

Sistema de detección de carril para vehículos autónomos

Trabajo Práctico Final
Procesamiento Digital de Imágenes
Universidad Nacional del Sur

Autor: Nicolás Salomón
DNI: 40.086.088
Año: 2021



Índice

1. Introducción	1
2. Objetivos	2
2.1. Objetivo general.....	2
2.2. Objetivos específicos	2
3. Marco Teórico	3
3.1. Espacios Cromáticos	3
3.1.1. RGB	3
3.1.2. HSL	4
3.1.3. Escala de grises	6
3.2. Técnicas de Procesamiento de Imágenes.....	7
3.2.1. Filtros direccionales: Operador Sobel	7
3.2.2. Ecuación de Histograma Adaptativo Limitado por Contraste (CLAHE) ..	9
3.2.3. Procesamiento morfológico: Dilatación y Cierre	10
3.3. Cambio de perspectiva: Perspective Transform.....	12
4. Implementación	15
4.1. Preprocesamiento y binarización	15
4.1.1. Umbral de luminancia	15
4.1.2. Binarización de imágenes con una luminancia adecuada	15
4.1.3. Binarización de imágenes con baja luminancia	18
4.2. Cambio de Perspectiva	20
4.3. Determinación del color de cada línea	21
4.4. Histograma de carriles	21
4.5. Algoritmo de ventana deslizante	22
4.6. Generación de la mejor curva.....	23
4.7. Cálculo del radio de curvatura y del offset con respecto al centro del carril..	24
4.8. Resultado final	26
5. Conclusiones y Mejoras a Futuro	28
6. Bibliografía	29

Índice de Figuras

Figura 1. Ejemplo de un sistema de detección de carril.....	1
Figura 2. Espacio de color RGB	4
Figura 3. Espacio de color HSL	4
Figura 4. Rueda de matiz o tono.....	5
Figura 5. Escala de saturación	5
Figura 6. Escala de luminosidad	5
Figura 7. Escala de grises porcentual.....	6
Figura 8. Imagen RGB vs Escala de grises	7
Figura 9. Ejemplo de zona de variación brusca en la luminancia de una imagen.....	7
Figura 10. Resultados de aplicar el operador Sobel. a) Operador Sobel en la dirección X (Este-Oeste). b) a) Operador Sobel en la dirección Y (Norte-Sur). c) Combinación de operadores en ambas direcciones.	8
Figura 11. Histograma de Luminancia para una imagen con bajo nivel de luminancia... 9	
Figura 12. Histograma de Luminancia para una imagen con alto nivel de luminancia.... 9	
Figura 13. Resultado de aplicar el algoritmo CLAHE en una imagen con baja luminancia.	10
Figura 14. Dilatación del cuadrado gris empleando un elemento estructurante cuadrangular. El resultado final es el cuadrado gris de mayor dimensión que el original.11	
Figura 15. Erosión del cuadrado gris empleando un elemento estructurante cuadrangular. El resultado final es el cuadrado gris de menor dimensión que el original 11	
Figura 16. Cierre morfológico de A empleando el elemento estructurante circula B. ..	12
Figura 17. Vista en perspectiva de una cámara montada arriba de un vehículo.....	13
Figura 18. Cambio de perspectiva	13
Figura 19. Cambio de perspectiva. a) Imagen original junto con la zona de interés. b) Resultado final del cambio de perspectiva de la zona de interés.....	14
Figura 20. a) Imagen con baja luminancia en la zona de interés (mitad inferior). b) Imagen con luminancia correcta en la zona de interés.....	15
Figura 21. Resultado de aplicar el filtrado de color amarillo.	16
Figura 22. Resultado de aplicar el filtrado de color blanco.....	17
Figura 23. Resultado final al aplicar los filtros de amarillo y blanco.	17
Figura 24. Imagen original junto con su histograma de luminancia.	18
Figura 25. Resultado de aplicar CLAHE junto a su histograma de luminancias.	18

Figura 26. Resultado de aplicar el operador Sobel (Magnitud)	19
Figura 27. Resultado de aplicar el operador Sobel (Fase)	19
Figura 28. Resultado de la binarización al combinar los filtros de color amarillo y del operador Sobel.	20
Figura 29. Resultado final al aplicar la operación morfológica de cierre.....	20
Figura 30. Cambio de perspectiva para obtener una vista superior.	21
Figura 31. Histograma de carriles.....	22
Figura 32. Resultado final de aplicar el algoritmo de ventanas deslizantes.	23
Figura 33. Resultado de calcular la mejor curva en base a los resultados del algoritmo de ventanas deslizantes.....	24
Figura 34. Radio de curvatura.	25
Figura 35. Mediciones en [m] de la vista superior.	25
Figura 36. Resultado final del procesamiento - Imagen 1.....	26
Figura 37. Resultado final del procesamiento - Imagen 2.....	27
Figura 38. Resultado final del procesamiento - Imagen 3.....	27
Figura 39. Resultado final del procesamiento - Imagen 4.....	28

1. Introducción

Ante el constante avance actual de los sistemas de navegación autónomos, en particular aquellos orientados a la conducción de vehículos terrestres (por ejemplo, automóviles, colectivos, entre otros), el desarrollo de sistemas que, a través de múltiples sensores, permitan establecer su posición en el entorno y la de los objetos que lo rodean con el fin de tomar decisiones para la navegación, se convirtió en un área de especial interés.

Uno de los sistemas a los cuales se hace referencia es el sistema de detección de carril. El reconocimiento de las zonas en las cuales el vehículo puede transitar, así como también el tipo y color de la demarcación sobre la ruta empleando las imágenes obtenidas gracias a una cámara de video representa un desafío que será abordado en el presente trabajo.

El uso de cámaras de video como sensores principales en este tipo de sistemas no se trata de una decisión trivial, las mismas tienen considerables ventajas frente a otros sistemas (como los sensores LIDAR). Entre dichas ventajas podemos nombrar su bajo costo y peso, facilidad de montaje, junto con la considerable resolución de las cámaras actuales. Pero, por otro lado, las mismas poseen una desventaja principal, el considerable procesamiento computacional requerido para tratamiento de las imágenes.

Aun así, las ventajas descriptas anteriormente convierten a los sistemas de detección de carril empleando cámaras de video en un área muy interesante para su investigación y desarrollo.



Figura 1. Ejemplo de un sistema de detección de carril

2. Objetivos

El presente trabajo se realizó teniendo en cuenta los siguientes objetivos.

2.1. Objetivo general

- Desarrollo e implementación de un sistema de detección de carril para vehículos autónomos empleando procesamiento digital de imágenes.

2.2. Objetivos específicos

- Estudiar e implementar un preprocesamiento de las imágenes para facilitar el reconocimiento de las líneas pertenecientes a los carriles.
- Tomar las imágenes preprocesadas y encontrar las zonas donde se encuentran las líneas de los carriles.
- Trazar las líneas de los carriles demarcados en la ruta.
- Reconocer el tipo y color de las líneas demarcadoras del carril.
- Establecer el radio de curvatura de las líneas para mejorar el sistema de control del vehículo
- Establecer la posición actual del vehículo con respecto al centro del carril.
- Presentar por pantalla todos los resultados anteriores

3. Marco Teórico

A continuación, veremos los conceptos teóricos necesarios para la implementación del detector de carriles para vehículos autónomos.

3.1. Espacios Cromáticos

Es posible definir un espacio cromático como un método para interpretar el color, que en general emplea valores numéricos para determinar la intensidad de sus componentes

En la actualidad existen una gran cantidad de espacios cromáticos (o espacios de color) distintos, cada uno de ellos con mayores ventajas que el resto en ciertas aplicaciones. A continuación, se hará un repaso sobre los espacios de color más relevantes en este trabajo.

3.1.1. RGB

Seguramente en algún momento de nuestras vidas hemos escuchado el término RGB, muy utilizado en la actualidad en términos referidos a la tecnología de pantallas digitales, cámaras de fotos, luces LED, entre otros.

El término RGB hace referencia al espacio de color (o espacio cromático) del mismo nombre, el cual emplea el color Rojo (Red, en inglés), Verde (Green, en inglés) y Azul (Blue, en inglés) como colores primarios aditivos para crear otros colores. De hecho, la mezcla de estos colores en distintas intensidades permite obtener más de 16 millones de colores distintos. Esto significa que, en teoría, el aspecto de cualquier color se puede simular comenzando con el negro (ausencia de luz) y añadiendo determinadas proporciones de luz roja, verde y azul. Cuando las cantidades de rojo, verde y azul son iguales y en su máxima intensidad, se consigue el blanco.

Una ventaja del espacio cromático RGB es que presenta un modelo útil para diseñar dispositivos de producción en masa que, o bien imitan el ojo (como escáneres y cámaras digitales), o bien intentan engañar al ojo para hacerle creer que ve muchos colores (como sucede en pantallas digitales y televisores). Por ejemplo, una cámara digital imita la visión en color midiendo las intensidades de luz roja, verde y azul que se reflejan de los objetos que se están fotografiando, mientras que un monitor de ordenador simula los colores mostrando distintas intensidades de luz roja, verde y azul.

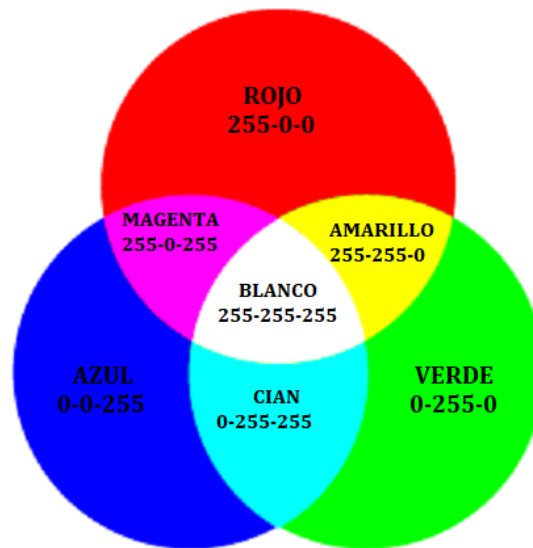


Figura 2. Espacio de color RGB

3.1.2. HSL

A diferencia del espacio de color RGB basado en la combinación de Rojo, Verde y Azul en diferentes intensidades, el espacio HSL basa su caracterización del color en término de su matiz (Hue, en inglés), saturación (Saturation, en inglés) y luminosidad (Lightness, en inglés)[1], parámetros mucho más útiles a la hora de identificar un mismo color ante cambios en la luminancia por ejemplo (una de las desventajas del espacio RGB), lo cual representa un escenario posible en la detección de carriles de una autopista.



Figura 3. Espacio de color HSL

El matiz o también denominada tono determina qué color del espectro es. El valor corresponde con el ángulo de la rueda de color de la Figura 3.

