectorial y visión por computador. que nos permitió realizar de manera automática la detección y reconocimiento de las señales de tránsito preventivas de la Av. Eduardo de Habich.
- El tiempo de procesamiento es una variable importante que permite determinar la velocidad del vehículo, teniendo como entrada la distancia de captura que se quiere evaluar, para realizar el levantamiento a tiempo real. En este proyecto obtenemos una velocidad entre 8.3- 20.8 km/h. La Av. Habich no es una vía muy congestionada, sin embargo, en vías rápidas sería difícil conducir a esa velocidad por lo que se considera hacer el procesamiento en gabinete.
- El uso del filtro de forma (distancia para el borde, Dtb) permite reducir en gran medida los falsos negativos, y por tanto el tiempo de procesamiento de la información de un fotograma.
- El uso de histograma de gradientes se usan dos tipos de tamaño de celda 4x4 y 8x8, el error generado por celdas de 8x8 es menor a las celdas generadas de 4x4, 0.0616 y 0.0824 por lo que se usó celdas de 8x8.
- El uso de una interfaz a tiempo real permitiría mejorar el proceso de toma de datos, de manera visual, y gestionar de mejor manera los datos recopilados en campo. Sin embargo, el tiempo de procesamiento de la metodología propuesta es muy lento para realizar un levantamiento en tiempo real, ya que en ambos casos propuestos tiene un tiempo de alrededor 0.5 s.

- En ambos modelos de color se obtiene un porcentaje de reconocidos de 62.5%, se debe tener en cuenta que la distancia desde el punto de captura hasta la señal es un factor importante, debido a la resolución de la imagen. En el caso de la investigación no se logró adquirir la información de 3 señales a lo largo de la avenida Habich una de ellas fue la señal de Ciclovía.
- Considerando que ambos modelos de color tienen como resultado el mismo porcentaje de detección, se tomó en cuenta la variable de tiempo de procesamiento como variable de comparación adicional, la segmentación del modelo de color HSV presentó una mejor performance respecto al modelo RGB normalizado hasta la etapa de conversión del modelo como se muestra en la Tabla 3, sin embargo el segundo filtro de color en su totalidad tiene un tiempo de procesamiento menor, por lo que se considera más óptimo.
- El algoritmo de reconocimiento permitirá generar una retroalimentación es decir a medida que se use permitirá ampliar la base de datos inicialmente propuesta.
- El algoritmo no detecta señales en horario nocturno o en contraluz debido al cambio de valores en cada componente según el modelo de color, los valores encontrados son para horario diurno.

RECOMENDACIONES

- Se recomienda el uso de un odómetro digital para el conteo del kilometraje de la vía que es información que se requiere para el inventario vial, que por cuestiones económicas no fue incluido dentro de este proyecto.
- El uso de otros algoritmos de reconocimiento profundo o Deep neural networks puede ser usado para siguientes investigaciones para optimizar el tiempo de procesamiento, además se debe tener en cuenta que para este tipo de algoritmos se requiere de una base de datos más amplia.
- Se recomienda ampliar la base de datos para las señales reglamentarias para realizar un inventario automatizado completo considerando tanto de señales preventivas como de señales reglamentarias.
- Es necesario investigar acerca de algoritmos que permitan detectar la señal y puedan brindar información acerca de su estado, según el manual de inventario viales existen 3 clasificaciones buena, regular y mala.

- Para problemas de clasificación múltiple es recomendable usar las máquinas de vectores soporte one-to-one ya que presentan mayor eficiencia que la de máquinas de vectores soporte.
- Para futuras investigaciones se pretende mejorar el aprendizaje reduciendo el error y reducir el tiempo de procesamiento para proponer un proceso a tiempo real.

