

# Modelo basado en visión por computadora para detección de carriles viales para la autonomía de vehículos

Juan Antonio Guerrero-Ibáñez<sup>1</sup>, Juan José Contreras-Castillo<sup>1</sup> y Pedro C. Santana-Mancilla<sup>1</sup>

<sup>1</sup> Facultad de Telemática – Universidad de Colima, Avenida Universidad 333, Colonia Las Víboras, Colima, Colima, 28040. México  
antonio\_guerrero@uacol.mx, juancont@uacol.mx, psantana@uacol.mx

**Resumen.** La industria automotriz se está centrando en desarrollar vehículos con un cierto nivel de autonomía hasta lograr el vehículo completamente autónomo. El objetivo global es generar un entorno de conducción más amigable y seguro tanto para los conductores, los pasajeros y los peatones. En este artículo presentamos un modelo basado en visión por computadora para detección de carriles que contribuya a proporcionar un cierto nivel de autonomía al vehículo. Los resultados obtenidos al aplicar el modelo mostraron una alta exactitud de detección de carriles, lo que permite contribuir al proceso de automatización del coche.

**Abstract.** The automotive industry is focusing on developing vehicles with some autonomy level until achieving the fully autonomous vehicle. The global objective is to create a more friendly and safe driving environment for drivers, passengers, and pedestrians. This paper presents a computer vision-based model for lane detection that contributes to providing a certain level of autonomy for the vehicle. The results obtained when the model was applied a high accuracy in the lane detection process, allowing a contribution to the vehicle automation process.

**Keywords:** coches, niveles de autonomía, visión por computadora, algoritmos, seguridad.

## 1. Introducción

De acuerdo con la Organización Mundial de la Salud (OMS), a nivel mundial alrededor de 1.25 millones de personas mueren como consecuencia de accidentes de tránsito [1]. De acuerdo al Departamento de Transporte de los Estados Unidos, las estadísticas de accidentes muestran que el 93% de los accidentes de coches se deben a errores humanos [2]. Aunado a esto tenemos los problemas que enfrenta la sociedad moderna con respecto a niveles de contaminación y congestión vial, pérdida de tiempo en las carreteras, por mencionar algunos.

Por estas razones, en las últimas décadas los fabricantes de vehículos se han enfocado en proporcionar a los usuarios una experiencia de manejo más agradable, segura y

amigable con el ambiente. Los esfuerzos están dirigidos a generar paulatinamente coches con cierto nivel de autonomía hasta que se alcance la autonomía total del vehículo. La idea es generar vehículos con capacidad de moverse y transportar de una manera segura, sin que un humano lo conduzca.

Con esto en mente, la Sociedad de Ingenieros Automotrices (SAE – Por sus siglas en inglés) [3] definió 5 niveles para determinar la autonomía de un coche de acuerdo al grado de involucramiento del conductor en el proceso de conducción (Figura 1).



**Fig. 1.** Niveles de automatización de un vehículo definidos por el SAE.

Uno de los puntos más relevantes para el éxito de la autonomía de los vehículos es la percepción. La percepción es la tarea de sensar el entorno de conducción, y consiste en un subconjunto de tareas tales como clasificación y detección de objetos, estimación de posición de los objetos, localización y mapeo simultáneo de objetos [4]. La percepción es un proceso crucial en la autonomía de los vehículos para una operación confiable y segura ya que todos los datos que se reciban del proceso de percepción son procesados por la computadora central y/o enviados a la nube y se utilizan para la toma de decisiones que defina el comportamiento del vehículo ante las situaciones detectadas (tráfico, cruce de peatones, señales de tráfico, etc).

El presente artículo es parte de un proyecto de investigación denominado DeOAVe, una plataforma para detección de Objetos aplicado a la Autonomía de Vehículos. En específico, en este trabajo se presenta un algoritmo para la detección de carriles en avenidas y carreteras basado en visión computacional.

## 2. Estado del Arte

La detección de carriles es un componente crítico en la autonomía de los coches y es uno de los tópicos de investigación más importantes para entender el escenario de conducción autónoma. Una vez que las posiciones de líneas de los carriles son obtenidas, el coche conocerá por donde desplazarse y evitar los riesgos de moverse a otros carriles o peor aún salirse de la carretera.

Los métodos para detección de carriles se dividen en dos categorías: métodos basados en sensores y métodos basados en visión. Los primeros utilizan dispositivos como LiDAR, radar o sensores láser así como GPS [5-6]. Por otro lado, los métodos basados en visión utilizan sistemas de cámaras y son divididos en dos categorías: los métodos basados en modelos, los cuales crean un modelo matemático de la estructura de la

carretera usando coordenadas geométricas de la cámara y la carretera como datos de entrada [7]; mientras que los métodos basados en características pueden distinguir áreas con marcas basados en características de la carretera como por ejemplo el color, el gradiente o el ángulo [8].