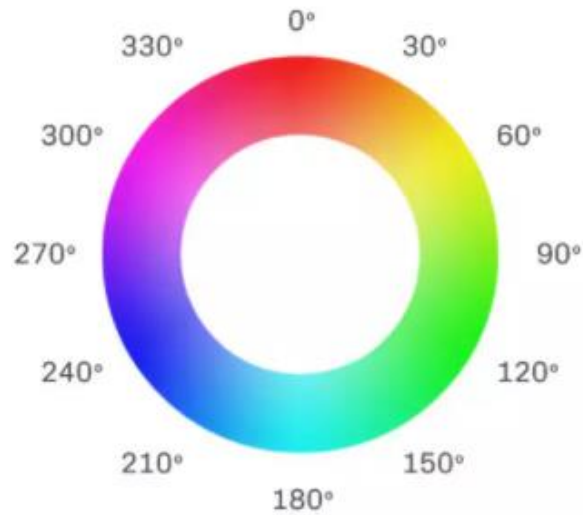


Figura 4. Rueda de matiz o tono

La saturación indica la intensidad de un tono concreto. Los valores van de 0% a 100%, siendo 100% el máximo de saturación posible y 0% el mínimo, que dará como resultado, gris. Lo descrito se observa en la Figura 5



Figura 5. Escala de saturación

La luminosidad, como su nombre lo indica, hace referencia a cuán claro u oscuro es un color. Si se quiere lograr un color más claro, se aumenta el valor de luminosidad, mientras que si se desea un loco más oscuro se disminuye el valor de luminosidad, tal como podemos observar en la Figura 6.



Figura 6. Escala de luminosidad

Las ecuaciones de conversión del espacio RGB al espacio HSL pueden observarse a continuación.

Considerando el valor de R, G y B de un pixel entre 0 y 1, su respectivo valor en el espacio HSL es el siguiente [2]:

$$H = \begin{cases} 0, & \text{si } MAX = MIN \\ \left(60 * \frac{G - B}{MAX - MIN} + 360\right) \bmod 360, & \text{si } MAX = R \\ \left(60 * \frac{B - R}{MAX - MIN} + 120\right), & \text{si } MAX = G \\ \left(60 * \frac{R - G}{MAX - MIN} + 240\right), & \text{si } MAX = B \end{cases}$$

$$L = 0.5 * (MAX + MIN)$$

$$S = \begin{cases} 0, & \text{si } MAX = MIN \\ \frac{MAX - MIN}{MAX + MIN} = \frac{MAX - MIN}{2L}, & \text{si } L < 0.5 \\ \frac{MAX - MIN}{2 - (MAX + MIN)} = \frac{MAX - MIN}{2 - 2L}, & \text{si } L > 0.5 \end{cases}$$

3.1.3. Escala de grises

Si bien la escala de grises puede no considerarse como un espacio cromático, es fundamental entender su composición. Es aquella en la que el valor de cada píxel representa solo una cantidad de luz, es decir, solo transporta información de intensidad del píxel. Las imágenes de este tipo, también conocidas como blanco y negro o monocromáticas, están compuestas exclusivamente por tonos de gris (ver Figura 7), que varían desde el negro en la intensidad más débil hasta el blanco en el más fuerte.

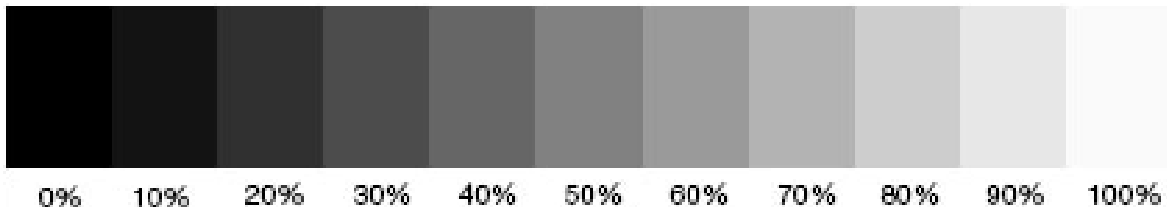


Figura 7. Escala de grises porcentual

La conversión desde el espacio RGB a escala de grises puede realizarse de forma muy sencilla empleando la siguiente ecuación, donde R, G y B representan el valor del píxel en el espacio RGB [3]:

$$GRAY = 0.299 * R + 0.587 * G + 0.114 * B$$

Los distintos valores por los cuales se multiplican los componentes de Rojo, Verde y Azul se encuentran fijados de acuerdo percepción relativa de la persona promedio del brillo de la luz roja, verde y azul.

Tomando como ejemplo distintos colores, al convertir dicha imagen a escala de grises obtenemos información muy valiosa en cuanto a la luminancia de cada píxel, considerando que mientras más cercano este el píxel al 100% de su valor, mayor luminancia posee. En la Figura 8 podemos apreciar lo descripto anteriormente, observando como aquellos colores más “brillosos” o más luminosos poseen un valor cercano al 100% en la escala de grises.



Figura 8. Imagen RGB vs Escala de grises

3.2. Técnicas de Procesamiento de Imágenes

3.2.1. Filtros direccionales: Operador Sobel

El operador Sobel es ampliamente utilizado en el procesamiento digital de imágenes para detectar bordes en las mismas en diferentes direcciones. Este método basa su funcionamiento en la operación de convolución entre la imagen y un kernel adecuado para lograr extraer de la misma únicamente aquellas zonas en las que existe una variación brusca de la luminancia, es decir, dos regiones de niveles de gris significativamente distintos. Aquellas zonas, justamente se corresponden con los bordes de una imagen como es posible apreciar en la Figura 9.

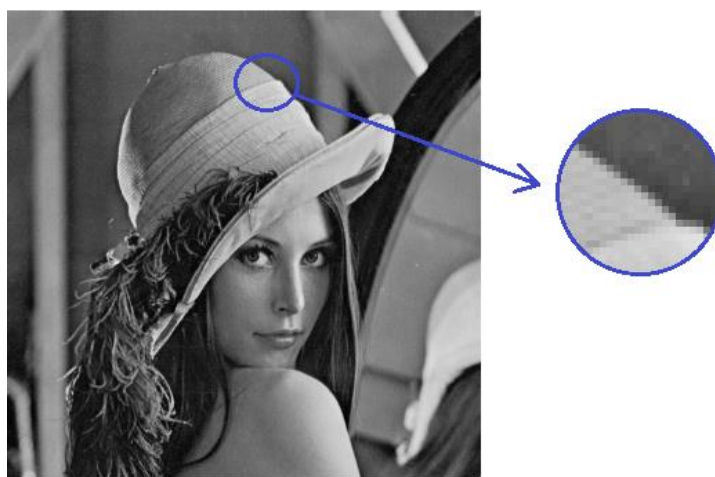


Figura 9. Ejemplo de zona de variación brusca en la luminancia de una imagen

Existen diferentes kernels que permiten extraer los bordes de una imagen. Uno de los más populares y que será empleado en el presente trabajo es el kernel u operador Sobel, el cual calcula el gradiente de la intensidad de una imagen en cada píxel. Así, para cada punto o píxel, este operador da la magnitud del mayor cambio posible, la dirección de este y el sentido desde oscuro a claro. El resultado muestra cómo de abruptamente o suavemente cambia una imagen en cada punto analizado y, en

consecuencia, cuán probable es que este represente un borde en la imagen y, también, la orientación a la que tiende ese borde.

Es posible obtener kernels para múltiples direcciones posibles (Norte-Sur, Noreste-Sudoeste, Este-Oeste, y así sucesivamente guiándose por los puntos cardinales).

En general, se emplean dos de estos kernels. Aquel que permite obtener el gradiente horizontal ($S_{horizontal}$) y vertical ($S_{vertical}$), y cuyos coeficientes se presentan a continuación:

$$S_{horizontal} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad S_{vertical} = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Luego ambos gradientes pueden ser combinados y así obtener la magnitud del gradiente, ver Figura 10:

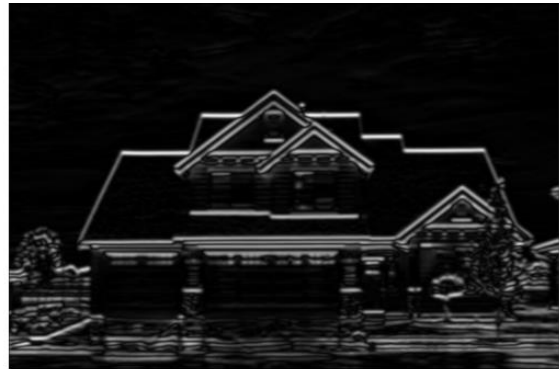
$$S = \sqrt{S_{horizontal}^2 + S_{vertical}^2}$$

Mientras que también es posible obtener la dirección (θ) de dicho gradiente:

$$\theta = \text{atan2}(S_{vertical}, S_{horizontal})$$



a)



b)



c)

Figura 10. Resultados de aplicar el operador Sobel. a) Operador Sobel en la dirección X (Este-Oeste). b) a) Operador Sobel en la dirección Y (Norte-Sur). c) Combinación de operadores en ambas direcciones.

3.2.2. Ecualización de Histograma Adaptativo Limitado por Contraste (CLAHE)

Uno de los problemas más frecuentes que se presentan en el procesamiento digital de imágenes y particularmente en nuestro caso de análisis (detección de carriles) es la variación de luminancia y crominancia en las imágenes a procesar. Será de particular interés resolver los problemas causados por las sombras y zonas oscuras de las imágenes, ya que dichas zonas dificultan el reconocimiento de líneas y características de la imagen, lo que cual representa nuestra principal finalidad.

Para resolver esta problemática existen diversas técnicas cuya finalidad es la de ecualizar el histograma de luminancia de una imagen, de esta forma logramos que la imagen tenga un histograma equiparado en cuanto a los niveles de luminancia.

En sencillas palabras, un histograma de luminancia es una representación gráfica de los niveles de luz de una imagen, en la cual la escala vertical representa la cantidad de píxeles, y la horizontal representa el nivel de luminancia de dichos píxeles.

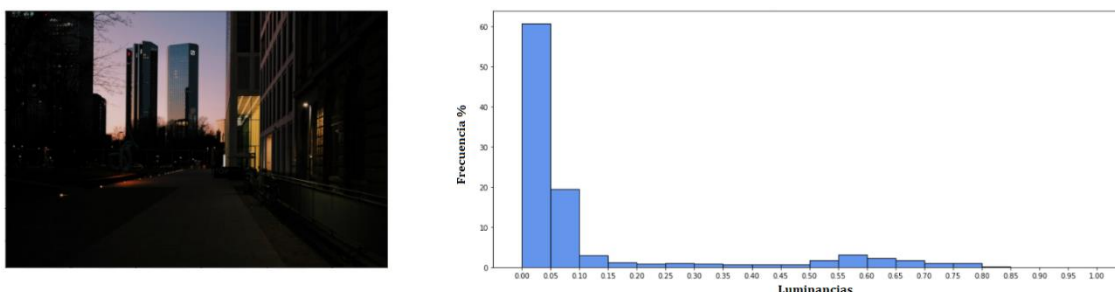


Figura 11. Histograma de Luminancia para una imagen con bajo nivel de luminancia.

