



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

"IMPLEMENTACIÓN DE UN PROTOTIPO PARA DETECCIÓN DE
ATASCOS VEHICULARES EN LOS TUNELES DE LA CIUDAD DE
GUAYAQUIL"

INFORME DE MATERIA DE GRADUACIÓN

Previa a la obtención del Título de:

INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES

Presentado por:

Israel Pablo Jiménez Paute

Roberto Fernando Jara Bravo

GUAYAQUIL - ECUADOR

2009

AGRADECIMIENTO

A Dios, quien cada día nos brinda sus bendiciones y a quien le debemos todo.

A las personas cuyos sacrificios, sueños y deseos nos han apoyado a seguir adelante.

DEDICATORIA

A Dios, nuestros Padres, hermanos y amigos cuya confianza se ha fortalecido con el paso del tiempo.

DECLARACIÓN EXPRESA

“La responsabilidad del contenido de este trabajo nos corresponde exclusivamente; y el patrimonio intelectual del mismo a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL”

(Reglamento de exámenes y títulos profesionales de la ESPOL)

Israel Pablo Jiménez Paute

Roberto Fernando Jara Bravo

TRIBUNAL DE SUSTENTACIÓN

Ing. Patricia Chávez

PROFESOR DE LA MATERIA

Ing. Juan Carlos Avilés

PROFESOR DELEGADO DEL DECANO

RESUMEN

La detección o reconocimiento de objetos es una aplicación del procesamiento digital de imágenes. Existen factores que resultan influyentes en la buena detección de un objeto, como la calidad de la imagen, la iluminación que tuvo el objeto o elementos que dificultan la visibilidad de la imagen.

En el presente proyecto se realiza la detección de vehículos, y se verifica si los vehículos se encuentran detenidos. Para esto se ha realizado un túnel a escala, con lo necesario para una simulación básica de tráfico.

Las imágenes obtenidas por la cámara que se ha colocado en el túnel son procesadas con la ayuda del programa matemático Matlab. Utilizando detección de bordes y operaciones morfológicas se logra la extracción de objetos para poder ser comparados y determinar si ha habido movimiento de los automóviles en el túnel.

Mediante comparación por correlación y por posición se determina si un objeto es el mismo o si se ha desplazado. Al tratarse del mismo vehículo se envía una señal indicando que ha habido un atasco. Si el vehículo se ha desplazado es posible obtener la velocidad aproximada a la que se movía.

INDICE GENERAL

INTRODUCCIÓN.....	1
CAPITULO 1	
1. DESCRIPCIÓN DEL PROBLEMA.....	2
1.1. Antecedentes.....	2
1.2. Limitantes.....	3
1.3. Consideraciones.....	4
2. ELABORACIÓN DE TÚNEL A ESCALA.....	7
2.1. Iluminación.....	7
2.1.1. Conexión de DAQ.....	8
2.1.1.1. Luminarias Internas.....	10
2.1.1.2. Paneles Externos.....	11
2.2. Medidas y Escalas.....	12
2.3. Ubicación de la Cámara.....	13
2.3.1. Ángulo de Visión.....	13
2.3.2. Relación pixeles-metros.....	14
2.3.2.1. Relación $dist_x$ vs α_x	15
2.3.2.2. Relación distancia x vs fila x	16
3. DETECCIÓN DE IMÁGENES.....	19

3.1. Generalidades.....	19
3.2. Multi-frame.....	19
3.3. Imagen de Intensidad.....	20
3.4. Detección de Bordes.....	20
3.5. Análisis Morfológico.....	22
3.5.1. Dilatación.....	23
3.5.2. Erosión.....	23
3.5.3. Apertura y Cierre.....	24
3.6. Etiquetado de Imágenes.....	26
3.7. Segmentación.....	27
4. COMPARACIÓN DE IMÁGENES.....	29
4.1. Generalidades.....	29
4.2. Comparación por correlación.....	30
4.3. Comparación por posición.....	32
4.4. Detección de atasco.....	33
5. ESTADÍSTICAS.....	34
CONCLUSIONES.....	39
RECOMENDACIONES.....	41
ANEXOS.....	42
BIBLIOGRAFÍA.....	56

INDICE DE IMÁGENES

Imagen 2.1. Vista Interna del Túnel.....	8
Imagen 2.2. Tarjeta DAQ.....	9
Imagen 2.3. Interface de la Tarjeta DAQ.....	9
Imagen 2.4. Control de iluminación interior del túnel.....	10
Imagen 2.5. Luminaria Interna 12 V AC.....	11
Imagen 2.6. Control de los paneles de entrada.....	11
Imagen 2.7. Túnel a escala.....	12
Imagen 2.8. Medidas del Túnel.....	12
Imagen 2.9. Medidas de Automóviles.....	13
Imagen 2.10. Visión de la cámara.....	14
Imagen 2.11. Ubicación de la cámara.....	15
Imagen 2.12. Captura bajo una resolución $m \times n$	17
Imagen 2.13. Relación α_x vs fila x	17
Imagen 3.1. Detección de bordes.....	21
Imagen 3.2. Formas básicas.....	22
Imagen 3.3. Dilatación de X con el elemento estructurante Y.....	23
Imagen 3.4. Erosión de X con el elemento estructurante Y.....	24
Imagen 3.5. Operación de Apertura de X con el elemento estructurante Y... ..	25
Imagen 3.6. Operación de Cierre de X con el elemento estructurante Y.....	25

Imagen 3.7. Imagen Binaria para ser etiquetada.....	27
Imagen 3.8. Identificación de objetos.....	28
Imagen 4.1. Diagrama del Sistema.....	29
Imagen 4.2. Correlación de A con h.....	31

INTRODUCCIÓN

Los túneles tienen gran importancia al permitir comunicación entre dos puntos separados por un obstáculo. Además de brindar organización urbana, aportan a la economía debido a la disminución de recorrido que brindan.

Los túneles al comunicar tramos deben estar disponibles al tráfico en todo momento, resultando puntos críticos. Es de gran ayuda saber la disponibilidad de un túnel. Si de alguna forma, se supiera de antemano si un área está congestionada o no es viable, el conductor tendría un conocimiento previo para tomar una decisión, esperar a que el túnel sea viable o escoger otra ruta para llegar a su destino.

Hemos realizado un prototipo a escala basados en los túneles de la ciudad de Guayaquil. Con ayuda de una cámara y procesamiento de imágenes mediante el programa matemático Matlab pretendemos realizar un detector de atascos.

Realizando una comparación de imágenes, verifica que el tráfico siga fluyendo y no se mantenga en el mismo lugar, si estuviese en el mismo lugar querrá indicar que existe una obstrucción.

CAPITULO I

1. DESCRIPCIÓN DEL PROBLEMA

1.1. Antecedentes

Los accidentes de tránsito suceden con mucha frecuencia, en la ciudad hasta septiembre de este año se han registrado 4539 accidentes [7]. Además de los problemas que causan a las personas involucradas, interrumpen la normalidad circulación de los vehículos.

Los túneles al establecer vías dedicadas necesitan estar disponibles en todo el tiempo, cuando ocurre un accidente puede quedar inhabilitado el carril en el que sucedió el accidente o bien todo el túnel, siendo un problema para los vehículos que recién ingresan al túnel.

Una de las dificultades en un accidente es que la notificación de haberse producido el accidente puede tomar mucho tiempo, por consecuencia el recibir la debida ayuda. El tiempo que transcurre entre el aviso del accidente y el dejar libre los carriles obstruidos podría ser disminuido si se supiera de una manera automática que ha ocurrido un atasco.

1.2. Limitantes

La detección de objetos en calles se ve afectada por el ambiente o estado climático, como por ejemplo la lluvia, neblina, o el cambio brusco de luminosidad debido al sol, incidiendo en la calidad de detección.

Una limitante en el desarrollo del proyecto es no contar con autos a escala cuyas radiofrecuencias sean independientes. En el mercado los autos a escala de distinto o igual modelo son fabricados sin tener en cuenta que sus radiocontroles causen interferencia entre sí. Utilizamos autos a la misma escala y cuya frecuencia de control es de 5 MHz. Al haber interferencia se puede usar un auto a la vez, esto restringe la simulación de tráfico, porque solamente se haría detección de un auto y no sería aplicable al ser este un caso de detección sencillo.

La cámara captura las imágenes de un mismo automóvil con diferentes intensidades, cuando el automóvil se desplaza existen pequeños lapsos de tiempo en que la intensidad de luz de las luminarias disminuye debido al límite entre una luminaria con la siguiente luminaria. El enfoque de la cámara también influye en las variaciones de intensidad de las imágenes capturadas, al acercarse un vehículo la cámara aumenta la intensidad y al no existir la intensidad de visión disminuye. Esto se considera una

limitante al realizar un comparador de color, un programa que determina el color del vehículo dada una imagen, porque al variar la intensidad de la imagen varían la intensidad de los pixeles, y esto hace que el programa determine otra tonalidad de color.