El problema de la detección de carriles se ha definido como un problema de segmentación [9]. La segmentación consiste en dividir la imagen en regiones de interés llamadas segmentos. Por ejemplo se han propuesto soluciones basados en segmentación y un método para la generación automática de datos etiquetados [10]. Por otro lado, en [11] se propone el uso del método de segmentación para realizar la detección de carriles y posteriormente aplican una transformación de perspectiva aprendida para desempeñar una corrección de carril.

En este trabajo se presenta un algoritmo de detección de carriles robusto y efectivo que utiliza el método basado en características. Primeramente se aplica una región de interés (ROI) a la imagen de entrada para eliminar la región que no pertenezca a la carretera. Posteriormente se aplica un mapeo inverso de perspectiva (IPM) para delimitar la región de visión. Como tercer paso se aplica el algoritmo de detección de segmentos de línea (LSD) para identificar todos los segmentos de línea de la imagen y se aplica un filtro para eliminar los segmentos de línea falsos de la imagen. Finalmente se aplica IPM para generar la imagen de salida.

### **3. Descripción de la propuesta**

El proceso de detección de carriles implica muchos retos a resolver, uno de ellos es evaluar los tipos de condiciones que se pueden encontrar durante el proceso de conducción (físicas de la carretera, ambientales, estado físico del conductor). Por ejemplo, las marcas de las carreteras o avenidas pueden variar enormemente. Las carreteras pueden ser marcadas por líneas sólidas bien definidas, líneas segmentadas, reflectores circulares, barreras físicas o en el peor de los casos con nada. Por otro lado, la superficie de la carretera o avenida puede estar compuesta de pavimento oscuro o claro o una combinación de ambos. Uno de los retos para detección exacta de líneas es controlar el ruido que aparece en la imagen de entrada, tales como sombras de objetos, marcas de frenados, entre otros.

La presente propuesta se centra en la detección de carriles en tiempo real mediante el uso de cámara. Generalmente las líneas para representar los carriles son líneas blancas o amarillas que se pintan en el pavimento de la carretera. La pregunta a resolver es ¿cómo detectar las marcas de los carriles? Para poder detectar líneas se definió el proceso que se muestra en la figura 2 y se describe en las siguientes sub-secciones.

#### **3.1 Captura de imagen**

Los datos que se utilizan de entrada son una secuencia de imágenes a color las cuales son capturadas desde la cámara del vehículo. La cámara se montó en el espejo retrovisor en la línea central. Toda la información que se captura se procesa por la computadora en tiempo real y se almacena en la memoria.

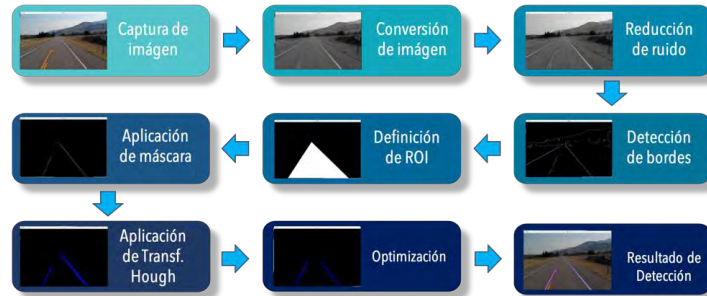


Fig. 2. Representación de los pasos de la propuesta de solución.

### 3.2 Conversión de imagen

Debido a que el procesamiento de imágenes a color implica muchas dificultades para la detección de bordes y el tiempo de procesamiento, las imágenes se convierten a escala de grises.

### 3.3 Reducción de ruido

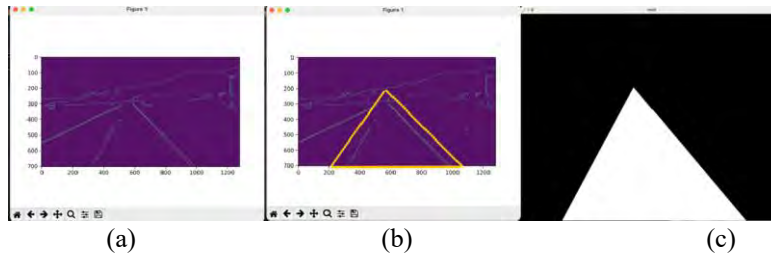
El ruido es un problema para todos los sistemas. La presencia de ruido dificulta la detección correcta de los bordes, por lo que es indispensable aplicar algún mecanismo para remover el ruido. El objetivo es suavizar la imagen para facilitar la detección del mayor número de bordes. Para lograr esto se filtra cualquier ruido de la imagen que pudiera generar un borde falso a través de un filtro Gaussiano. Cada imagen se almacena como una colección de píxeles discretos a escala de grises los cuales se representan mediante un número que representa la brillantez del píxel. Para suavizarlo se modifica el valor del píxel con el valor promedio de la intensidad de los píxeles a su alrededor. El promedio se realiza mediante el kernel. El kernel se configuró a un tamaño de 5x5 y se aplica a todos los píxeles de la imagen.