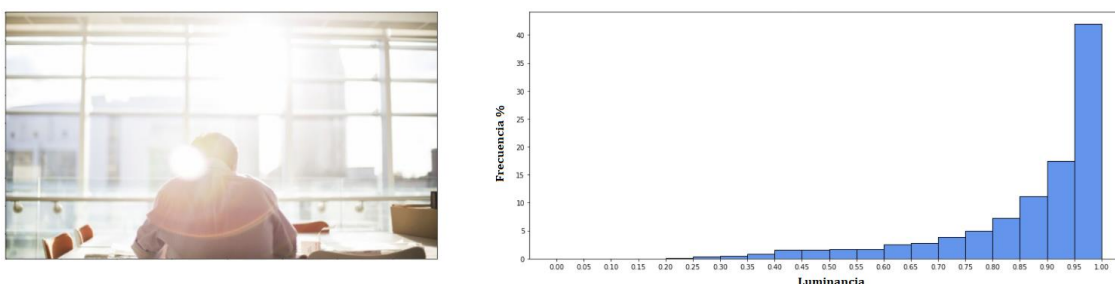


Figura 12. Histograma de Luminancia para una imagen con alto nivel de luminancia

Una de las técnicas de ecualización nombradas anteriormente, y que se emplea en el presente trabajo, es la Ecualización de Histograma Adaptativo Limitado por Contraste (del inglés, Contrast Limited Adaptive Histogram Equalization (CLAHE)).

Al realizar una ecualización del histograma considerando toda la imagen, provoca que el contraste se sobreamplifique en zonas que ya poseen un contraste adecuado, dando resultados poco satisfactorios para ciertas aplicaciones. El algoritmo CLAHE divide las imágenes en regiones contextuales, llamadas mosaicos (del inglés, tiles) y aplica la ecualización del histograma a cada una. Esto nivela la distribución de los valores de gris utilizados y, por lo tanto, hace que las características ocultas de la imagen sean más visibles. Luego, los mosaicos vecinos se combinan mediante una interpolación bilineal para eliminar los límites artificiales creados por cada mosaico [4].

El resultado de aplicar el algoritmo de CLAHE se puede apreciar en la Figura 13. Se observa como los detalles en las zonas oscuras son mucho más visibles.

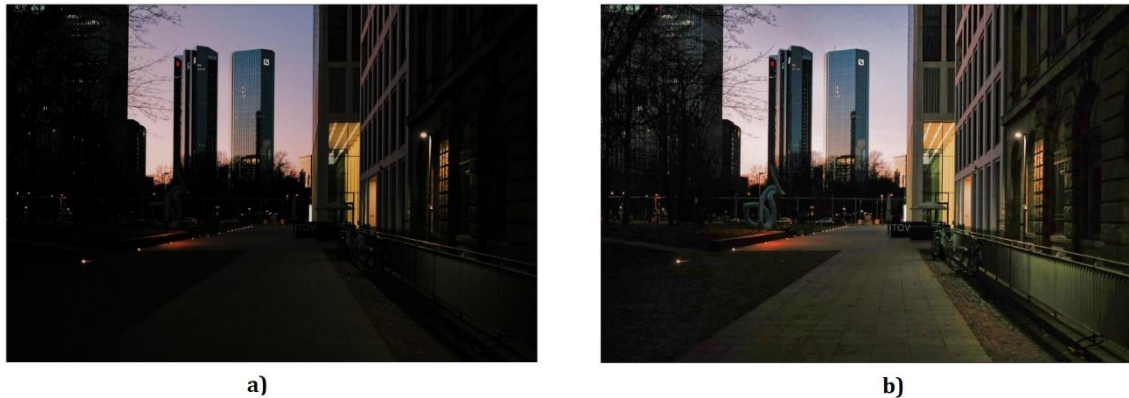


Figura 13. Resultado de aplicar el algoritmo CLAHE en una imagen con baja luminancia.

3.2.3. Procesamiento morfológico: Dilatación y Cierre

El procesamiento morfológico de imágenes es una herramienta para la extracción de componentes de la imagen que sean útiles en la representación y descripción de regiones. En otras palabras, las operaciones morfológicas simplifican las imágenes y conservan las principales características de forma de los objetos, también se utilizan para tareas de pre y post procesamiento como ser el filtrado morfológico y el adelgazamiento y engrosamiento.

La morfología matemática se basa en operaciones de teoría de conjuntos. En el caso de imágenes binarias, los conjuntos tratados son subconjuntos de Z^2 y en el de las imágenes en escala de grises, se trata de conjuntos de puntos con coordenadas en Z^3 . Por ejemplo, el conjunto de todos los píxeles negros de una imagen binaria (1 = blanco, 0 = negro), es una descripción morfológica de la imagen donde cada elemento de un conjunto es un par de coordenadas de un pixel blanco de la imagen. En el caso de imágenes en escala de grises, un elemento del conjunto está formado por las coordenadas del pixel, y su nivel de gris, por consiguientes pertenecen al espacio Z^3 .

DILATACIÓN

La dilatación se corresponde con una de las operaciones morfológicas sobre una imagen y se describe como un crecimiento de píxeles. Se basa en definir un elemento estructurante (por ejemplo, una ventana de 3x3 píxeles, o una circunferencia de un cierto radio) y un punto de inserción (por ejemplo, el pixel central de la ventana o la circunferencia), luego se desplaza dicho elemento y se marca con 1 la parte del fondo de la imagen que toque un pixel que forma parte de la región. Esto permite que aumente un pixel alrededor del elemento estructurante de cada región y así poder incrementar dimensiones, lo cual ayuda a rellenar hoyos dentro de la región. Un ejemplo del proceso de dilatación puede observarse en la Figura 14.

La dilatación de A por el elemento estructurante B se define por:

$$A \oplus B = \bigcup_{a \in A} B_a$$

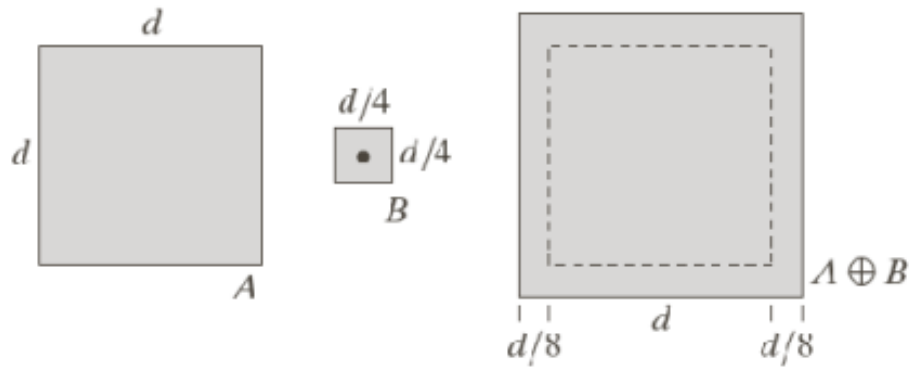


Figura 14. Dilatación del cuadrado gris empleando un elemento estructurante cuadrático. El resultado final es el cuadrado gris de mayor dimensión que el original.

EROSIÓN

La erosión es otra operación morfológica básica que puede realizarse sobre una imagen. A diferencia de la dilatación, se desplaza el elemento estructurante y se marca con 0 la parte del fondo de la imagen que toque un pixel que forma parte de la región. Esto permite que se reduzca un pixel alrededor del elemento estructurante de cada región y así poder decrementar dimensiones, lo cual ayuda a “afinar” los objetos de una imagen binaria. Un ejemplo del proceso de dilatación puede observarse en la Figura 15.

La erosión de A por el elemento estructurante B se define por:

$$A \ominus B = \overline{\overline{A} \oplus \overline{B}}$$

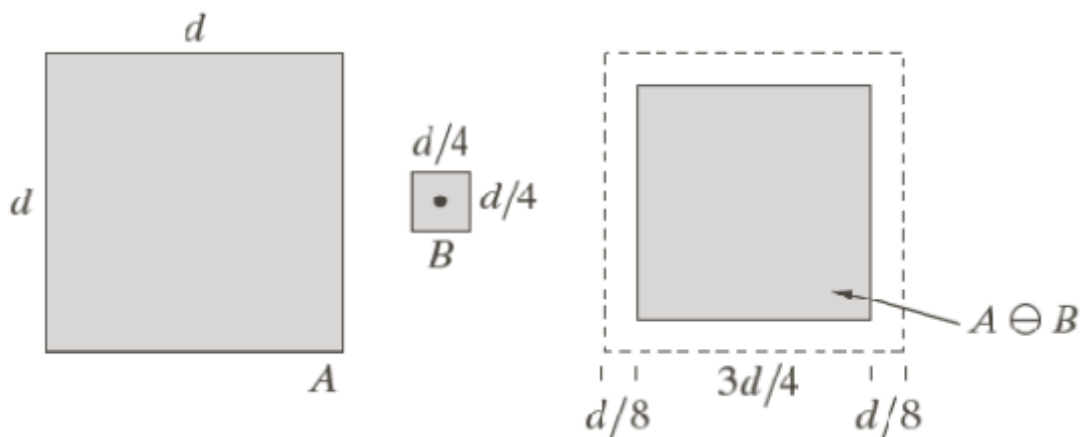


Figura 15. Erosión del cuadrado gris empleando un elemento estructurante cuadrático. El resultado final es el cuadrado gris de menor dimensión que el original

CIERRE

La operación morfológica de cierre surge como una combinación de las operaciones de dilatación seguida de una operación de erosión. Es el complemento de la unión de todas las traslaciones de B que no se solapan con A, siendo a la imagen y B el elemento estructurante. Esta operación también tiende a suavizar contornos de objetos, pero rellena agujeros más pequeños que el elemento estructural.

El cierre de A por el elemento estructurante B se define por:

$$(A \oplus B) \ominus B$$

El resultado del proceso de cierre morfológico puede apreciarse en la Figura 16.

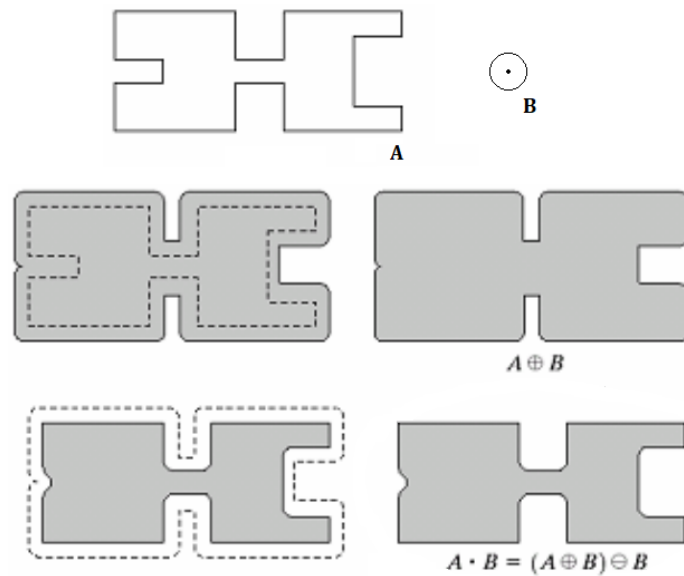


Figura 16. Cierre morfológico de A empleando el elemento estructural círculo B.

3.3. Cambio de perspectiva: Perspective Transform

Uno de los principales desafíos a la hora de analizar imágenes obtenidas de una cámara ubicada arriba de un vehículo, para luego detectar los carriles de una autopista o ruta, es lograr independizar el procesamiento de las mismas del efecto de perspectiva generado.

Desde la perspectiva de dicha cámara las líneas de los carriles poseen una forma trapezoidal, como puede apreciarse en la Figura 17, lo que dificulta entre otras cosas, el reconocimiento de las líneas de carril o el cálculo del radio de curvatura, ya que, desde la vista de la cámara, las líneas de los carriles parecen encontrarse en un punto lejano.