En el trayecto del túnel que se ha realizado de escala 1:35 [8], y una longitud de 110 cm. se pueden hacer pocas tomas de imágenes dado el procesamiento del computador de 1GB de RAM que utilizamos y la velocidad máxima que es de 50 Km/h o equivalente a 13.8 m/s, esto es porque el rango de visión de la cámara alcanza aproximadamente 60 cm. Si se desea hacer un control de velocidad máxima no sería posible, debido a no existir la cantidad suficiente de imágenes ya que se captura una imagen cada 40 cm. como máximo, para velocidades menores la distancia que avanza cada vehículo disminuye, es por esto que para velocidades entre 30 a 40 km/h se pueden capturar de 2 a 3 imágenes. Para realizar un control de velocidad máxima sería necesario tener más de una cámara en un trayecto de mayor longitud.

1.3. Consideraciones

La detección de vehículos se la puede realizar en cualquier lugar en donde se encuentren presentes, como por ejemplo: calles públicas,

carreteras, estacionamientos, puentes, túneles, etc. Dadas las características que reúnen los túneles, como iluminación permanente adecuada para los conductores siendo uniforme por tramos y que únicamente los extremos de los túneles se ven afectados por el estado climático, hacen que este caso sea el más factible para realizar la detección de vehículos, es por ese motivo que los lugares que se han escogido para detección han sido los túneles.

La siguiente consideración que se ha tomado es realizar la detección en un túnel a escala que hemos elaborado, con una escala de 1:35, y cuyas medidas de longitud, ancho y altura son 110 cm, 30 cm y 17cm. correspondientes a aproximadamente 38.5 m, 10.5 m y 6 m. Se encuentra distribuido en dos carriles de 12.7 cm equivalente a 4.5 m cada uno. La ubicación de la cámara se encuentra entre los dos carriles y a una altura de 14 cm (4.9 m) para una detección equitativa en ambos carriles. El rango de visión de la cámara es de 60 cm aproximadamente, a 20 cm del inicio del túnel.

Los túneles cuentan con diferentes intensidades de iluminación, dependiendo en donde se ubique cada tramo, así los túneles se dividen en: acceso, umbral, transición, interior y salida [3]. Las zonas intermedias

de iluminación dependen de la iluminación de los extremos que se encuentran afectados por el medio ambiente. Para facilidad de construcción del túnel y de detección de imágenes se ha considerado realizar la iluminación interior del túnel de una forma uniforme, de esta manera simular un solo tramo de túnel.

Para realizar la detección de los vehículos con los diferentes casos de tráfico, así como los atascos se capturó una a una la imagen. Al colocar un automóvil en la calzada del túnel se captura la imagen, luego de esto la siguiente imagen capturada será del automóvil desplazado cierta distancia. La velocidad del vehículo dependerá de la distancia que se lo ha desplazado con respecto a su imagen anterior.

CAPITULO 2

2. ELABORACIÓN DE TÚNEL A ESCALA

2.1. Iluminación

La iluminación es una parte fundamental en el diseño de un túnel, mantiene una relación estrecha con las dimensiones y características de la infraestructura, incluso se considera la incidencia de la iluminación solar. En la elaboración del diseño se toman en cuenta las diferentes zonas de iluminación, que varían en la intensidad de luz de acuerdo a su ubicación, tanto la zona de acceso como de salida deben permitir que la retina de los conductores se adecue al cambio de luminosidad, para precautelar la seguridad de los conductores [3].

En la elaboración del túnel a escala, se representa simplemente un tramo, tratando así de evitar todo lo que implica los cálculos para la iluminación. Se ha realizado de una manera tal que parezca una iluminación uniforme, adecuado para la cámara, así como se muestra en la Imagen 2.1.



Imagen 2.1. Vista Interna del Túnel.

2.1.1. Conexión de DAQ

Para administrar el sistema eléctrico en el túnel a escala se utiliza una tarjeta de adquisición de datos, la cual se conecta a las luminarias internas y a los paneles externos de señalización.

Se emplea un módulo en simulink como interface para controlar las salidas de la tarjeta DAQ, así se tiene un control para los paneles de entrada y el encendido de luces.



Imagen 2.2. Tarjeta DAQ

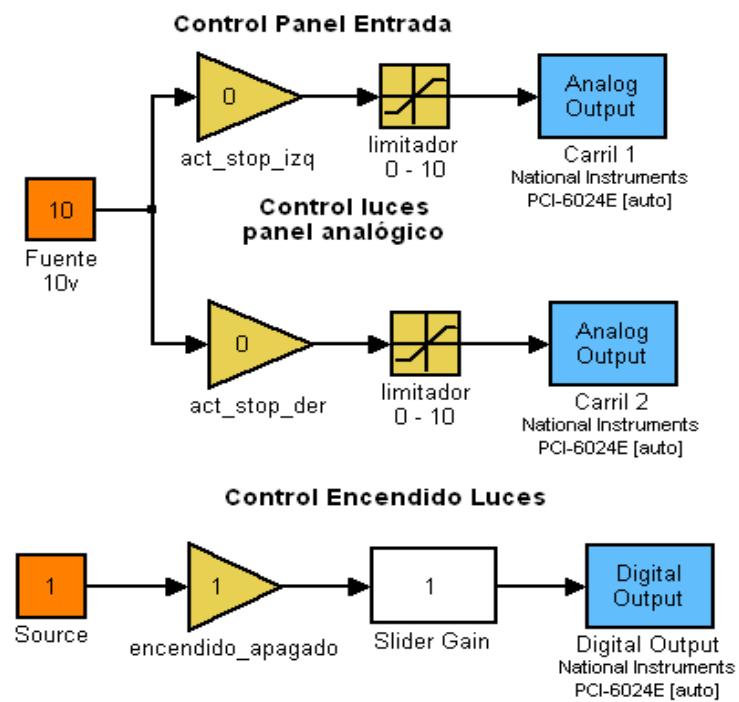


Imagen 2.3. Interface de la Tarjeta DAQ.

2.1.1.1. Luminarias Internas

Se utilizan lámparas a 12 voltios AC conectadas a sus correspondientes transformadores.

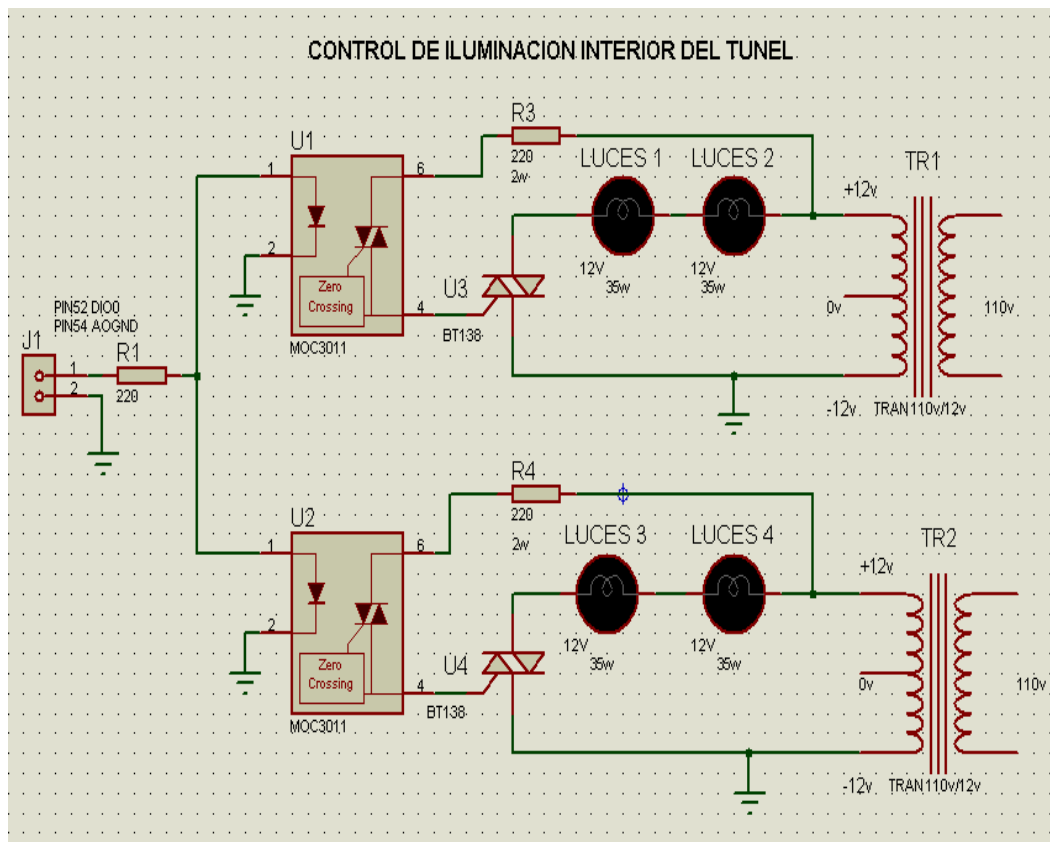


Imagen 2.4 Control de iluminación interior del túnel.



Imagen 2.5 Luminaria Interna 12 V AC.

2.1.1.2. Paneles externos

La disponibilidad de cada carril es indicada mediante un panel de señalización. Cuando el tránsito fluye normalmente, el panel muestra un círculo de color verde, en caso contrario mostrará una "X" de color rojo.