BIBLIOGRAFÍA

- Abedin, M. Dhar, P. Deb, K., "Traffic Sign recognition using SURF: speeded up robust feature descriptor and artificial neural network classifier". IEEE 9th International conference on electrical and computer engineering (ICECE). Dhaka, 2016.
- Ayuque, K., "Diseño de sistema de clasificación de señales de tránsito vehicular utilizando redes neuronales convoluciones". Lima, 2016.
- Boser, B., Guyon, I. & Vapnik, V., "A training algorithm for optimal margin classifiers". D. Haussler: 5th Annual ACM workshop on COLT. ACM Press Pittsburg PA, 1992.
- Carmona, E., "Tutorial sobre Maquinas de vectores de clasificación (SVM)" Madrid, 2014.
- Cardarelli, E., Medici, P., Porta, P. & Ghisio, G., "Road signs shapes detection based on sobel phase analysis" IEEE 2009 IEEE Intelligent Vehicles Symposium. Xi'an, 2009.
- Ciresan, D., Meier, U. & Schmidhuber, J., "Multi-Column Deep Neural Network for Traffic Sign Classification". Computer Vision and Pattern Recognition (CVPR 2012). Providence RI, 2012.
- Cortes, C. & Vapnik, V., "Support vector machine". Kluwer Academic Publishers. Boston, 1995
- Dalal, N. & Triggs B., "Histograms of oriented gradients for human detection" IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). San Diego, CA 2005.
- Ellahyani, A., El Ansari, M. & El Jaafari I., "Traffic sign detection and recognition based on random forests". Applied Soft Computing. 2016.
- Escalera, A., Armingol, J. & Mata, M., "Traffic sign recognition and analysis for intelligent vehicles". Image and Vision Computing. 2003.
- Fang, C.Y., Chen, S.W., & Fuh C.S., "Road Sing Detection and Tracking". IEEE Transactions on intelligent transportation systems, 2003
- Fang, C.Y., Fuh, C.S., Yen P.S., Cherng, S & Chen, S.W., "An automatic road sign recognition system based on a computational model of human recognition processing", Computer vision and image understanding, 2004

- Greenhalgh, J. & Mirmehdi, M., "Real-Time Detection and Recognition of Road Traffic Signs". IEEE Transactions on intelligent transportation systems, 2012.
- Gomez, H., Maldonado, S., Gil, P. & Lafuente, S., "Goal evaluation of segmentation algorithms for traffic sign recognition". IEEE Transaction on intelligent transportation systems, 2010
- Gonzalez, R., & Woods, R., "Digital Image Processing". Edición N° 2, Editorial Prentice Hall, New Jersey, 2008.
- Hua, X. Zhua, X. Lia, D. Li. H., "Traffic sign recognition using scale invariant feature transform and SVM". A special joint symposium of ISPRS Technical Commission IV & AutoCarto in conjunction with ASPRS/CaGIS 2010 Fall Specialty Conference. Orlando, Florida, 2010.
- Isasi, P. & Galván, I., "Redes de Neuronas Artificiales. Un enfoque práctico". Pearson Prentice Hall. Madrid, 2003
- Karungaru, S. Nakano, CH. & Fukumi, M., "Road Traffic Signs Recognition Using Genetic Algorithms and Neural Networks". International Journal of Machine Learning and Computing, 2013.
- King, L., Li, A. & Kah, S., "New hybrid technique for traffic sign recognition", International symposium on intelligent signal processing and communications systems, 2009
- Liu, H., Liu, D. & Xin, J., "Real-time recognition of road traffic sign in motion image based on genetic algorithm". Proceedings of the First International Conference on Machine Learning and Cyberneucs, Beijing, 2002
- López, A., Valveny, E. & Vanrell, M., "Detección de objetos". Curso virtual de la Universidad Autónoma de Barcelona:
<https://www.coursera.org/learn/deteccion-objetos/home/welcome>. Barcelona, 2015
- Mahammad, H., Safat, W. & Tan, P., "Traffic Sign Classification based on Neural Network for Advance Driver Assistance System". Universiti Kebangsaan Malaysia, Bangi, 2014.
- Maldonado, S., "Sistema de detección y reconocimiento de señalización en carretera mediante técnicas de procesamiento digital de imagen e inteligencia artificial". (Tesis Doctoral). Universidad de Alcalá, Alcalá de Henares, 2014.