Figura 17. Vista en perspectiva de una cámara montada arriba de un vehículo

Una forma de superar este desafío sería obtener una vista desde arriba. Dicha vista también es conocida como “vista de ojo de ave”, gracias a la cual se remueve el efecto de perspectiva y las líneas de los carriles pasan a ser paralelas entre sí.

Para generar esta vista superior de la imagen existen diversos modelos disponibles, algunos se basan en parámetros de calibración de la cámara (por ejemplo, el Inverse Perspective Mapping”), otros emplean una matriz de transformación afín para mapear un conjunto de 4 puntos de la imagen original, con coordenadas (u, v) , a un conjunto de otros 4 puntos en la vista superior (x, y) . En este trabajo se empleará la segunda técnica debido a su simplicidad y flexibilidad dado que no requiere de conocer los parámetros de calibración de la cámara (ver Figura 18).

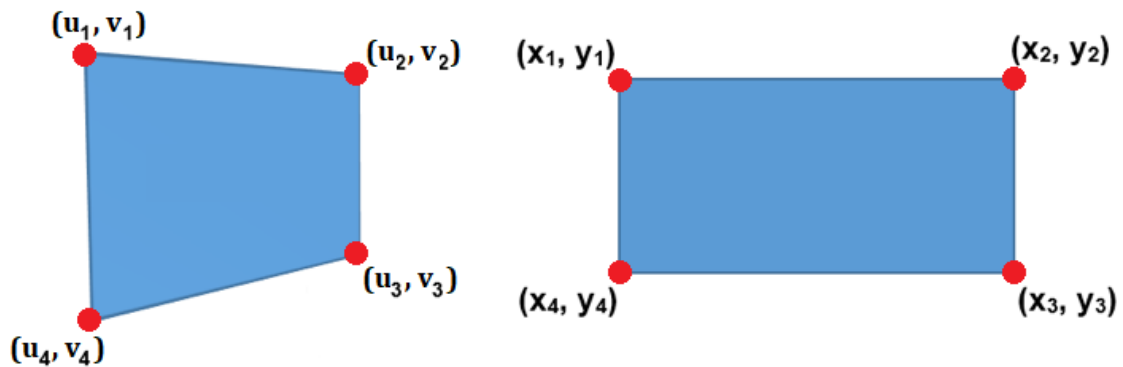


Figura 18. Cambio de perspectiva

El modelo mencionado puede representarse matemáticamente con la siguiente ecuación [5]:

$$[x, y, w] = [u, v, 1] * \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

Donde $\{a_{11}, a_{12}, a_{13}, a_{21}, a_{22}, a_{23}, a_{31}, a_{32}, a_{33}\}$ son los parámetros o coeficientes de la matriz de transformación relacionados con la traslación, rotación y escalado de la

imagen, (u, v) son los vértices del cuadrilátero en la imagen original, y (x, y) , los vértices del cuadrilátero de salida. Para calcular los ocho parámetros o coeficientes, se necesitan cuatro pares de vértices de puntos de correspondencia, (u_i, v_i) , y (x_i, y_i) .

Usualmente se toma $a_{33} = 1$, mientras que los coeficientes restantes se determinan resolviendo el sistema de ecuaciones lineales producto de reemplazar los 4 puntos de partida y de destino en la ecuación anterior:

$$\begin{bmatrix} u_0 & v_0 & 1 & 0 & 0 & 0 & -u_0 * x_0 & -v_0 * x_0 \\ u_1 & v_1 & 1 & 0 & 0 & 0 & -u_1 * x_1 & -v_1 * x_1 \\ u_2 & v_2 & 1 & 0 & 0 & 0 & -u_2 * x_2 & -v_2 * x_2 \\ u_3 & v_3 & 1 & 0 & 0 & 0 & -u_3 * x_3 & -v_3 * x_3 \\ 0 & 0 & 0 & u_0 & v_0 & 1 & -u_0 * y_0 & -v_0 * y_0 \\ 0 & 0 & 0 & u_1 & v_1 & 1 & -u_1 * y_1 & -v_1 * y_1 \\ 0 & 0 & 0 & u_2 & v_2 & 1 & -u_2 * y_2 & -v_2 * y_2 \\ 0 & 0 & 0 & u_3 & v_3 & 1 & -u_3 * y_3 & -v_3 * y_3 \end{bmatrix} * \begin{bmatrix} a_{11} \\ a_{21} \\ a_{31} \\ a_{12} \\ a_{22} \\ a_{32} \\ a_{13} \\ a_{23} \end{bmatrix} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ y_0 \\ y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

Luego de obtener los parámetros o coeficientes de la matriz de transformación, cualquier punto (u, v) de la imagen original puede transformarse a un punto (x, y) empleando:

$$x = \frac{u * a_{11} + v * a_{21} + a_{31}}{u * a_{13} + v * a_{23} + 1}$$

$$y = \frac{u * a_{12} + v * a_{22} + a_{32}}{u * a_{13} + v * a_{23} + 1}$$

Los resultados de aplicar esta transformación afín pueden apreciarse en la Figura 19.

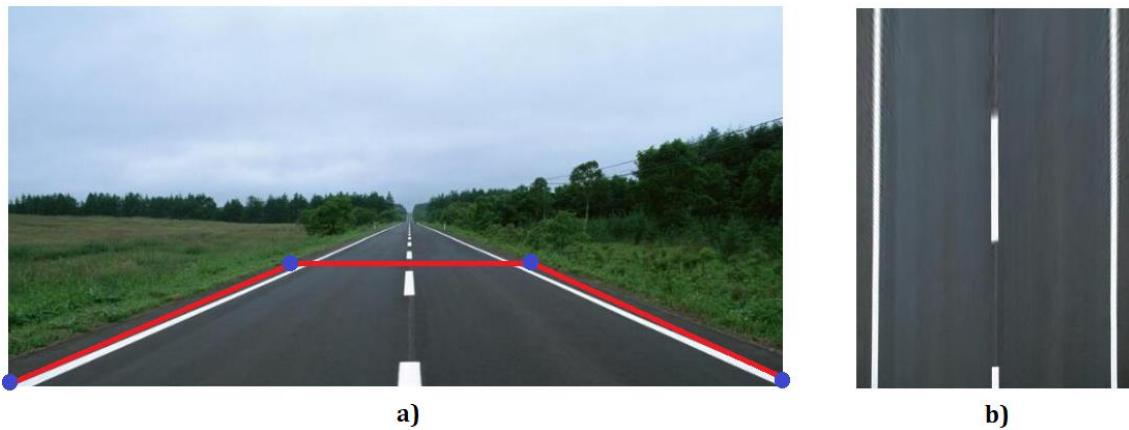


Figura 19. Cambio de perspectiva. a) Imagen original junto con la zona de interés. b) Resultado final del cambio de perspectiva de la zona de interés

4. Implementación

4.1. Preprocesamiento y binarización

Como se explicó en la sección 3, existen muchas maneras de representar los colores en una imagen, empleando diferentes espacios de cromáticos, y con diferentes aplicaciones en cada uno de ellos.

Para los fines de este trabajo, la detección de carriles en una autopista o ruta, será necesario emplear más de un espacio de color y luego combinar los resultados para obtener una detección correcta.

4.1.1. Umbral de luminancia

El primer paso a realizar será una simple prueba para detectar cuando una imagen posee baja luminancia y en base a ello elegir el método de binarización correcto, con el fin de obtener una imagen binarizada (solo con pixeles blancos y negros) en la cual los pixeles blancos representen únicamente los carriles de la ruta. Dicha prueba se basa en realizar un análisis sobre la intensidad de sus pixeles, en este caso emplearemos la escala de grises explicada en la sección 3. Una vez que poseemos la imagen en escala de grises calculamos el valor medio de los pixeles que se encuentran en la mitad inferior de la misma, dado que la cámara se encuentra en la parte superior del vehículo, no es de importancia los pixeles más allá de la mitad de la imagen (ver Figura 20)

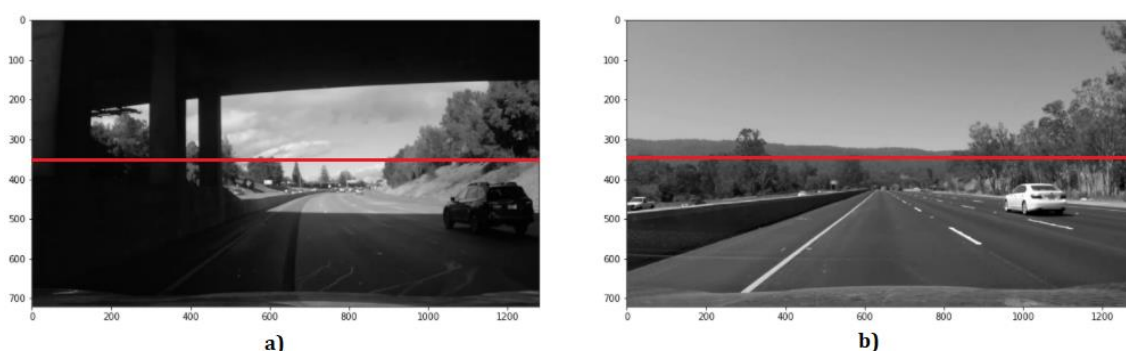


Figura 20. a) Imagen con baja luminancia en la zona de interés (mitad inferior). b) Imagen con luminancia correcta en la zona de interés.

A continuación, se fija un valor umbral adecuado (en base a pruebas realizada con imágenes con baja y con correcta luminancia), en nuestro caso se tomó un valor umbral de 70, por debajo del cual se considera que la imagen posee un nivel de luminancia bajo. En la Figura 19-a, el valor medio obtenido fue de 41, mientras que en la imagen b), dicho valor fue de 90.

4.1.2. Binarización de imágenes con una luminancia adecuada

En base al resultado anterior, si la imagen posee un nivel de luminancia adecuado se procesa a realizar un filtrado para aquellas zonas de color amarillo y blanco, dado que las líneas demarcadoras de carriles en autopistas y rutas sólo poseen dichos colores.

Para realizar el filtrado del color amarillo, se empleó el espacio de color HLS, dado que el mismo tiene un mejor rendimiento ante pequeños cambios de iluminación o efectos del reflejo de la luz solar, además de poder determinar fácilmente el rango de valores pertenecientes al color amarillo (ver Figura 4 en la sección 3). Los rangos tomados que dieron mejores resultados en diversas pruebas realizadas fueron los siguientes:

H: 10 – 50

L: 0 – 200

S: 50 – 255

El resultado de aplicar este filtrado sobre la Figura 20 b) puede apreciarse en la Figura 21.