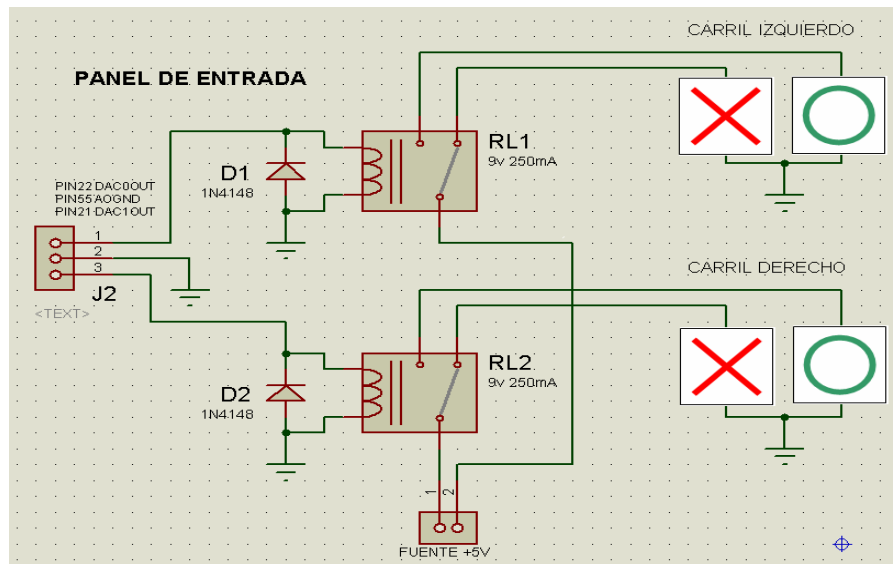


Imagen 2.6. Control de los paneles de entrada.

2.2. Medidas y Escalas

La escala que se utiliza es 1:35, así presentamos las medidas que tienen tanto el túnel como los automóviles.

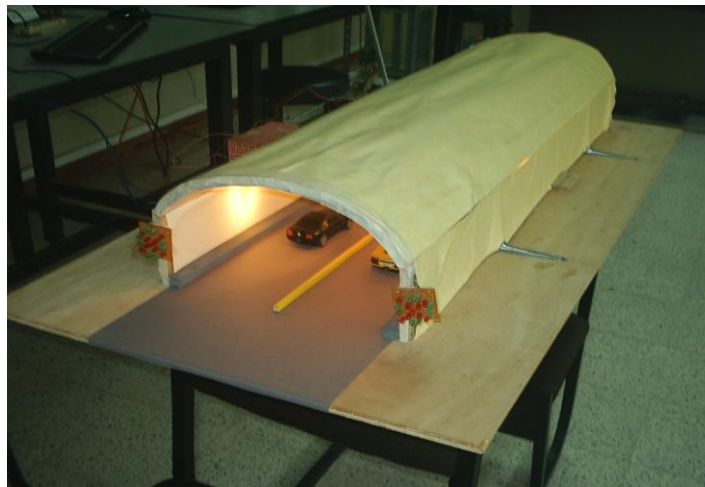


Imagen 2.7. Túnel a escala.

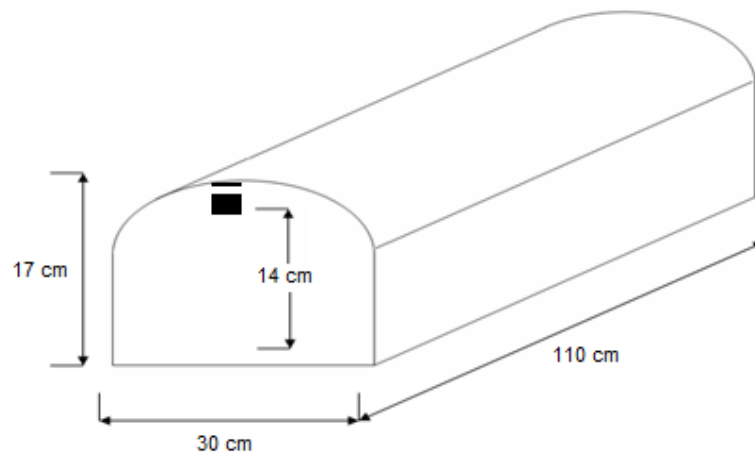


Imagen 2.8. Medidas del Túnel

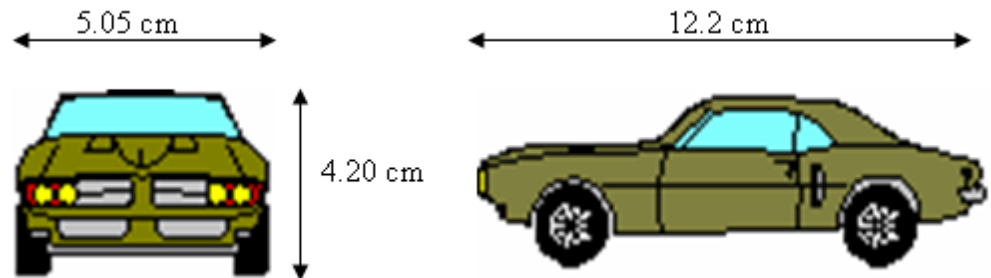


Imagen 2.9. Medidas de Automóviles.

2.3. Ubicación de la Cámara

Se ha creído conveniente que la ubicación de la cámara sea en la parte central para una vista simétrica, aprovechando de la mejor manera el monitoreo de los dos carriles.

2.3.1. Ángulo de visión

Dado que la cámara de video tiene una visión radial con respecto a su vista superior podemos decir que el valor de la distancia d con respecto a la webcam es igual que la distancia con d' con la cámara de video. Dicho esto el cálculo de la distancia recorrida se puede determinar basado en la posición del eje central.

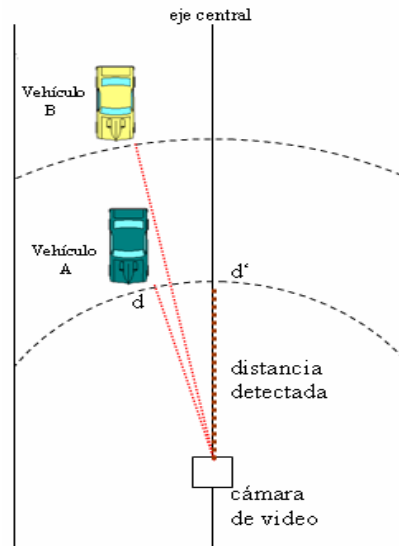


Imagen 2.10 Visión de la cámara.

2.3.2. Relación pixeles-metros

Basados en la ubicación del objeto desplazado sobre el eje central se puede determinar la posición en centímetros del vehículo con respecto a la posición de la cámara.

Como datos tenemos que la altura de la cámara es de 14 cm, la distancia mínima y máxima señaladas en la Imagen 2.11 son 23 cm y 83 cm correspondientemente.

La relación entre α_x que es el ángulo comprendido entre la distancia del inicio del objeto hasta el punto mínimo de visión, será determinada a través de cálculos basados en trigonometría hasta encontrar nuestra relación no lineal distancia α_x vs α_x

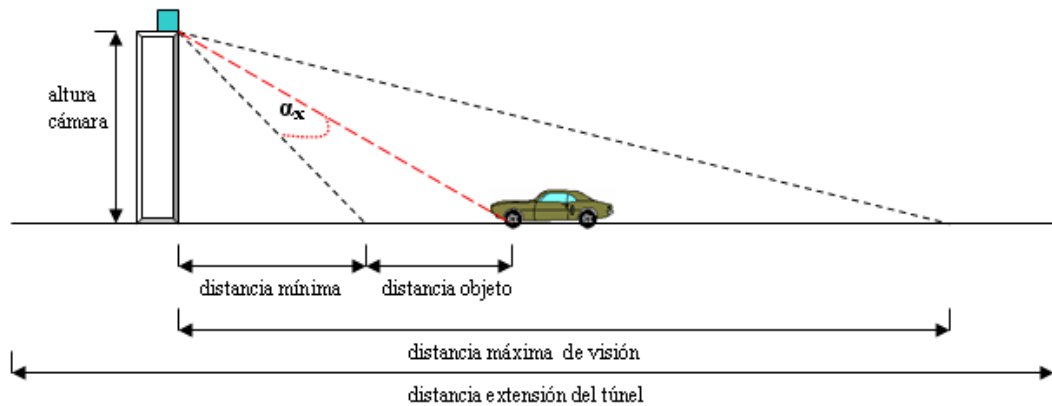


Imagen 2.11 Ubicación de la cámara.