- Maldonado, S., Lafuente, S., & Gil, P., "Road-Sign Detection and Recognition Based on Support Vector Machines". IEEE Transactions on intelligent transportation systems, 2007
- Mathworks, Learning Machine. Web:
https://la.mathworks.com/solutions/machine-learning.html?s_tid=hp_brand_machine
- Ministerio de Transporte y comunicaciones, "Manual de Dispositivos de control del tránsito automotor para calles y carreteras". Edición Mayo. Lima, 2018
- Ministerio de Transporte y comunicaciones, "Manual de Inventarios Viales". Lima, 2016
- Miura, J., Kanda, T., Nakatani, S. & Shirai, Y., "An active vision system for on-line traffic sign recognition". IEICE transactions on information and systems. 2000
- Pérez, C., "Emotions Recognition using Deep Learning". Trabajo final de Master. Universidad Politécnica de Valencia, Valencia, 2016.
- Qin, F., Fang B. & Zhao H., "Traffic sign segmentation and recognition in scene images". Chinese conference on pattern recognition (CCPR). Chongqing, 2010.
- Rocha, C. & Escorcía, J., "Sistema de Visión artificial para la detección y el reconocimiento de señales de tráfico basado en redes neuronales". LACCEI. Arequipa, 2010
- Saadna, Y. & Behloul, A., "An overview of traffic sign detection and classification method". Springer-Verlag, Londres, 2017.
- Sathish, P., Bharathi, D., "Automatic road sign detection and recognition based on SIFT feature matching algorithm". Springer: Proceedings of international conference on soft computing systems. India, 2016.
- Schmit, R., "Las tecnologías de construcción de carreteras han sido diseñadas y desarrolladas para mejorar la eficiencia y la producción" Research and Innovation Seminars. Piura, 2015
- Sermanet, P. & LeCun, Y., "Traffic sign recognition with multiscale convolutional networks". IEEE The 2011 international joint conference on neural networks (IJCNN). San Jose, 2011.
- Vélez, J., Moreno, A. & Sánchez, A., "Visión por computador", S.L.–DYKINSON, Madrid, 2003.

- Vitabile, S., Pollacia, G., Pilato, G. & Sorbello, F., "Road signs recognition using a dynamic pixel aggregation technique in the HSV color space". IEEE Proceeding 11th International conference on image Analysis and processing. Palermo, 2001
- Wang, G., Ren, G., Wu, Z., Zhao, Y. & Jiang, L., "A robust, coarse to fine traffic sign detection method". IJCNN. Dallas, 2013
- Wang, Y. & Yuan, B., "A novel approach for human face detection from color images under complex background", Pattern Recognition, 2001.
- Wu, Y., Liu, Y., Li, J., Lin, H. & Hu, X., "Traffic Sign Detection based on convolutional neural networks". Proceeding of international joint conference on neural networks. Dallas, Texas, 2013
- Yakimov, P., "Traffic signs detection using tracking with prediction". International conference on E-business and telecommunications. Colmar, 2015
- Youssef, A., Albani, D., Nardi, D. & Bloisi DD., "Fast traffic sign recognition using color segmentation and deep convolutional networks". Springer International conference on advanced concepts for intelligent vision systems. 2016
- Zaklouta, F., Stanciulescu, B. & Hamdoun, O., "Traffic sign classification using kd trees and random forest". IEEE The 2011 international joint conference on neural networks (IJCNN). San Jose, 2011.
- Zhu, Z., Liang, D., Zhang, S., Huang, X., Li, B. & Hu, S., "Traffic-sign detection and classification in the wild". IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, 2016.