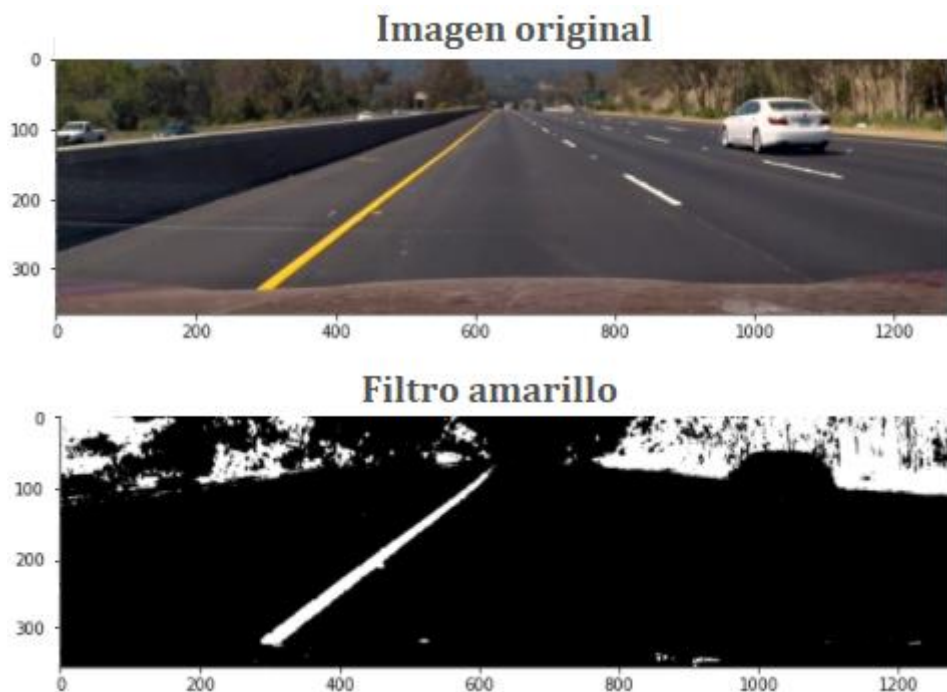


Figura 21. Resultado de aplicar el filtrado de color amarillo.

Por otro lado, las líneas blancas se filtraron empleando un umbral sobre la imagen en escala de grises, el mismo fue fijado tal que, para todos aquellos pixeles que posean una intensidad mayor a 180, se les asigna el valor 255, mientras que para aquellos que posean una intensidad menor a dicho umbral, se les asigna el valor 0. El resultado de aplicar este umbral sobre la Figura 20-b puede apreciarse en la Figura 22, destacando que también se detectan las líneas amarillas dado que el umbral empleado en la imagen de escala de grises, incluye la intensidad de los pixeles amarillos al convertirlos en escala de grises.

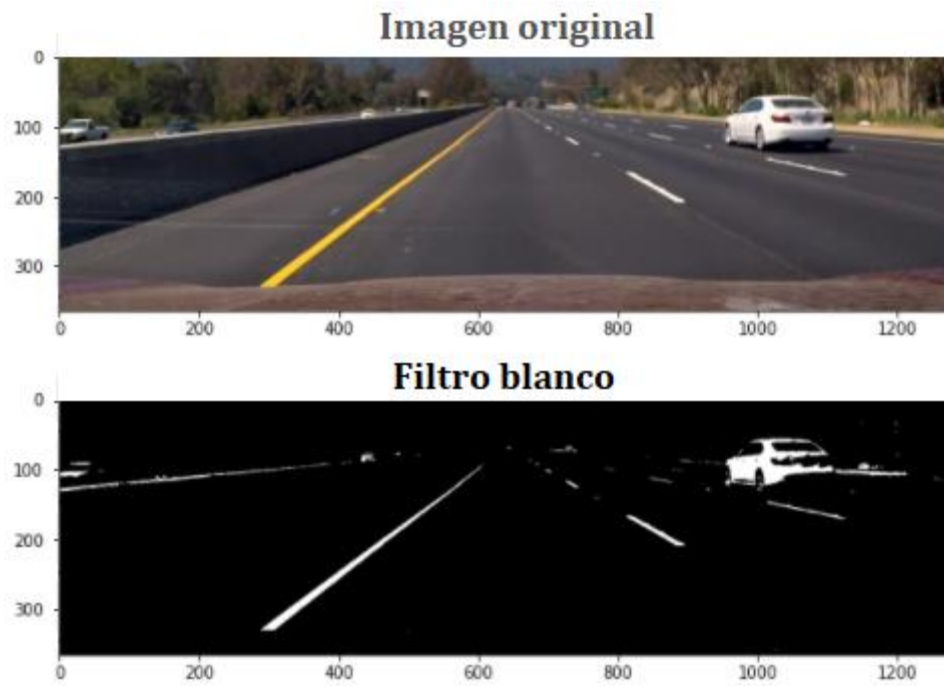


Figura 22. Resultado de aplicar el filtrado de color blanco.

Por último, ambas imágenes son combinadas empleando la función OR entre ambos resultados, el resultado puede apreciarse en la Figura 23.

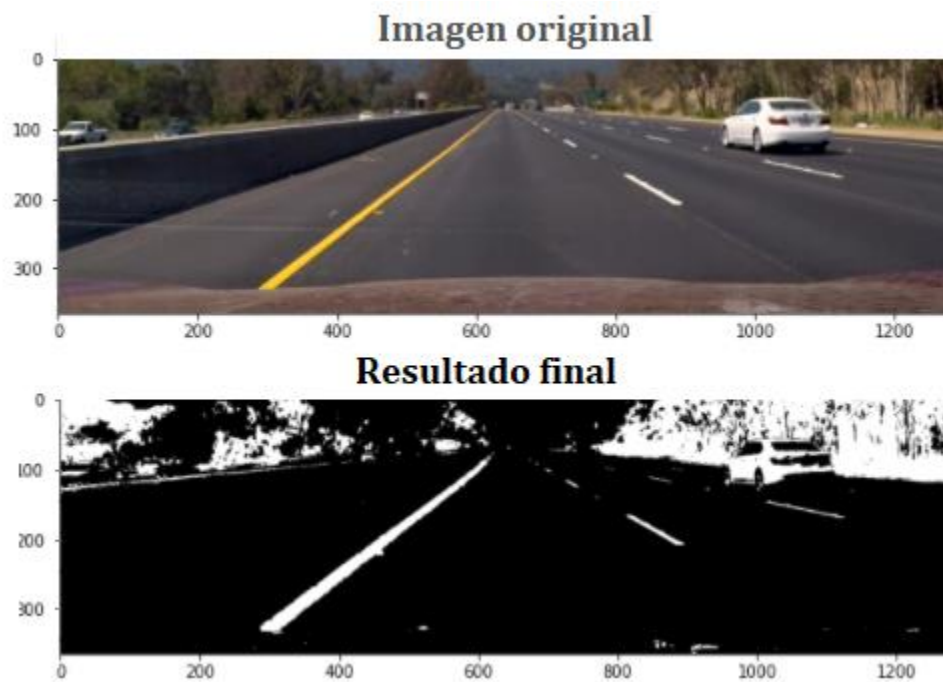
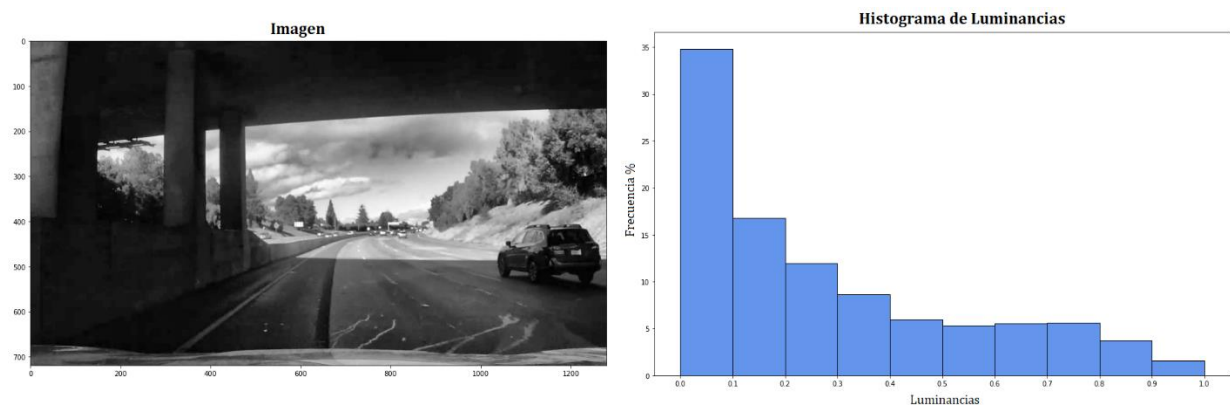
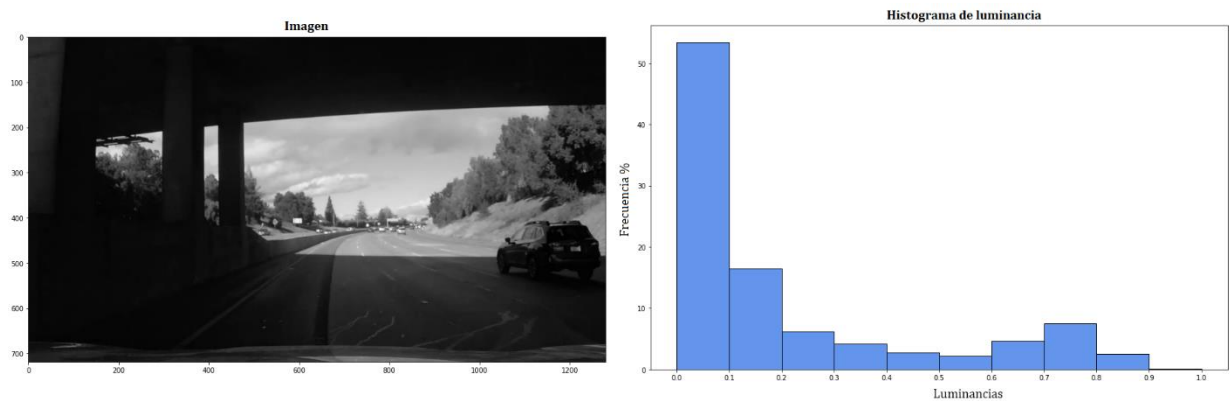


Figura 23. Resultado final al aplicar los filtros de amarillo y blanco.

4.1.3. Binarización de imágenes con baja luminancia

Para aquellas imágenes que poseen baja luminancia (Figura 20-a), el proceso de filtrado por color no posee un buen resultado por sí solo, debido a que en las zonas oscuras la cámara no llega a distinguir bien los colores, por ello, para este tipo de imágenes el procesamiento y la binarización será un poco más complejo.

En primer lugar, buscaremos ecualizar el histograma de luminancia de la imagen a través de la Ecualización de Histograma Adaptativo Limitado por Contraste, CLAHE. El mismo logra aumentar la luminancia en las zonas oscuras de la imagen, sin saturar las zonas con una luminancia correcta. El funcionamiento de este algoritmo se explicó en la sección 3.2.2 y los resultados del mismo aplicado a la Figura 20-b pueden apreciarse en la Figura 24 y 25, junto con su respectivo histograma de luminancias, se observa un histograma mucho más equilibrado en cuanto a las frecuencias porcentuales de las distintas luminancias.



Luego de obtener una imagen con un histograma de luminancia más equilibrado, las líneas de los carriles son bastante más distinguibles.

Como siguiente paso se realizará una detección de líneas empleando el operador Sobel descrito en la sección 3.2.1. estableciendo umbrales en la detección para filtrar las líneas de interés. En base a ensayos realizados, se determina un umbral igual a 30 para la magnitud del operador Sobel, mientras que se fija un rango de entre 0.8 [rad] (45°

aproximadamente) y 1.4 [rad] (80° aproximadamente) en la detección de líneas, dado que las líneas de carriles en general poseen estos rangos en los ángulos formados por las mismas y el eje horizontal.