2.3.2.1. Relación dist_x vs α_x

Primero obtenemos las expresiones para calcular α_{min} y α_{max} , luego la relación dist_x vs α_x .

$$\alpha_{min} = \arctg\left(\frac{dist_min}{altura}\right) \quad ec.1$$

$$\alpha \text{ min} = \text{arctg}\left(\frac{23 \text{ cm}}{14 \text{ cm}}\right) = 58.67^\circ$$

$$\alpha \text{ max} = \text{arctg}\left(\frac{\text{dist_max}}{\text{altura}}\right) \quad \text{ec.2}$$

$$\alpha \text{ max} = \text{arctg}\left(\frac{83 \text{ cm}}{14 \text{ cm}}\right) = 80.43^\circ$$

$$\text{tg}(\alpha x + \alpha \text{ min}) = \frac{\text{dist_x} + \text{dist_min}}{\text{altura}}$$

$$\text{dist_x} + \text{dist_min} = [\text{altura} * \text{tg}(\alpha x + \alpha \text{ min})]$$

$$\text{dist_x} = [\text{altura} * \text{tg}(\alpha x + \alpha \text{ min})] - \text{dist_min} \quad \text{ec.3}$$

$$\text{dist_x} = [14 * \text{tg}(\alpha x + 58.67^\circ)] - 23 \text{ cm}$$

2.3.2.2. Relación distancia x vs fila x

Ya relacionado la distancia con el ángulo α_x queda por determinar la relación entre α_x con la posición de la fila donde se detectó el inicio del objeto m_x .

Para una comprensión mejor se puede observar la Imagen 2.12, en donde se encuentra una imagen de un vehículo y las divisiones en pixeles con m filas y n columnas.

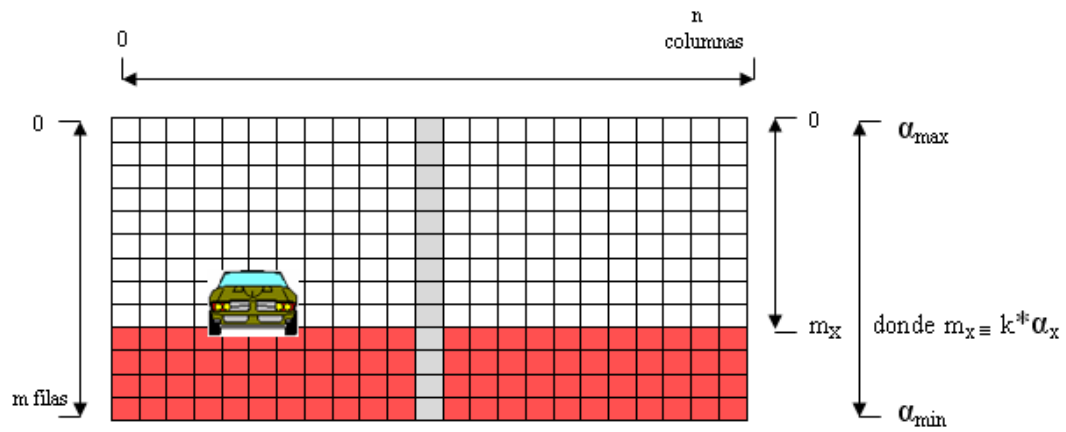


Imagen 2.12 Captura bajo una resolución $m \times n$.

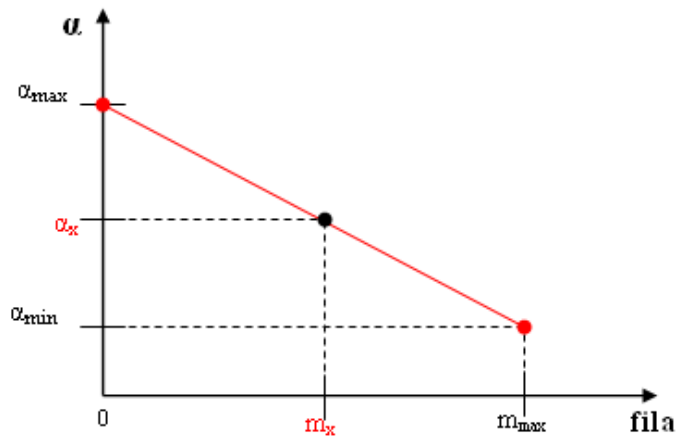


Imagen 2.13 Relación α_x vs fila_x .

$$(\alpha_{\min} - \alpha_{\max}) = k * (m_{\max} - 0)$$

$$k = \frac{(\alpha_{\min} - \alpha_{\max})}{m_{\max}} \quad \text{ec.4}$$

$$\alpha.x = k * mx + \alpha \max \quad ec.5$$

Por lo que nos lleva uniendo todas las ecuaciones a la relación **distancia_x ∝ fila_x**

$$dist_x = \left[altura * tg \left(\left(\frac{\arctg\left(\frac{dist_min}{altura}\right) - \arctg\left(\frac{dist_max}{altura}\right)}{fila_max} \right) * fila_x + \arctg\left(\frac{dist_max}{altura}\right) + \arctg\left(\frac{dist_min}{altura}\right) \right) \right] - dist_min$$

CAPITULO 3

3. DETECCIÓN DE IMÁGENES

3.1. Generalidades

La Detección de imágenes permite obtener o diferenciar características de objetos, de esta manera se adquiere la información requerida para su debida aplicación y brindar soluciones. Se emplea la detección en el monitoreo del flujo de tráfico, con los datos procesados se indica la viabilidad de los carriles, verificando si existe algún incidente, y se realizan estadísticas.

Un sistema de monitoreo de tráfico es uno de los sistemas que se emplea en el sistema de control de un túnel, muy útil para brindar un buen servicio y mantener la seguridad.

3.2. Multi-frame

Multiframe es un arreglo de 4 dimensiones, el cual contiene una serie de frames o imágenes, dando la facilidad de almacenarse en un solo archivo.

El formato de Multiframe es el que se utiliza, debido a la secuencia de imágenes, se realiza el mismo proceso de detección en cada frame para generar las señales de salida necesarias.

3.3. Imagen de Intensidad

Imagen de Intensidad o Imagen en escala de grises es la representación de cada píxel en una tonalidad de gris. Una imagen de intensidad de clase double, asigna un número flotante (número con decimales), entre cero y uno a cada píxel. El valor cero corresponde al Negro y el uno al Blanco[1]. En la clase uint8 se asigna un entero entre 0 y 255 para representar la brillantez de un píxel. Los valores cercanos al cero son más oscuros y los valores cercanos a 255 son más claros [4].

Las operaciones morfológicas, detección de bordes, segmentación, uso de filtros para el procesamiento de imágenes se las realiza en imágenes de intensidad, debido a eso es la importancia de mencionarlas.

3.4. Detección de bordes

En la detección de bordes se verifica si existe alguna discontinuidad de intensidad en los píxeles de una imagen en escala de grises o binaria

[5], como resultado se obtiene una imagen binaria, si se encontró bordes los pixeles toman el valor de uno, en caso contrario los pixeles toman el valor de cero.

Se utiliza el método de Canny para la detección, el cual aplica métodos de diferencias finitas en base a la primera derivada. Considerado uno de los mejores métodos debido a su sencillez y buenos resultados, con el beneficio de una gran velocidad de procesamiento [6].

Se procede a hacer un filtrado con una máscara predefinida que contiene los contornos del túnel junto con la detección de bordes para eliminar algunos detalles. Se obtiene una imagen binaria, en la cual se encuentran los contornos de los automóviles junto con rasgos no deseados, esto puede ser debido a luminosidad, las sombras en la rugosidad de las vías o a la resolución de la cámara.



Imagen 3.1 Detección de bordes.

3.5. Análisis Morfológico

Se basa en los campos de la geometría y la forma, siendo una técnica no lineal. Su uso puede darse para:

- Separar determinadas regiones
- Unir regiones
- Suavizar bordes
- Facilitar el cómputo de las regiones de una imagen

Para las transformaciones morfológicas se suele utilizar un conjunto de forma conocida llamado elemento estructurante. La función `strel(shape, parameters)`, permite retornar un elemento estructurante definido por su forma (shape). En nuestro caso, utilizamos dos de las formas básicas: disco (disk) y cuadrado (square) [2].



Imagen 3.2 Formas básicas.

Se tiene las siguientes operaciones morfológicas: dilatación, erosión, apertura y cierre.

3.5.1. Dilatación

Permite añadir puntos en un objeto en los pixeles que tocan el borde en una imagen binaria o escala de grises, aumentando la definición de la imagen [2]. El comando que permite realizar esto en matlab es `imdilate(Im, Se)`, recibe una imagen y un elemento estructurante.

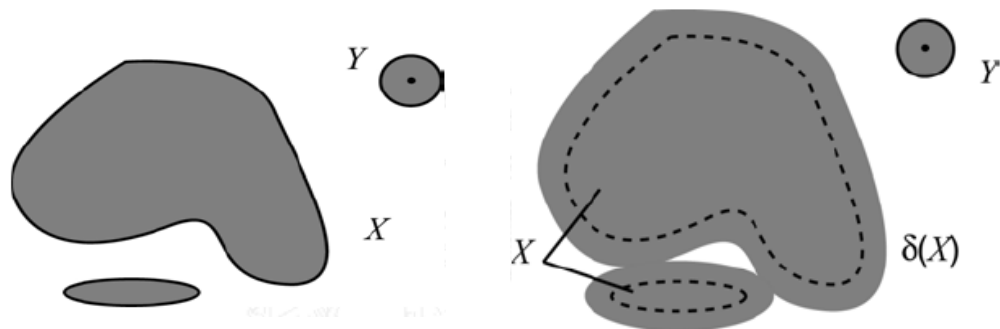


Imagen 3.3 Dilatación de X con el elemento estructurante Y .