REFERENCIAS BIBLIOGRÁFICAS

ANEXOS

Anexo 1: Formatos de campo

SIC-22 SEÑALIZACIÓN VERTICAL

CARRETERA	
TRAMO	
UBICACIÓN	

ASISTENTE DE CAMPO	
DÍA Y FECHA	

RUTA	CALZADA	UBICACIÓN DE INICIO		UBICACIÓN DE FIN		LADO	TIPO	MATERIAL	CÓDIGO	CONDICIÓN	OBSERVACIONES
		CÓDIGO PR	DISTANCIA	CÓDIGO PR	DISTANCIA						

SIB-07 SEÑALIZACIÓN VERTICAL

CARRETERA	
TRAMO	
UBICACIÓN	

ASISTENTE DE CAMPO	
--------------------	--

COD_UBIGEO	RUTA	SEÑAL	CLASIFICACIÓN	UBICACIÓN	LADO	SOPORTE	MATERIAL	LATITUD	LONGITUD	ALTITUD	FECHA

Anexo 2: Código base

Distancia al borde

```
function Dts1=bordes(img)
tic
    if nnz(img)~=0
        img=imresize(img,[40 40]);
        T=size(img);
        bin=round(T(1)/20);
        D1=zeros(1,T(1)/2);
        D2=zeros(1,T(1)/2);
        D3=zeros(1,T(1)/2);
        D4=zeros(1,T(1)/2);
        for i=1:T(1)/2
            if 0== nnz(img(i*2,:))
                D1(1,i)=0.5;
                D2(1,i)=0.5;
            else
                D1(1,i)=(min(find(img(i*2,:))))/T(1);
                D2(1,i)=1-(max(find(img(i*2,:))))/T(1);
            end
            if 0== nnz(img(:,i*2))
                D1(1,i)=0.5;
                D2(1,i)=0.5;
            else
                D3(1,i)=(min(find(img(:,i*2))))/T(1);
                D4(1,i)=1-(max(find(img(:,i*2))))/T(1);
            end
        end
        Dts1=horzcat(D1,D2,D3,D4);
    end
end
```

Entrenamiento del clasificador

Limpiar la ventana de trabajo

```
clear all;
```

```
close all;
```

```
clc;
```

Ingreso de datos para el modelo

```
% Obtener las imágenes de las carpetas
```

```
cd ..; cd codigo;
```

```
trainingDirPrev =uigetdir
```

```
imtrainSetPrev = datastore(trainingDirPrev, 'IncludeSubfolders',
true, 'FileExtensions', '.png', 'Type', 'image', 'LabelSource',
'foldernames');%cambiar extension de archivo

load mask_prev.mat

mask_prev=imresize(mask_prev,[64 64]);

% Informacion para HOG

cellSize=4;

Clasificacion usando HOG-SVM para señales preventivas

nimagesPrev = numel(imtrainSetPrev.Files);

trainFeaturesPrev = zeros(nimagesPrev,DescriptorSize, 'single');

%Redimensionar las imagenes

for i=1:nimagesPrev

    image = readimage(imtrainSetPrev, i);

    image = rgb2gray(image);

    image = medfilt2(image, [3 3]);

    image = imresize(image, [64 64]);

    image= maskprev(image,mask_prev);

    [Hog,hogVisualization] =
extractHOGFeatures(image, 'CellSize',[cellSize cellSize]);

    trainFeaturesPrev(i,:) = Hog(:)';

end

% Etiquetar las imágenes

trainLabelsPrev= imtrainSetPrev.Labels;

disp('class');

%SVM

tic

classifierPrev = fitcecoc(trainFeaturesPrev, trainLabelsPrev);
```

```
save('modelPrevcell4.mat','classifierPrev');  
  
toc  
  
Clasificación por forma  
  
tic  
  
load Datatotal.mat  
  
load Datatotalrec.mat  
  
[ah aw]=size(Totalborde);  
  
[bh bw]=size(Totalborderec);  
  
Label=ones(ah,1);  
  
Labelrec=zeros(bh,1);  
  
trainShape=[Totalborde;Totalborderec];  
  
trainTshape=[Label; Labelrec];  
  
classifiershape = fitcecoc(trainShape, trainTshape);  
  
save('modelshape.mat','classifiershape');  
  
toc
```