El resultado de aplicar el operador Sobel junto con el filtro en la magnitud establecida se aprecia en la figura 26, observando cómo se obtiene una detección aceptable de las líneas de carril, en conjunto con otras líneas que luego serán filtradas.

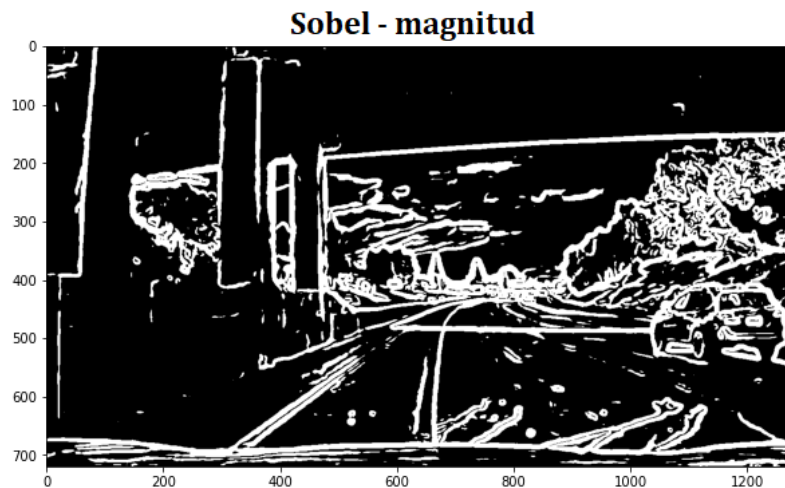


Figura 26. Resultado de aplicar el operador Sobel (Magnitud)

Por otro lado, el resultado del filtrado aplicando el operador Sobel considerando la fase del mismo puede apreciarse en la Figura 27. Se observa como aquellas líneas que no se encuentren en un ángulo distinto entre los 45° y 80° poseen un valor igual a 0.



Figura 27. Resultado de aplicar el operador Sobel (Fase)

Al combinar ambos resultados empleando la función booleana AND, junto con el filtrado de líneas de color amarillo (idéntico al empleado en la sección 4.1.2) se obtiene la Figura 28. Se observa como en la zona de interés (frente al vehículo) las líneas del carril son fácilmente identificables.

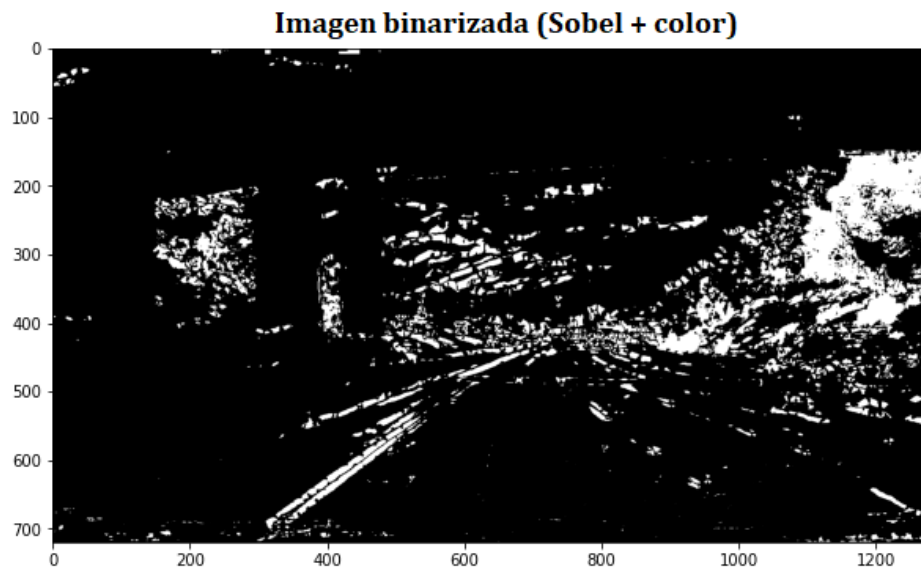


Figura 28. Resultado de la binarización al combinar los filtros de color amarillo y del operador Sobel.

Por último, para mejorar aún más el proceso de reconocimiento de las líneas de los carriles, se aplica la operación morfológica de cierre, descrita en la sección 3.2.3. El resultado de esta operación se aprecia en la Figura 29.

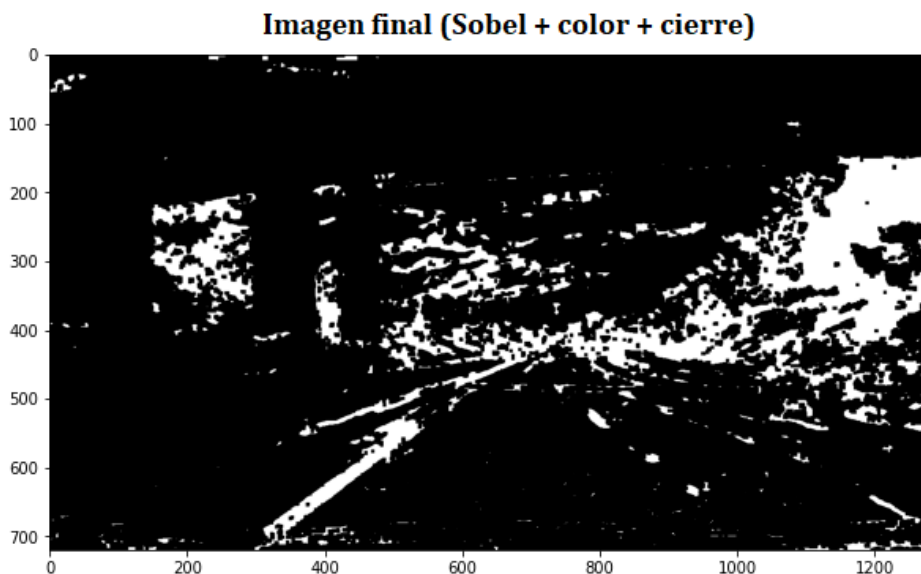


Figura 29. Resultado final al aplicar la operación morfológica de cierre.

4.2. Cambio de Perspectiva

Una vez filtrada y binarizada la imagen, se procede a realizar el cambio de perspectiva y obtener una vista superior de la misma, con el fin de evitar los errores que pueden introducir los efectos de perspectiva de la imagen, tal como se explicó en la sección 3.3. Para demostrar el funcionamiento de esta transformación, aplicaremos la misma a la imagen 20-b, luego de su respectiva binarización, como se explicó en la sección anterior. El resultado de la misma se puede apreciar en la Figura 30, observando cómo se puede reconocer la curva hacia la derecha que se aproxima al frente del vehículo.

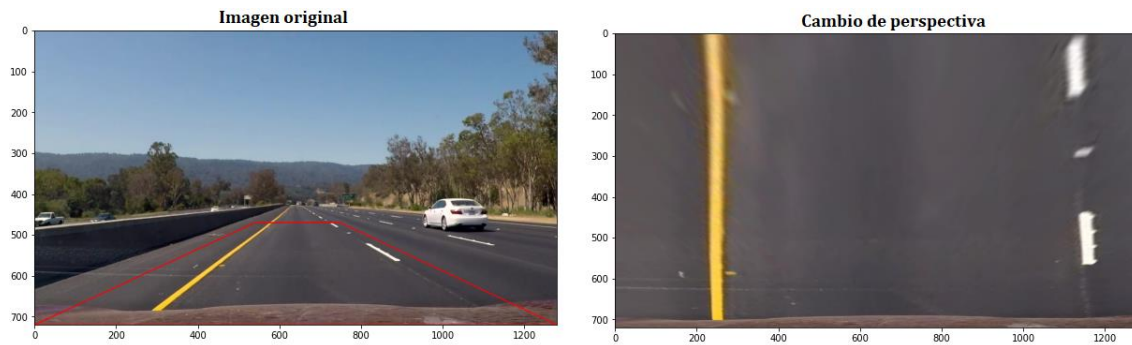


Figura 30. Cambio de perspectiva para obtener una vista superior.

4.3. Determinación del color de cada línea

Dado que las líneas indicadoras de cada carril en una autopista o ruta solo poseen dos colores posibles, amarillo o blanco, en el presente trabajo se propone considerar en un comienzo a todas las líneas como blancas hasta que se demuestre que las mismas son amarillas por medio de un filtrado similar al aplicado en la sección 4.1.

El primer paso a realizar será convertir la imagen del espacio RGB al espacio HLS, para luego aplicar el mismo filtro explicado en la sección 4.1, el mismo poseía los siguientes valores límites:

H: 10 – 50

L: 0 – 200

S: 50 – 255

Para acelerar el procesamiento de la misma y del detector en general, se considera como línea derecha, toda detección encontrada en la mitad derecha de la imagen, mientras que se considera como línea izquierda toda detección efectuada sobre la mitad izquierda.

De esta forma se analiza cada imagen o frame de un video para determinar el color de las líneas derecha e izquierda demarcadoras de los carriles.

4.4. Histograma de carriles

El paso siguiente será determinar aquellas áreas de la imagen que posean una alta concentración de píxeles blancos. Para ello se calculará y graficará el histograma de la imagen binarizada, previa transformación de perspectiva para obtener su vista superior.

Idealmente el histograma debería tener dos picos en aquellas zonas donde existe una mayor concentración de píxeles blancos, dichos picos representarán las líneas de los carriles izquierdo y derecho respectivamente. Por ello, se identifica la posición de dichos picos, como puede apreciarse en la Figura 31.

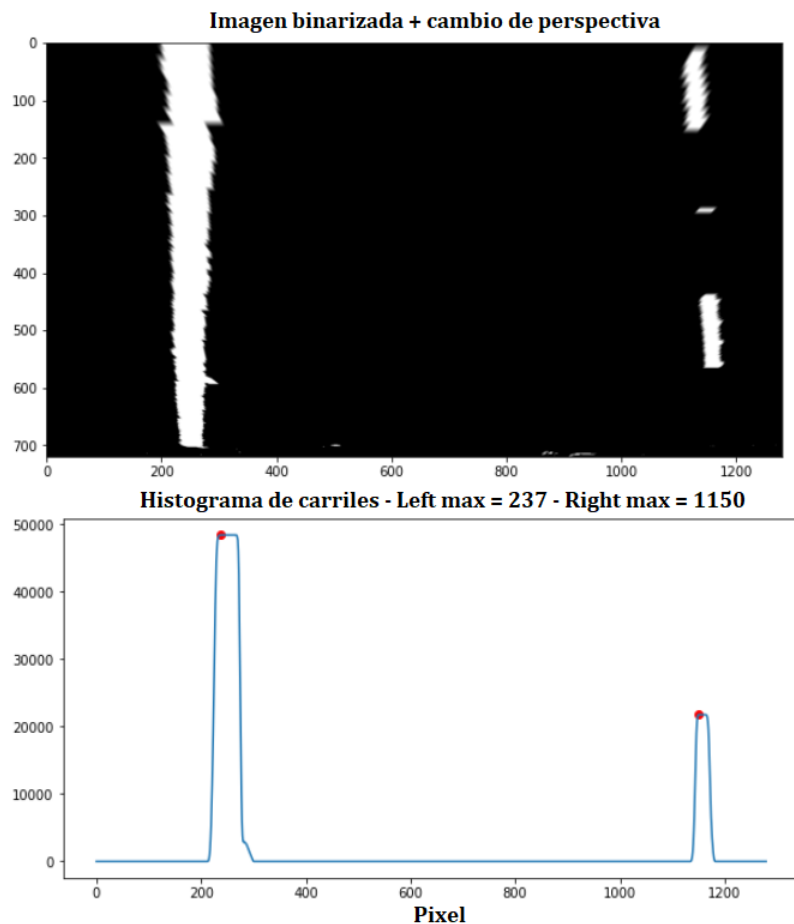


Figura 31. Histograma de carriles.