3.5.2. Erosión

Utiliza la sustracción de vectores o comparación, verificando que el elemento estructurante se encuentre contenido en la imagen, se obtendría una salida de uno si estos coinciden [2]. Lo cual permite simplificar una imagen o eliminar detalles con poca importancia. En

matlab se usa el comando `imerode(Im,Se)`, recibe la imagen binaria o en escala de grises, y el elemento estructural.

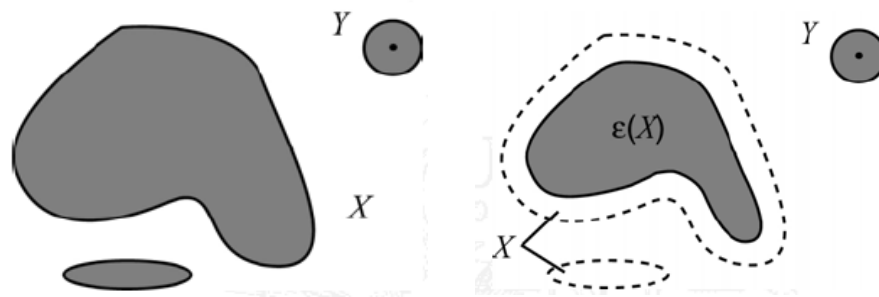


Imagen 3.4 Erosión de X con el elemento estructurante Y .

3.5.3. Apertura y Cierre

Combinan la dilatación y erosión, con la finalidad de eliminar elementos de menor tamaño al elemento estructural, esto permite atenuar el ruido y la extracción selectiva de objetos en la imagen [2]. La operación que realiza luego del proceso de erosión el de dilatación se la conoce como apertura, en cambio si se efectúa primero dilatación y a continuación el proceso de erosión se produce la operación de cierre [2].

Estas operaciones morfológicas se conocen como filtros básicos, permitiendo al realizar diversas combinaciones el desarrollo de filtros más complejos.

Como ilustración de estas operaciones se presenta en las Imágenes 3.5 y 3.6 estas operaciones.

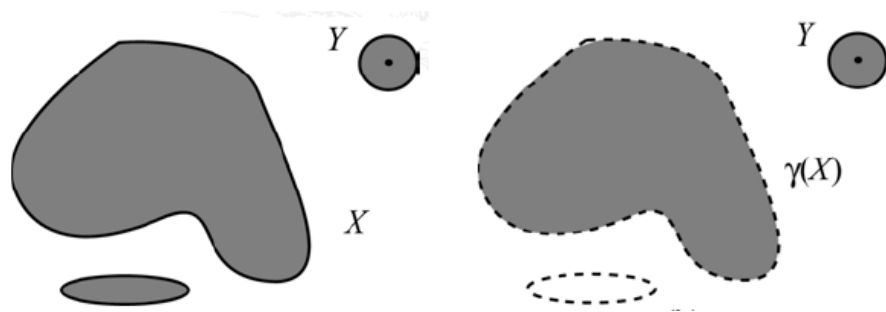


Imagen 3.5 Operación de Apertura de X con el elemento estructurante Y.

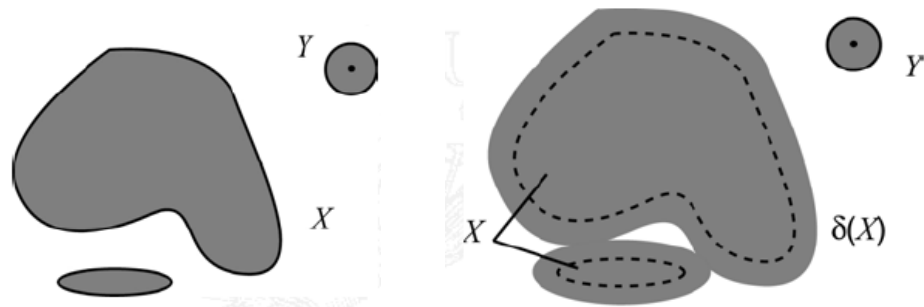


Imagen 3.6 Operación de Cierre de X con el elemento estructurante Y.

Se tiene el comando `Imclose (Im,Se)`, el cual requiere una imagen en escala de grises o binaria y el elemento estructurante. Como respuesta se obtiene la imagen cuyos espacios internos de los objetos han sido cerrados, de acuerdo al elemento estructural. Aquí se realiza la operación de cierre, la operación dual a esta estaría dada por el comando `imopen (Im,Se)` que recibe los mismos argumentos.

`imfill(Bw,'holes')`, rellena los agujeros de los objetos de una imagen `Bw` binaria o en escala de grises. Refiriéndose a agujeros como el grupo de píxeles de fondo que están limitados por los bordes de la imagen.

`bwareaopen(Bw,P,conn)`, elimina de una imagen binaria `Bw` todos los componentes conectados con un tamaño de píxeles menor a `P`, la conectividad (`conn`) por defecto para dos dimensiones es 8.

3.6. Etiquetado de imágenes

Realizados los diferentes procesos para eliminar elementos no requeridos, se necesita obtener cada objeto que ha quedado en la

imagen. No obstante, se obtiene objetos que no están completos, es por eso que utilizaremos las ubicaciones o coordenadas de estos, para posteriormente extraerlos de la imagen original.

`[L,num] = bwlabel(Bw,n)`, empleamos este comando, que toma por argumento una imagen binaria `Bw` y un valor `n` que representa la conectividad, esto puede ser un valor de 4 o también 8. Retorna los objetos conectados (`L`), y el número de objetos (`num`).



Imagen 3.7 Imagen Binaria para ser etiquetada.

3.7. Segmentación

Identificados los objetos en la imagen, se hace un barrido para obtener las coordenadas, y se analiza cada objeto. Con las coordenadas y la

imagen original se extrae el objeto y se lo enmarca, en este caso la imagen de un vehículo.

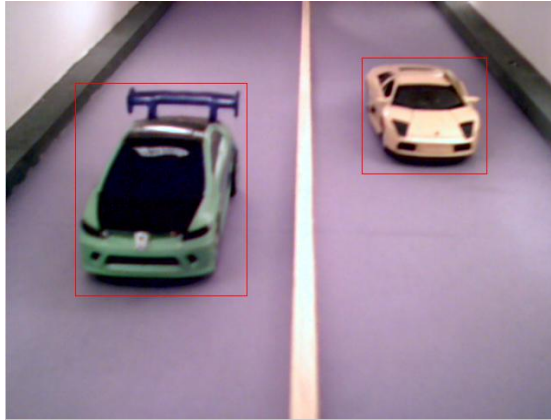


Imagen 3.8 Identificación de objetos.

CAPITULO 4

4. COMPARACIÓN DE IMÁGENES

4.1. Generalidades

Teniendo las imágenes segmentadas, con los objetos disponibles para su análisis se procede a identificar a cual carril pertenece cada vehículo. Para una simplificación se ha asumido que los automóviles permanecen durante todo el trayecto en su respectivo carril. Según nuestra figura estaríamos en la parte de almacenamiento de objetos y sus posiciones.

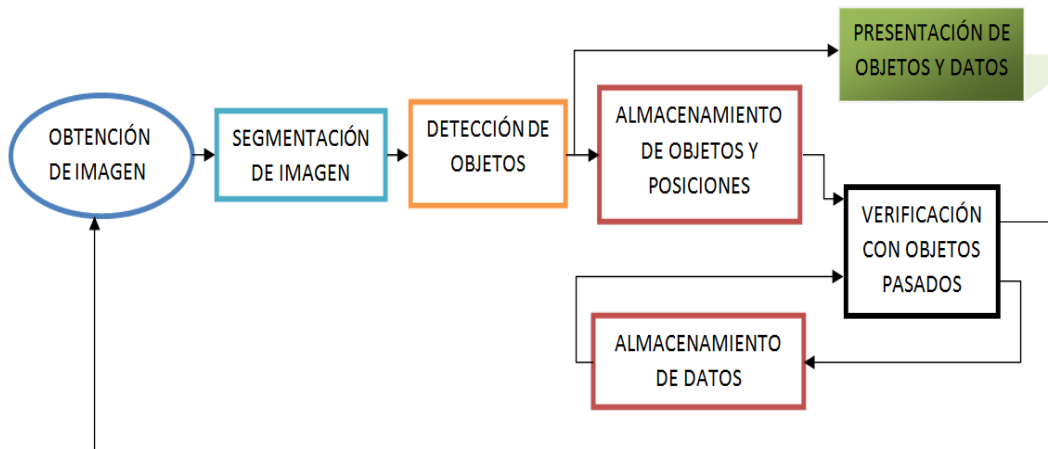


Imagen 4.1. Diagrama del Sistema

Cuando se captura un nuevo frame se realiza el mismo procedimiento, y con los datos almacenados anteriormente se procede hacer las verificaciones.

Se realizan dos tipos de evaluaciones para determinar si se trata del mismo objeto, pudiendo producirse dos de los casos: un vehículo detenido o un vehículo que avanza. Las evaluaciones son: Comparación por Correlación y Comparación por Posición.