Test del clasificador

```
% Cargar la data Entrenada  
load isClassifier.mat  
load modelPrevcell8.mat  
load modelshape.mat  
load mask_prev.mat  
remask=imresize(mask_prev,[64 64]);  
  
warning('off');  
  
load vidCounter.mat  
cd ..;  
cd Salida/;  
vidWriter=VideoWriter(sprintf('videofinal%d.avi',vidCounter));
```

```
vidWriter.FrameRate = 30;
cd ../codigo;

open(vidWriter);
for i=1:30:6331
    tic
    cd ../

    image1=imread(strcat('D:\FIC_UNI\2017-III\00 PROYECTO IIFIC\05_DATA
CAMPO\01_DATA CAMPO HABICH VIDEOS Y RECORTES\FRAMES
HABICH',sprintf('/%00d.jpg',i)));%colocar la ruta de la carpeta donde se
encuentran las imágenes
    image1=imcrop(image1,[0 0 750 300]);

    tempImage=zeros(size(image1,1),size(image1,2),size(image1,3));
    image1 = imgaussfilt(image1,0.3);
    cd ../codigo\;

    globalStore={};
    globalCount=1;

    image=double(image1);

    [a,b,shap]=yellowsegm(image(:,:,1),image(:,:,2),image(:,:,3));

    %[a,shap]=hsvyellowv1(image1);
    if size(a,2)~=0

        aPatch=get_patch(image1,a);

        for x=1:size(aPatch,1)
            [aHog,hogVisualization] =
extractHOGFeatures(maskprev(aPatch{x,1},remask), 'CellSize',[8 8]);
            DtB=bordes(shap{x});
            [predictedLabels,scoreb]=
signPredict(aHog,DtB,classifiershape,classifienPrev);

            for labelB={'00014','00011', '00015', '00008'}
```

```
        labelA=labelB{1};
    if predictedLabels == labelA
        globalStore{globalCount,1}=a{x}.BoundingBox;
        globalStore{globalCount,2}=labelB;
        globalCount=globalCount+1;
        break;
    end
end

end
end

figure(1);
imshow(image1);hold on;
for index=1:size(globalStore,1)
rectangle('Position',globalStore{index,1},'EdgeColor','r','linewidth',2);
    text(globalStore{index,1}(1)-5,...
        globalStore{index,1}(2)-10,...
        globalStore{index,2}(1), 'FontSize', 20, 'FontWeight',
'Bold',...
        'Color', 'g');
    end
writeVideo(vidWriter,getframe);
hold off;
pause(1/500);
timerec=toc;
timet(i)=timerec;
end

close(vidWriter);
close all;
vidCounter=vidCounter+1;
save('vidCounter.mat','vidCounter');
```

Función de filtro de color (Modelo: HSV)

```
function [new_s, Shape]= hsvyellowv1(image1)
HSV=rgb2hsv(image1);
```

```
Thresh=HSV(:,:,1)>=0.05 & HSV(:,:,1)<=0.17 & HSV(:,:,2)>=0.25 &
HSV(:,:,3)>=0.25;
bw_2_mejorada = imdilate(Thresh, ones(3));
normred_mejorada=imfill(bw_2_mejorada,'holes');
bw_2_mejorada = imerode(normred_mejorada, ones(3));% cierre morfológico
bw_2_mejorada = imerode(bw_2_mejorada, ones(3));
bw_2_mejorada = imdilate(bw_2_mejorada, ones(3));% apertura morfológica
imshow(bw_2_mejorada, 'Border', 'tight')
Etiquetas=bwlabel(bw_2_mejorada);
s = regionprops(bw_2_mejorada,{'Area', 'Perimeter', 'BoundingBox'});
new_s = cell(1,0);j = 1;
n=size(HSV);
for i=1:length(s)
    if s(i).BoundingBox(4)/s(i).BoundingBox(3) > 0.7 ...
        && s(i).BoundingBox(4)/s(i).BoundingBox(3) < 1.5 ...
        && s(i).Perimeter/s(i).Area<0.15...
        new_s{j}= s(i);
        im_test = (Etiquetas == i);
        Shape{j}=imcrop(im_test,s(i).BoundingBox);
        j = j+1;
    end
end
if j==1
Shape=0;
end
end
```