4.5. Algoritmo de ventana deslizante

El siguiente paso se basa en emplear un método que permita ir identificando las líneas derecha e izquierda de los carriles en cada imagen o frame de un video para luego conocer su posición y trazar una curva que represente dichas líneas.

Básicamente, el algoritmo de ventanas deslizantes se basa en determinar un punto de inicio en la zona inferior de la imagen, para lo cual nos valdremos del histograma calculado en la sección anterior, junto con una determinada dimensión para las ventanas, e ir desplazándose hacia arriba en la imagen.

La primera ventana cuenta la cantidad de pixeles blancos que se encuentran dentro de ella. Si dicha cantidad supera un umbral, la posición central de la próxima ventana se desplaza hacia la posición lateral media de los pixeles blancos detectados. Si la cantidad de pixeles contados no supera el umbral, la próxima ventana conserva la misma posición. Este proceso continúa hasta que la ventana llega al límite superior de la imagen. Por otro lado, por cada pixel detectado dentro de la ventana, se agrega su posición a una lista para luego trazar la mejor curva que pase por todos los puntos de dicha lista.

Además, se aprovecha el desplazamiento de las ventanas para determinar si las líneas son continuas o discontinuas, un dato sumamente importante en la conducción de vehículos, para determinar si es seguro o no el cambio de carril.

El resultado de este algoritmo se aprecia en la Figura 32, notando como las ventanas “siguen” correctamente a las líneas demarcadoras de carriles, aunque estas tengan una cierta curvatura.

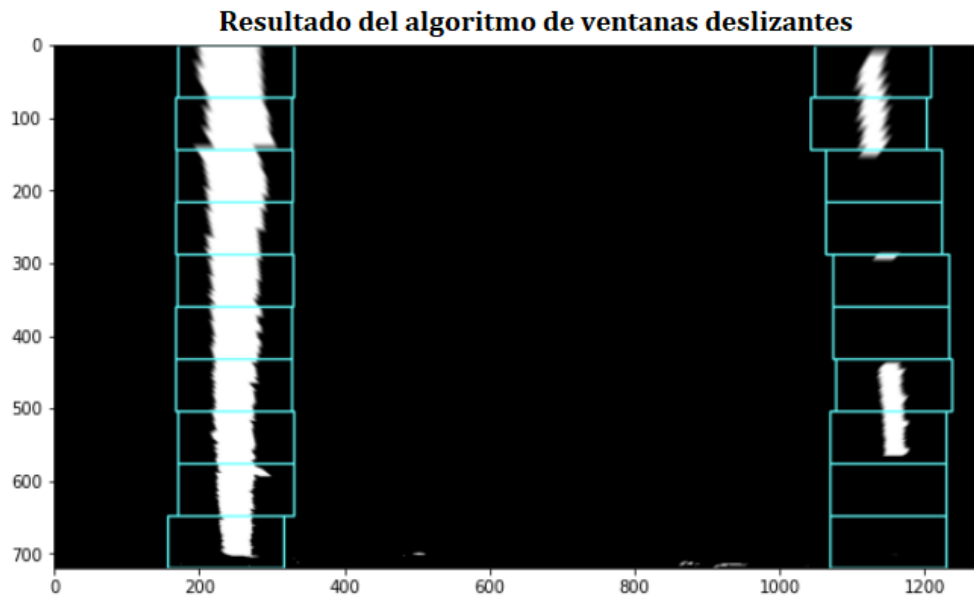


Figura 32. Resultado final de aplicar el algoritmo de ventanas deslizantes.

4.6. Generación de la mejor curva

Una vez determinadas las posiciones de los píxeles pertenecientes a las líneas de los carriles por medio del algoritmo de ventanas deslizantes, se aplica una regresión polinomial para encontrar los coeficientes de una curva que mejor se adapte a los valores de x e y encontrados para cada uno de los píxeles, considerando la ecuación de la curva:

$$Y = a * x^2 + b * x + c$$

En la Figura 33 se observa el resultado obtenido para la mejor curva en base al resultado obtenido por el algoritmo de ventanas deslizantes, en conjunto con el color de las líneas detectadas. Puede apreciarse como se obtiene un resultado bastante bueno.

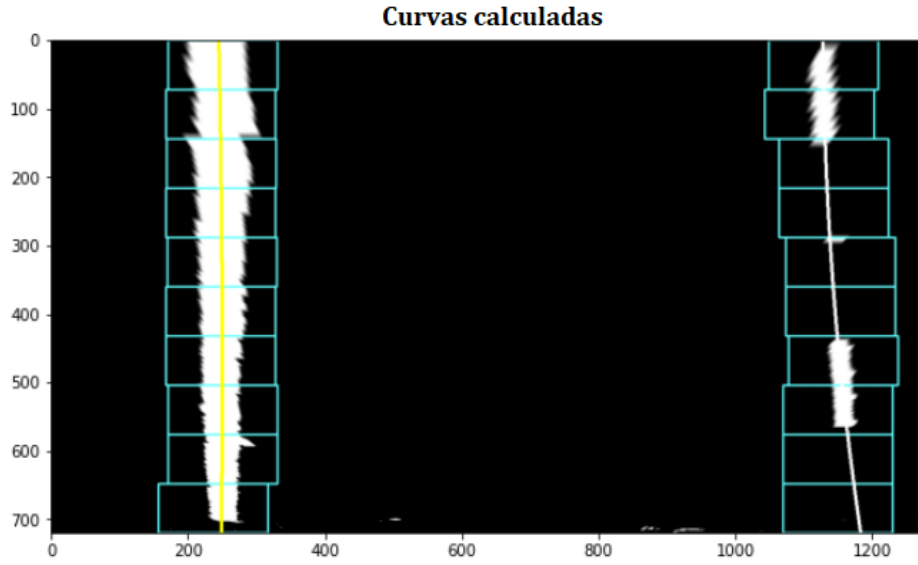


Figura 33. Resultado de calcular la mejor curva en base a los resultados del algoritmo de ventanas deslizantes.

Por último, para agregar cierta inmunidad con respecto a errores que pueden ocurrir al calcular los coeficientes de la curva, por ejemplo, al procesar un video y que el frame siguiente no posea líneas de carril o no sean distinguibles al aplicar el preprocesado y la binarización explicada en la sección 4.1, los coeficientes obtenidos se almacenan para luego utilizarse en los próximos frames si es que sucede el problema mencionado anteriormente.

4.7. Cálculo del radio de curvatura y del offset con respecto al centro del carril

El radio de curvatura de las líneas demarcadoras de carriles es un parámetro muy importante para el control de la dirección de un vehículo. Siendo nuestra función

$$Y = a * x^2 + b * x + c$$

El mismo se define de la siguiente forma:

$$R_c = \frac{\left[1 + \left(\frac{dY}{dx}\right)^2\right]^{\frac{3}{2}}}{\left|\frac{d^2Y}{dx^2}\right|} = \frac{[1 + (2 * a * x + b)^2]^{\frac{3}{2}}}{|2 * a|}$$

Una representación gráfica puede apreciarse en la Figura 34.

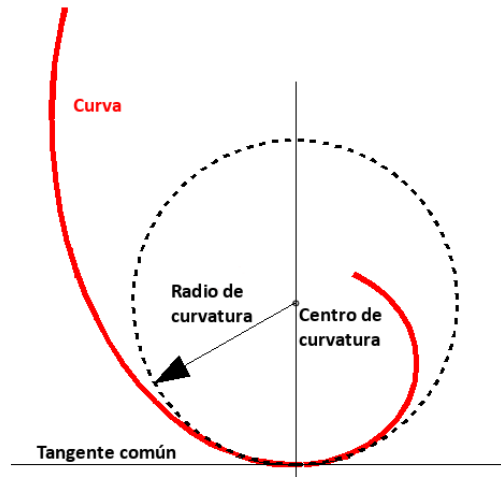


Figura 34. Radio de curvatura.

Por último, para poder calcular el radio de curvatura y expresarlo en unidades del Sistema Internacional, es necesario realizar una pequeña conversión entre pixeles y metros. Para ello, medimos el ancho y alto en pixeles de las líneas demarcadoras de carriles desde la vista superior, para luego comparar estas medidas junto con las regulaciones gubernamentales del lugar donde se filmó el video o se obtuvo la imagen a analizar.

En nuestro caso, la Figura 30 se obtuvo de un frame de un video de una autopista en EEUU, donde las líneas demarcadoras discontinuas tienen un largo aproximado de 3 metros y un ancho de carril de 3,7 [m] [6] (ver Figura 35).

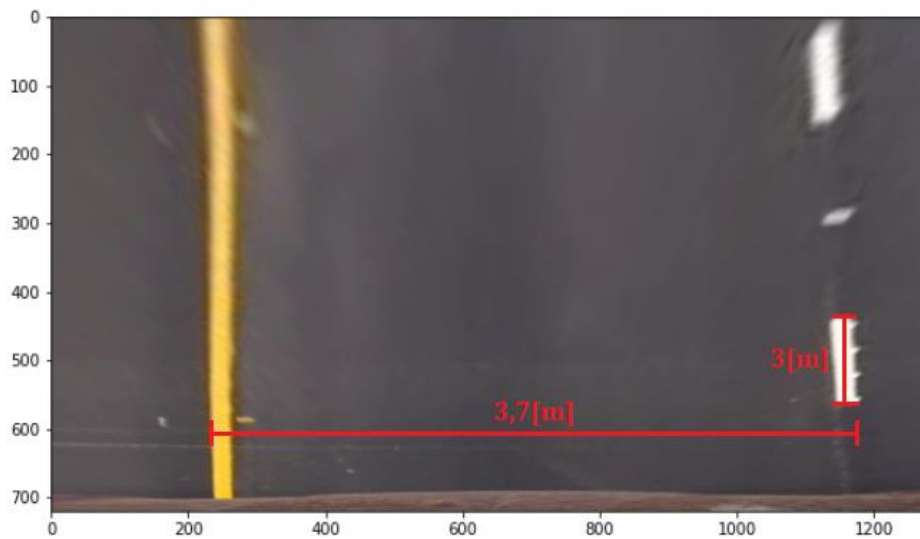


Figura 35. Mediciones en [m] de la vista superior.

En base a esto, se establece una relación de:

$$\text{metros por pixel para el eje } y = \frac{3[m]}{60 [\text{pixel}]} = 0,05 \left[\frac{m}{\text{pixel}} \right]$$

$$\text{metros por pixel para el eje } x = \frac{3,7[m]}{450 [\text{pixel}]} = 0,0082 \left[\frac{m}{\text{pixel}} \right]$$

En base a las relaciones anteriores es posible determinar el radio de curvatura en unidades de metros. Para ello, obtenemos la ecuación de la curva descrita en la sección 4.6 considerando dichas relaciones, para luego calcular el radio mencionado.