4.2. Comparación por Correlación

Teniendo las dos imágenes a compararse, se realiza un procedimiento para igualar sus dimensiones y se las convierte a escala de grises.

En la correlación cada pixel de salida se calcula como la suma ponderada de sus pixeles vecinos. Para esto se utiliza una máscara llamada núcleo o kernel. Si se realiza la correlación de A con h en el valor (2,4) como lo muestra la Imagen 3.2, se deberá ubicar el valor central de la máscara en el valor de interés, y multiplicar uno a uno los valores correspondientes, para luego ser sumados. Se tendría $1 \cdot 8 + 8 \cdot 1 + 15 \cdot 6 + 7 \cdot 3 + 14 \cdot 5 + 16 \cdot 7 + 13 \cdot 4 + 20 \cdot 9 + 22 \cdot 2 = 585$ [1].

17	24	8	1	6
23	5	3	5	7
4	6	4	9	2
10	12	19	21	3
11	18	25	2	9

Imagen 4.2. Correlación de A con h.

La correlación está dada por:

$$g(x, y) = h(x, y) \circ f(x, y) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} f * (i, j) h(x + i, y + j)$$

Las ventajas en este proceso es que la estimación de resultados no depende de la luz incidente en los objetos o la calidad de la cámara, sin embargo la operación de correlación toma tiempo de acuerdo a las dimensiones de las imágenes. Se genera también una dificultad con el cambio del ángulo de visión en los automóviles, porque al estar lejos o acercarse estos a la cámara se notan nuevos detalles y otros ya no son

tan perceptibles, es por eso que la correlación de dos frames en secuencia de un automóvil que ha avanzado no tendrá un acierto de cien por ciento como respuesta.

Cuando la correlación entre dos objetos arroja un resultado con un 90 por ciento o mayor de acierto, se asume que se trata del mismo objeto. Aunque esto no indique efectividad en que sea el mismo objeto, es un buen indicador.

4.3. Comparación por Posición

Cuando se etiqueta un objeto se guardan las coordenadas que lo limitan. En la siguiente secuencia de frame, también se guardarán las nuevas coordenadas del vehículo. Al compararse las coordenadas de ambos frames y al tener una diferencia mayor a ocho pixeles, se concluye que el objeto se ha desplazado. Esto puede constituir una limitante si el mismo vehículo en una secuencia de tramas no se detecta igual manera, se detecta con coordenadas diferentes y se procesa como si se hubiese desplazado, esto puede ser debido a que un evento el vehículo con respecto al siguiente evento el mismo vehículo cuentan con diferente tipo de iluminación.

4.4. Detección de Atasco

Si en las comparaciones con sus porcentajes de acierto se determina que se trata del mismo vehículo, se designa como un vehículo detenido. Se envía una alarma como aviso al ocurrir que un vehículo se mantiene en el mismo lugar detenido durante un lapso de cuatro segundos o durante cuatro tramas seguidas. Al enviarse la alarma se cambia la señalización en el panel de entrada, indicando que el carril no está disponible, hasta que el atasco se resuelva.

CAPITULO 5

5. ESTADISTICAS

Teniendo como muestras 195 tramas, que incluyen 113 eventos en que se encuentran vehículos; hemos podido calcular las diferentes probabilidades y porcentajes que corresponden a: Probabilidad en una trama, Probabilidad entre tramas y Probabilidad de Alarma.

Al nombrar Probabilidad en una trama nos referimos a la probabilidad que un vehículo sea detectado dado que no existe (falso positivo) y a la probabilidad que un vehículo no sea detectado dado que se encuentra en la calzada (falso negativo). Hemos designado a la probabilidad de un falso positivo como $P(fp)$ y a la probabilidad de un falso negativo como $P(fn)$.

En caso de un falso negativo, cuando existe un auto en la calzada, este puede ser o no detectado, asumiendo que existe la suficiente distancia entre un auto y otro para ser vistos como eventos independientes. Teniendo 122 eventos, se tiene que 9 de estos no se detectaron y con una probabilidad de éxito de 0.5 al ser detectados. La probabilidad de encontrar x cantidad de falsos negativos esta determinada por la distribución binomial.

Se tiene que la probabilidad de que 9 vehículos no sean detectados es igual

$$a: P(f_n(9)) = \binom{122}{9} 0.5^9 0.5^{113} \approx 0$$

En cambio si obtenemos la probabilidad de f_n como un evento simple tenemos:

$$P(f_n) = 9/122 = 0.0687, \% f_n = 6.87\%$$

Realizando del mismo modo la probabilidad de un falso positivo, de encontrar un vehículo dado que no existe se tiene:

$$P(f_p) = 5/65 = 0.0769, \% f_p = 7.69\%$$

Para poder referirnos a la probabilidad entre tramas tomamos en cuenta a la probabilidad que un vehículo en la siguiente secuencia no sea detectado como el mismo vehículo, a la probabilidad que un vehículo sea detectado como el mismo vehículo dado que el segundo vehículo se encuentra en una posición cercana en donde se encontraba el primer vehículo y a la probabilidad que un vehículo no se mueva y sea detectado como si fuera otro vehículo en la siguiente trama.

Designando como probabilidad de un falso negativo entre tramas tipo A (fntA) a la probabilidad que un vehículo no se mueva y en la siguiente trama se detecte como si se hubiera desplazado, se tiene que en una muestra de 62 secuencias, 4 de ellas se detectaron como si el vehículo se hubiese desplazado. Refiriéndonos como secuencias a dos tramas seguidas.

$$P(\text{fntA}) = 4/62 = 0.0645, \% \text{ fntA} = 6.45\%$$

Llamamos falso negativo entre tramas tipo B (fntB) cuando un mismo vehículo en la siguiente trama no sea detectado como el mismo vehículo habiéndose desplazado. En una muestra de 62 secuencias se encontró 3 vehículos no detectados al desplazarse.

$$P(\text{fntB}) = 3/62 = 0.0483, \% \text{ fntB} = 4.83\%$$

Probabilidad de un falso positivo entre tramas (fpt) es la probabilidad en que un vehículo sea detectado como el mismo vehículo en la siguiente trama al ocupar una posición cercana al primer vehículo, para esto se obtuvo una serie de tramas con vehículos diferentes en una posición cercana de un vehículo de la trama anterior. Se obtuvo que de 13 secuencias 3 casos resultaron si fueran el vehículo anterior.

$$P(\text{fpt}) = 3/13 = 0.2307, \% \text{ fpt} = 23.07 \%$$

Para enviar alarma de atasco debe producirse que un vehículo sea detectado como si hubiese permanecido en el mismo lugar durante cuatro tramas, si cada trama es tomada cada segundo.

Para la probabilidad de alarma tomamos en cuenta la probabilidad que la alarma sea enviada al no existir ningún un atasco y la probabilidad de no enviarse la alarma existiendo un atasco.

La probabilidad de falso positivo de alarma (fpAl) se puede dar cuando se envía la señal de alarma al detectar diferentes vehículos en una misma posición y se procesa como si fuese el mismo vehículo durante cuatro tramas, esto puede incluir el reconocimiento de vehículos no existentes. Dicho esto tenemos que: $P(\text{fpAl}) = \text{Pr}(\text{fpt}) + \text{Pr}(\text{fp}) + \text{Pr}(\text{fp} \cap \text{fp})$, en donde Pr es la probabilidad resultante durante las cuatro tramas.

$$P(\text{fpAl}) = 0.2307^3 + 0.0769^4 + [(0.2307^2)(0.0769) + (0.2307)(0.0769^2)]$$

$$P(\text{fpAl}) = 0.0122 + 3.49 \times 10^{-5} + 5.45 \times 10^{-3} = 0.01777, \% \text{ fpAl} = 1.78\%$$

La probabilidad de falso negativo de alarma (fnAl) ocurre al no enviarse la señal de alarma al ocurrir un atasco, esto puede ser que durante un atasco

se detecto que el vehículo se desplazo cuando se mantuvo en el mismo lugar, también puede suceder que no se detecto al vehículo o algún otro vehículo esta tan cercano que lo reconoce como un nuevo objeto. Teniendo las probabilidades anteriores y a la probabilidad de que uno o más vehículos cercanos causen error de detección (fc) obtenemos lo siguiente:

$$P(\text{fnAl}) = \Pr(\text{fntA}) + \Pr(\text{fn}) + \Pr(\text{fntA} \cap \text{fn}) + \Pr(\text{fc})$$

$$\Pr(\text{fntA}) = 0.0645^3 + 0.0645^2 + 0.0645 = 0.06893$$

$$\Pr(\text{fn}) = 0.0687^4 + 0.0687^3 + 0.0687^2 + 0.0687 = 0.07376$$

$$\Pr(\text{fntA} \cap \text{fn}) = (0.0645)(0.0687^2) + (0.0645^2)(0.0687) = 5.902 \times 10^{-4}$$

$$P(\text{fnAl}) = 0.06893 + 0.07376 + 5.902 \times 10^{-4} + \Pr(\text{fc}) = 0.1433 + \Pr(\text{fc})$$

Se tiene entonces que el porcentaje de que no se envié alarma existiendo un atasco es de 14,33% sin tomar en cuenta los errores de detección al estar dos o más vehículos muy cercanos. La obtención de $\Pr(\text{fc})$ es dependiendo de los eventos, constituyendo fc una variable aleatoria, que puede afectar excesivamente al envío de alarma si llegan haber vehículos muy cercanos que son tomados como un solo objeto.