Función de filtro de color (Modelo: RGB normalizado)

```
function [new_s,mask, Shape]= yellowsegm2(r,g,b)
magnituedeimage=sqrt(r.*r+g.*g+b.*b);
normred=r./magnituedeimage;
normgreen=g./magnituedeimage;
normblue=b./magnituedeimage;
thresh=normred>0.6 & normred<0.85 & normgreen>0.5 & normgreen<0.65 &
normblue<0.5;
```



```
imshow(thresh)
pause(0.5)
bw_2_mejorada = imerode(thresh, ones(4));
normred_mejorada=imfill(bw_2_mejorada, 'holes');
bw_2_mejorada = imdilate(normred_mejorada, ones(4));% cierre
morfológico
bw_2_mejorada = imdilate(bw_2_mejorada, ones(4));
bw_2_mejorada = imerode(bw_2_mejorada, ones(4));% apertura morfológica
figure(2)
imshow(bw_2_mejorada)
Etiquetas=bwlabel(bw_2_mejorada);
s = regionprops(bw_2_mejorada,{'Area','Perimeter','BoundingBox'});
mask = zeros(size(Etiquetas));
new_s = cell(1,0);j = 1;
n=size(normred);
for i=1:length(s)
    if s(i).BoundingBox(4)/s(i).BoundingBox(3) > 0.7 ...
        && s(i).BoundingBox(4)/s(i).BoundingBox(3) < 1.3 ...
        && s(i).Perimeter/s(i).Area<0.15...
        && s(i).Area/(
s(i).BoundingBox(4)*s(i).BoundingBox(3))>0.45 ...
        && 15*s(i).BoundingBox(4)*s(i).BoundingBox(3)<n(1)*n(2)...
        && s(i).BoundingBox(4)*s(i).BoundingBox(3)>0.001*n(1)*n(2)
        new_s{j}= s(i);
        im_test = (Etiquetas == i);
        Shape{j}=imcrop(im_test,s(i).BoundingBox);
        mask = mask+ im_test;
        imshow(mask)
        j = j+1;
    end
end
if j==1
Shape=0;
end
end
```

Función de lectura de coordenadas

```
function [Velocidad Longitud Latitud]=Coordenadas(imagen)
im1=rgb2gray(imagen);
V=imcrop(im1,[1075 557 55 40]);
```

```
Lon=imcrop(im1,[1113 605 132 35]);
Lat=imcrop(im1,[1127 653 102 35]);
% Velocidad
Vbin=V>251;
Vbin = imdilate(Vbin, ones(3));
Vbin = imerode(Vbin, ones(3));
% Longitud
Lonbin=Lon>251;
Lonbin = imdilate(Lonbin, ones(3));
Lonbin = imerode(Lonbin, ones(3));
%Latitud
Latbin=Lat>251;
Latbin = imdilate(Latbin, ones(3));
Latbin = imerode(Latbin, ones(3));
%Extraer texto
resultados = ocr (Vbin, 'TextLayout' , 'Block' );
resultados1 = ocr (Lonbin, 'TextLayout' , 'Block' );
resultados2 = ocr (Latbin, 'TextLayout' , 'Block' );

Velocidad= str2num(resultados.Text);
Longitud= str2num(resultados1.Text);
Latitud= str2num(resultados2.Text);
end
```

Código para mostrar señales en el mapa

```
Calzada=''
Material=''
Tipo=''
latitud=''
longitud=''
p = geopoint(latitud, longitud,'Tipo', Tipo,'Calzada',Calzada,'Material',
Material);
attribspec = makeattribspec(p);
webmap('Open Street Map', 'WrapAround', false);
wmmarker(p,'Description', attribspec,'OverlayName', 'Inventario Vial Av.
Habich')
```

Anexo 3: Mapa de salida

Mapa web-Matlab