Por último, la desviación del vehículo con respecto al centro del carril se lleva a cabo haciendo la suposición de que la cámara del auto se encuentra centrada en dicho vehículo, por lo que el centro del carril se puede obtener como el ancho total de la imagen dividida entre 2. Por otro lado, la posición actual del auto se obtiene gracias a la curva obtenida anteriormente para calcular el radio de curvatura. Finalmente, la diferencia entre ambos valores nos da la desviación del vehículo con respecto al centro del carril. Un valor negativo indica que es necesario realizar una corrección y desplazar el vehículo hacia la derecha y, por consiguiente, un valor positivo indica que es necesario realizar una corrección y desplazar el vehículo hacia la izquierda.

4.8. Resultado final

Todos los resultados descritos anteriormente son presentados por pantalla en diferentes imágenes de carriles en la Figuras 36 a 39.



Figura 36. Resultado final del procesamiento - Imagen 1.

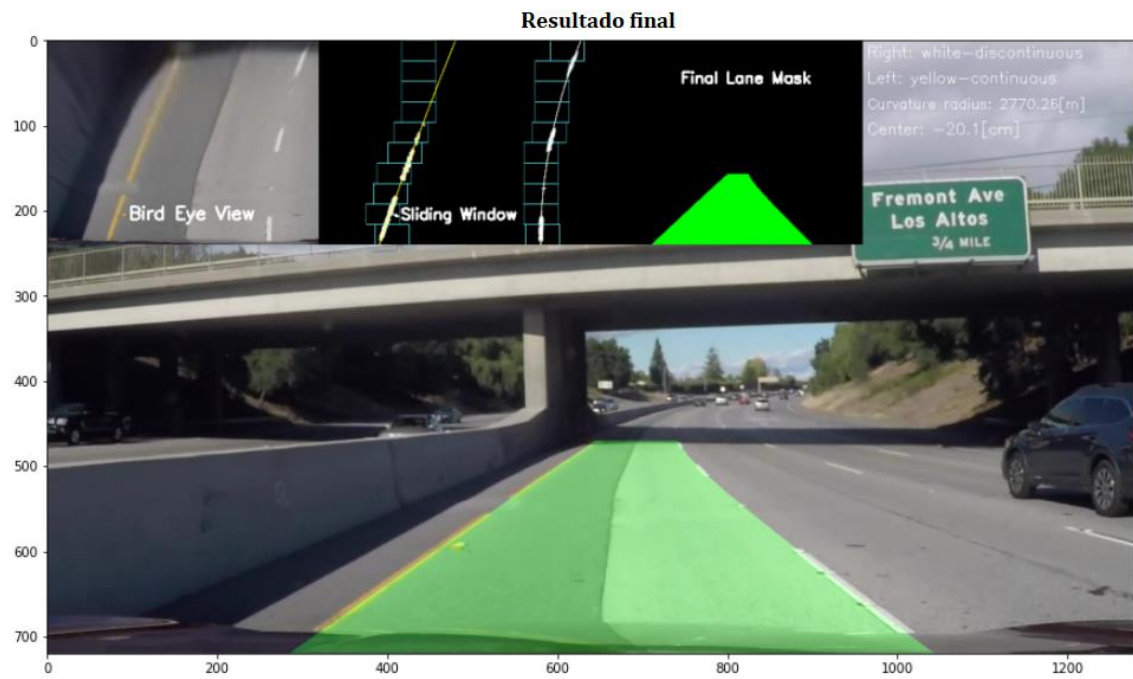


Figura 37. Resultado final del procesamiento - Imagen 2

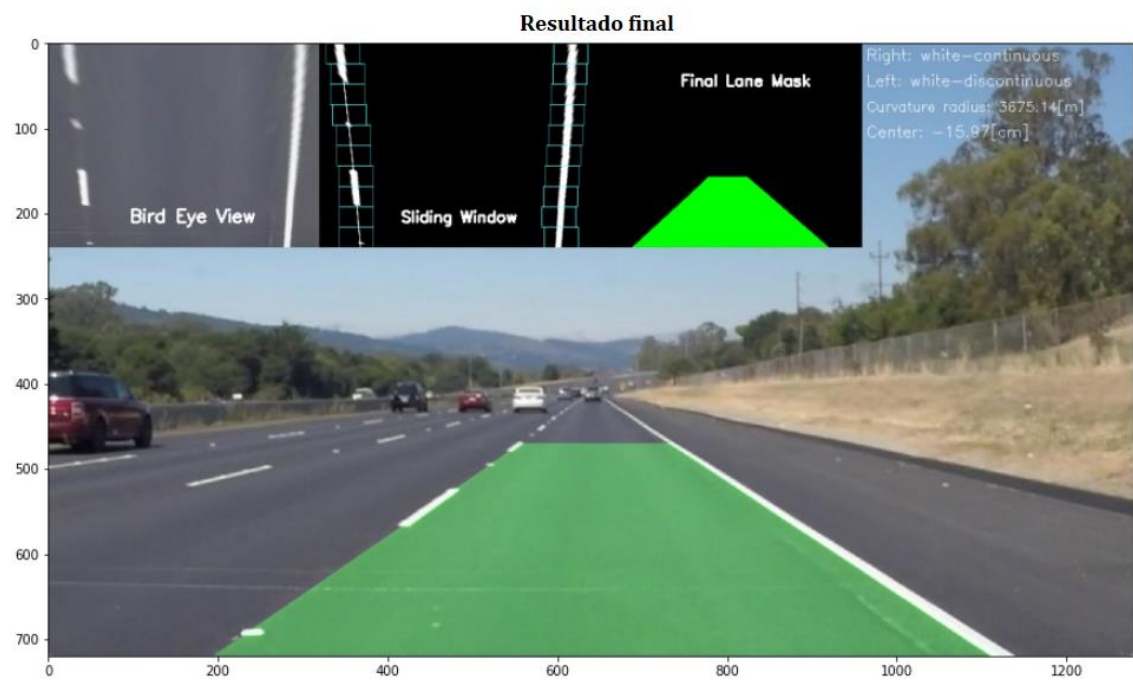


Figura 38. Resultado final del procesamiento - Imagen 3

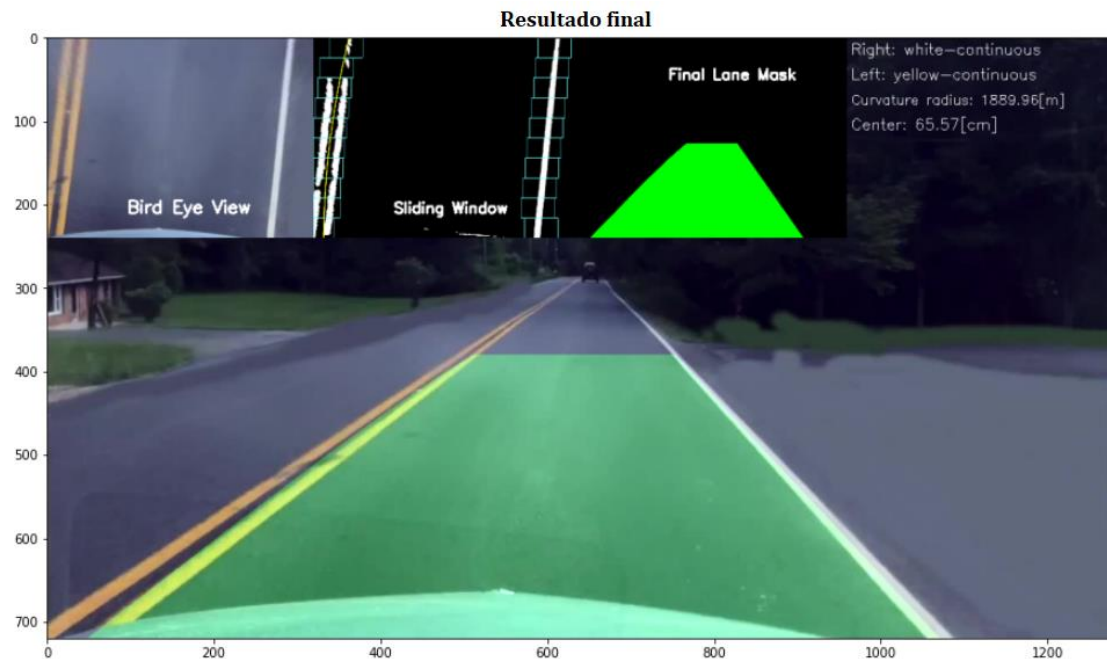


Figura 39. Resultado final del procesamiento - Imagen 4

Por último, se adjuntan dos videos demostración del resultado obtenido al procesar cada frame. Se obtuvo una velocidad de procesamiento en torno a los 10 [FPS], haciendo uso de un CPU.

5. Conclusiones y Mejoras a Futuro

Luego de desarrollar el presente detector de carriles para vehículos autónomos se puede concluir en que se cumplieron los objetivos generales y específicos planteados en la sección 2.

Fue posible implementar numerosas técnicas de procesamiento y segmentación de imágenes, así como también experimentar el uso de espacios cromáticos distintos al ya conocido espacio RGB para lograr, luego de un largo proceso de reinención y prueba, el presente detector.

A pesar de las problemáticas que se fueron presentando, en particular para aquellas imágenes que poseían un nivel de luminancia bajo y que ocasionaban problemas al aplicar el método explicado en la sección 4.1.2, fue posible superar la misma gracias a la implementación del método de la sección 4.1.3.

Se logró desarrollar un detector que funciona con tiempos de procesamiento aceptables para haber sido ejecutado empleando una CPU, dicho tiempo seguramente puede disminuir considerablemente al emplear memorias gráficas (GPU).

Por último, como trabajo y mejora a futuro se plantea el uso de Machine Learning e Inteligencia Artificial, para construir un detector aún más robusto frente a cambios en la iluminación, roturas o manchas en el pavimento, caminos mucho más sinuosos, entre otros, el cual en combinación con el presente trabajo conformará un detector mucho más confiable para su uso en vehículos autónomos.

6. Bibliografía

- [1] John C. Russ and F. Brent Neal. 2015. The Image Processing Handbook (5th. ed.). CRC Press, Inc., USA.
- [2] Saravanan, G. & Yamuna, G. & Nandhini, S.. (2016). Real time implementation of RGB to HSV/HSI/HSL and its reverse color space models. 462-466.
doi:10.1109/ICCSP.2016.7754179.
- [3] Chandran, Saravanan. (2010). Color Image to Grayscale Image Conversion. 196 - 199. doi: 10.1109/ICCEA.2010.192.
- [4] P. Musa, F. A. Rafi and M. Lamsani, "A Review: Contrast-Limited Adaptive Histogram Equalization (CLAHE) methods to help the application of face recognition," 2018 Third International Conference on Informatics and Computing (ICIC), 2018, pp. 1-6, doi: 10.1109/IAC.2018.8780492.
- [5] Monteiro, P., Ascenso, J. & Pereira, F. Perspective transform motion modeling for improved side information creation. EURASIP J. Adv. Signal Process. 2013, 189 (2013). <https://doi.org/10.1186/1687-6180-2013-189>
- [6] Roadway Design Manual (2020). Chapter 4: Horizontal Alignment. url: http://onlinemanuals.txdot.gov/txdotmanuals/rdw/horizontal_alignment.htm#BGBHGEGC