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

- 1) El procesamiento de imágenes se hace indispensable conforme la tecnología aumenta, surgiendo mejoras en la detección de objetos así como también los procesos que involucran la detección. Esto se debe a que al hacer detección de objetos hace posible que los procesos requeridos sean autónomos.
- 2) La detección de un objeto se ve afectada por la calidad de detección de la cámara, el proceso de detección y por los elementos de ambiente a los que esta expuesto ya que incide en la iluminación. Este último factor es más determinante que los otros y más complicado de ser controlable.
- 3) Dependiendo del tipo de iluminación se puede hacer que un objeto, en nuestro caso automóviles sean definidos mejor o se contrasten, ya que pueden existir sombras ocultas y esto hace que no se obtenga un buen color. Cuando existe una buena diferencia de tonalidad de color de un mismo objeto entre tramas, el reconocer el color principal que lo compone

se hace muy complicado, esto es debido a que la mayoría de los pixeles se ha definido en otro umbral de color.

- 4) Se tiene que la probabilidad parcial de que no se envíe alarma existiendo un atasco es de 14.33% y la probabilidad que se envíe alarma no existiendo atasco es de 1.78 %. La probabilidad de que no se envíe alarma es elevado, esto se puede mejorar utilizando una cámara de mejor resolución aunque esto implica mayor consumo en procesamiento y costos. Al usar una cámara de menor calidad de 640 x 480 pixeles el costo total del proyecto como se muestra en el Anexo 1 disminuiría en un 26%, en cambio al usar una cámara IP de 1280x1024 pixeles, con mejores características, el costo del prototipo aumentaría en mas del 100% al costo total.
- 5) Durante el desarrollo del proyecto hemos empleado los conocimientos impartidos acerca de procesamiento digital de señales utilizando la herramienta matemática MATLAB, lo cual consideramos satisfactorio al reforzar los conocimientos.

RECOMENDACIONES

- 1) Se recomienda que para el proceso de adquisición de tramas para los diferentes análisis de tráfico la cámara sea previamente calibrada y se mantenga de esa manera durante todo el proceso.
- 2) Como una mejora a futuro de detección de atascos y control de velocidad máxima se recomienda utilizar por lo menos dos tramas de túnel lo que implica la utilización de mas de dos cámaras trabajando secuencialmente.
- 3) Para realizar un procesamiento más rápido de las imágenes y la detección de los vehículos se recomienda utilizar un computador con mayor memoria RAM a la que hemos utilizado. Esto fue por motivo que este computador contenía la interface para la DAQ.

ANEXO 1

TABLA DE COSTOS DEL PROTOTIPO

MATERIALES	CANTIDAD	TOTAL
Plywood 9mm ½	1	10.44
Tira de laurel 1x3x2	8	6.20
Libra de alambre #16	1	1.50
Libra de alambre #20	1	1.80
Broca 1/16	1	0.50
Palillos cuadrados 8x8x90	1	0.25
Palillos cuadrados 5x5x90	4	0.72
Papel aperlado 150 gr 79x110	1	0.87
Cartulina kimberly	1	1.18
Cartulina para encuadernar gris	10	1.46
Focos 12v	4	4.00
Resistencia ½ w	17	0.80
Cinta de papel	1	0.27
Borneras	5	1.00
Cable UTP (1 m.)	1	0.40
Led	17	1.36
Cola blanca	1	1.00
Cámara Genius Slim 322	1	52.00
Autos de juguete a escala	2	44.00
TOTAL		130.75

ANEXO 2

Manual de Usuario de GUIDE Control de tráfico en el interior de un túnel



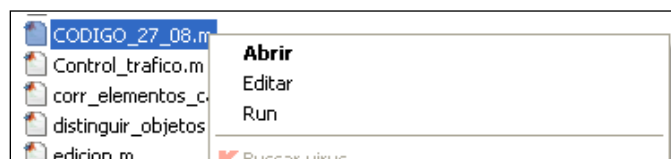
Ejecución .m File

Para ejecutar nuestra aplicación GUIDE nos dirigimos a la carpeta de matlab donde se encuentra nuestro código. Esta carpeta consta de dos archivos: *Control_trafico.m* y *Control_trafico.fig*.

Para ejecutar una Interfaz Gráfica, si la hemos etiquetado con el nombre: *Control_trafico.fig*, simplemente ejecutamos en la ventana de comandos

`>> Control_trafico.`

O haciendo click derecho en el m-file y seleccionando la opción *RUN*.



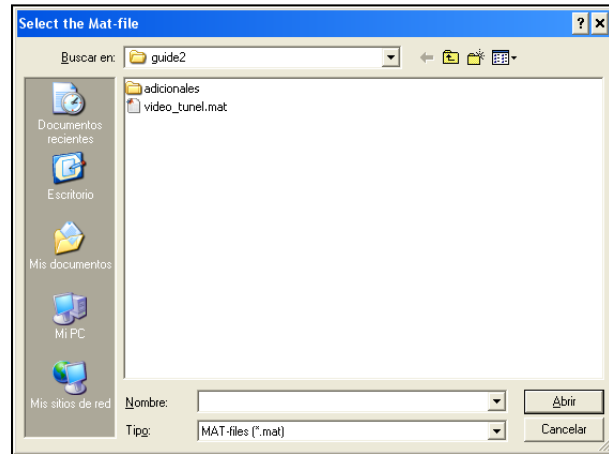
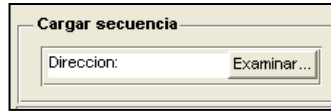
Inicio Guide

Aquí se presenta la pantalla inicial de nuestro programa:

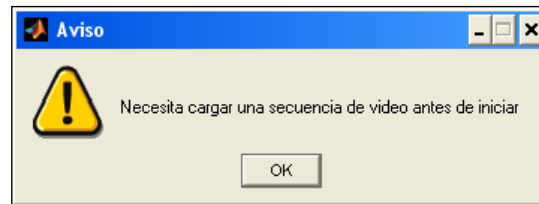


Lo primero que se debe realizar para la ejecución del programa es realizar la carga de secuencias de tráfico capturadas vía webcam dentro de nuestra maqueta de túnel, las cuales se encuentra almacenadas en varios archivos extensión .mat localizadas en nuestra carpeta principal *Matlab*.

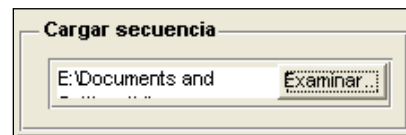
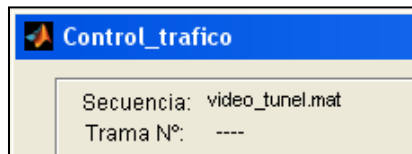
La selección se realiza desde la opción *Cargar secuencia* -> *Examinar...*



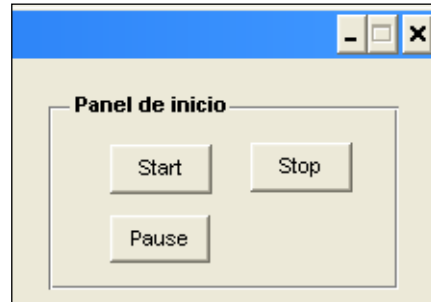
En caso de ejecutar el botón “*Start*” desde el **Panel de inicio** se generará una ventana de advertencia como la que se presenta a continuación:



Al cargarse correctamente la secuencia de video se observará que el nombre del archivo se presentará en el indicador Secuencia así como se mostrará la dirección del mismo



Ahora sí se puede mandar a correr la simulación a través de nuestro botón “*Start*” que se encuentra en nuestro **Panel de inicio** como se muestra en la figura



Al inicio de nuestra simulación se puede apreciar principalmente nuestra pantalla central que muestra la visión de nuestra videocámara dentro de nuestro túnel.

A su alrededor se presenta el número de tramas actual de simulación así como un par de paneles denominados Carril Izquierdo y Carril derecho aquí se indica que el tráfico que se realiza en el túnel es normal sin bloqueos ni atascos (alerta de atasco desactivada: color verde).



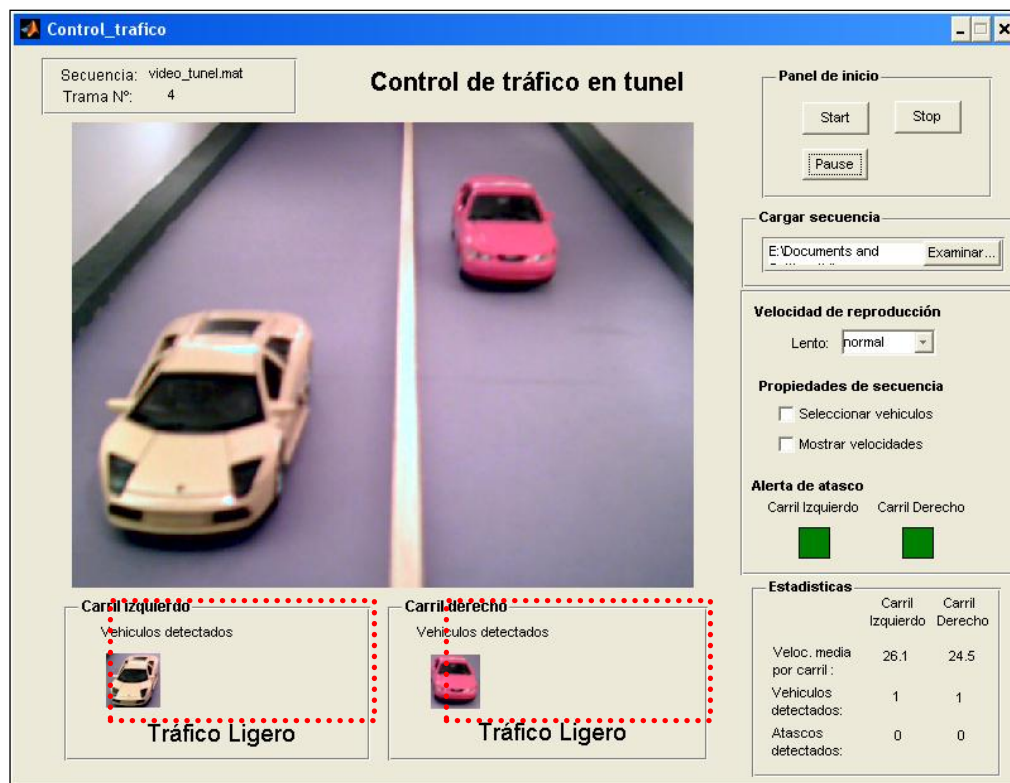
Se realiza la identificación analizando una a una las tramas del video cargado



Panel central – interior túnel

Tan pronto comience a circular vehículos estos serán detectados y almacenados en una base de datos para diferentes análisis cuantitativos y cualitativos, todos estos exámenes basados en un procesamiento de video.

Al detectarse movimiento de tráfico en el interior del túnel se presentará inmediatamente la imagen de este o de estos vehículos en nuestro panel inferior así como sus velocidades promedios en nuestro panel de **Estadísticas**.

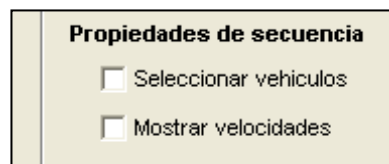


Visualización de vehículos detectados por carril:



Opciones adicionales sobre visualización sobre las tramas

En el momento de la identificación se tienen las ventanas de checkbox. Esta tiene dos opciones: *Seleccionar vehículos* así como de *Mostrar velocidades*



Al aceptar *Seleccionar vehículos* se puede apreciar que se enmarca el vehículo ya identificado en un rectángulo rojo, esto es solo para verificar la eficiencia del algoritmo.

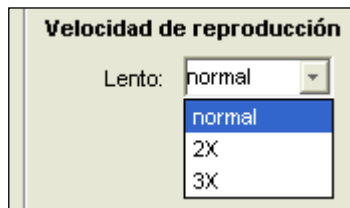


En el caso de la opción *Mostrar velocidades* se presenta un tag con su velocidad



En caso que se desee observar los parámetros por los cuales se realiza el cálculo de velocidades, distancias, colores y valores de correlación estos se presentan en nuestro **command windows** de Matlab.

El inconveniente es que al realizar una simulación rápida no es tan legible visualizar dichos parámetros, debido a esto existe la opción de reducir la velocidad de simulación para realizar cualquier observación del algoritmo.



Identificación de un atasco

En el momento en el cual un vehículo reduzca velocidad considerablemente o se detenga se generará un tag de alerta el cual se mostrará sobre el vehículo indicando "vehículo detenido" como se muestra a continuación:

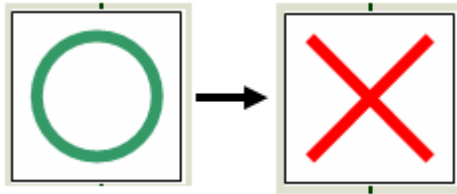


En el caso que la detención sea por más tres tramas inmediatamente se generará una alerta de atasco tanto visual en nuestro panel carril izquierdo como se encenderá el casillero de alerta atasco cambiando de un color verde a uno rojo.



En este momento se ejecuta también una alerta sonora la cual alertará inmediatamente al usuario de que existe un atasco y que debe ser supervisado dicho evento.

A su vez a través de líneas de código dirigidas a nuestro documento de simulink en caso de encontrarse conectado la simulación al prototipo se procederá a activar el panel frontal del túnel el cual interrumpirá el flujo de vehículos por dicho carril afectado por el atasco cambiando inmediatamente de:



Tablero principal entrada túnel

En caso de que el vehículo arranque y continúe su marcha normal después de haber alertado al usuario por la detención, la alarma visual se desactivará inmediatamente así como la alarma sonora y se restablecerá el panel frontal de entrada “circulación” para el ingreso de nuevo tráfico.

The screenshot shows the 'Control de tráfico en túnel' software interface. The main window displays a video feed of a tunnel with three cars: a pink car in the left lane, a yellow car in the left lane, and a green car in the right lane. The interface includes several control panels:

- Panel de inicio:** Contains 'Start', 'Stop', and 'Pause' buttons.
- Cargar secuencia:** Includes a file selection field and an 'Examinar...' button.
- Velocidad de reproducción:** A dropdown menu set to 'normal'.
- Propiedades de secuencia:** Checkboxes for 'Seleccionar vehículos' and 'Mostrar velocidades'.
- Alerta de atasco:** Two green indicator lights for 'Carril Izquierdo' and 'Carril Derecho', both of which are currently lit.
- Estadísticas:** A table showing traffic data for both lanes.

	Carril Izquierdo	Carril Derecho
Veloc. media por carril:	15.9	32.9
Vehículos detectados:	6	6
Atascos detectados:	1	0

Below the video feed, there are two panels for lane detection:

- Carril izquierdo:** Shows 'Vehículos detectados' with icons of the pink and yellow cars. A red dashed box highlights the text 'Tráfico Ligero' below the icons.
- Carril derecho:** Shows 'Vehículos detectados' with an icon of the green car. The text 'Tráfico Ligero' is displayed below the icon.

El proceso continúa trabajando hasta que termine finalmente con la cantidad de tramas de esa secuencia y dichos eventos se almacena en las estadísticas así como los vehículos detectados y la velocidad media por carril. Volviendo a su estado inicial.

Control_tráfico

Secuencia: video_tunel.mat
Trama N°: ----

Control de tráfico en tunel



Panel de inicio

Start Stop

Pause

Cargar secuencia

E:\Documents and ... Examinar...

Velocidad de reproducción

Lento: normal

Propiedades de secuencia

Seleccionar vehiculos
 Mostrar velocidades

Alerta de atasco

Carril Izquierdo Carril Derecho

Estadísticas

	Carril Izquierdo	Carril Derecho
Veloc. media por carril:	20.6	20.5
Vehiculos detectados:	6	6
Atascos detectados:	1	0

Carril izquierdo

Carril derecho

BIBLIOGRAFÍA

1. Esqueda, José; Palafox, Luis. Fundamentos para el procesamiento de imágenes. UABC, México, 2005, pp. 31-32,121-131.
2. Ortiz, Francisco. Procesamiento morfológico de imágenes en color: Aplicación a la reconstrucción geodésica, Universidad de Alicante, 2002, pp. 42 – 63.
3. Geoconsult España Ingenieros Consultores S.A., Manual de construcción de túneles, http://www.geoconsult.es/fotos/Publicaciones/Manual/13_ilum.pdf, pp. 172 – 180, 2009.
4. Escuela Biomédica de la Universidad Don Bosco, Introducción a MatLab y Tollbox Image Processing, <http://www.udb.edu.sv/Academia/Laboratorios/biomedica/Imagenes%20Medicas/guia1IM.pdf>, 2009
5. Grupo de Tecnología Industrial de la Universidad Miguel Hernández, Detección de Bordes en una Imagen, <http://isa.umh.es/doct/pava/DeteccionBordes.pdf>, 2009.

6. Valverde, Jorge. Detección de bordes mediante el algoritmo de Canny, <http://www.seccperu.org/files/Detecci%C3%B3ndeBordes-Canny.pdf>, 2009.
7. CTG, Estadísticas, <http://www.ctg.gov.ec/index.php?mod=21>, 2009.
8. Diseño Definitivos de los túneles de San Eduardo, LAHMEYER INTERNATIONAL – CONYFIS – CONSULSISMICA, http://www.ventilacion-tuneles.es/casos_practicos/tuneles_de_cerro_san_eduardo/resumen_ejecutivo_del_tuneles.pdf, 2009.
9. Becerra, Pablo, Adquisición de imágenes con Matlab, <http://www.elai.upm.es/spain/Investiga/GCII/personal/pbecerra/index.htm#LUZ%20ESTRUCTURADA>, 2009.