### 3.4 Detección de bordes

Para identificar el perímetro de las líneas se realiza un análisis de contraste de forma entre la superficie de la carretera y las líneas pintadas o la superficie no pavimentada. Con esto se reduce la cantidad de datos de aprendizaje requeridos y se simplifica la imagen considerablemente. Para llevar a cabo esta actividad se utilizó el método de Canny el cual es un algoritmo de múltiples etapas que permite detectar una amplia gama de bordes en imágenes.

### 3.5 Detección de área de interés

Debido a que dentro del escenario de conducción existen diferentes objetos a parte de las líneas, lo que se hace es aplicar un mecanismo que permita ignorar los objetos no deseados del escenario de conducción. Para resolver este problema, se disminuye el área de interés con la que se desea trabajar. El objetivo de esta etapa es definir el área donde se localizan las líneas de los carriles, la cual se conoce como Región de Interés (ROI- por sus siglas en inglés). Primero, se define una gráfica donde se puedan representar las coordenadas para poder definir los puntos del área de interés (figura 3a). Posteriormente se definen las coordenadas para definir un polígono como la región de interés (figura 3b). Finalmente se aplica una máscara a la imagen para cambiar los valores de los píxeles de la región deseada. De esta manera se cambian todos los valores de la región deseada, colocando toda la imagen en negro y la ROI en blanco (Figura 3c).



**Fig. 3.** Proceso de definición de área de interés, a) medición de imagen, b) definición de área, c) Definición de polígono.

### 3.6 Aplicación de máscara

En esta parte del proceso se aplica la máscara a la imagen donde se obtuvieron los bordes y mediante operaciones AND con números binarios se obtiene la primera representación de líneas de guía de la imagen original. Al hacer operaciones AND, solamente cuando los dos valores de los píxeles son 1 se hará un uno. La figura 8 presenta el resultado de esta etapa.

### 3.7 Aplicación de Transformada de Hough

Después de identificar los contornos en nuestra imagen y aislar la región de interés, se usó una técnica que detecta las líneas rectas en la imagen y de esta forma identificar las líneas guías en la carretera. La transformada de Hough es una técnica para la detección de figuras en imágenes digitales. Esta técnica es mayormente usada en el campo de Visión por Computadora. Con la transformada de Hough es posible encontrar todo tipo de figuras que puedan ser expresadas matemáticamente, tales como rectas, circunferencias o elipses.

El caso más simple para la transformada de Hough es la transformación lineal para detectar líneas rectas. En el espacio de la imagen, la recta se puede representar con la ecuación  $y=m*x+n$  y se puede graficar para cada par  $(x,y)$  de la imagen. En la

transformada de Hough, la idea principal es considerar las características de una recta en término de sus parámetros  $(m,n)$  y no como puntos de la imagen  $(x_1,y_1)$ ,  $(x_n,y_n)$ . Basándose en lo anterior, la recta  $y=m*x+n$  se puede representar como un punto  $(m,n)$  en el espacio de parámetros. Sin embargo, cuando se tienen rectas verticales, los parámetros de la recta  $(m,n)$  se indefinen. Por esta razón es mejor usar los parámetros que describen una recta en coordenadas polares, denotados  $(\rho, \theta)$ .

El parámetro  $\rho$  representa la distancia entre el origen de coordenadas y el punto  $(x,y)$ , mientras que  $\theta$  es el ángulo del vector director de la recta perpendicular a la recta original y que pasa por el origen de coordenadas.

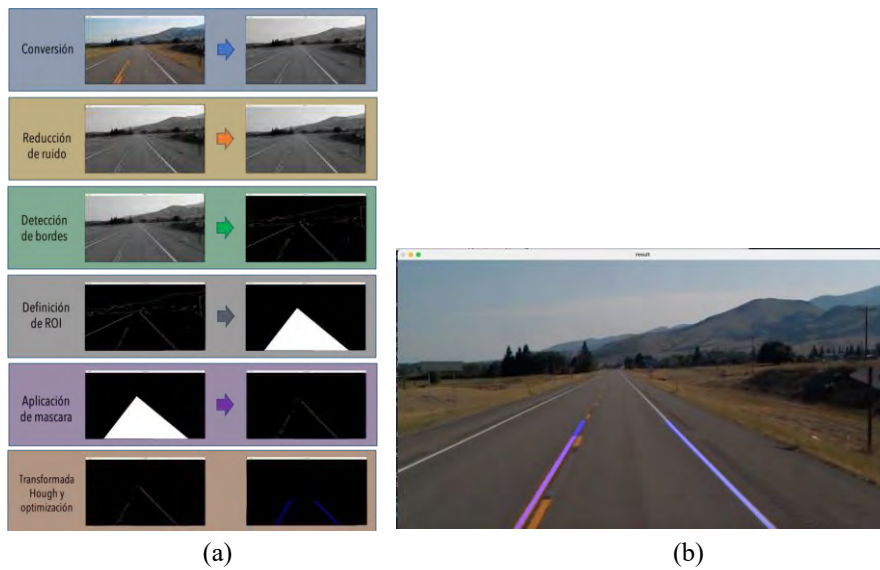
### 3.8 Optimización

Después de tener una detección de las líneas, aplicamos un proceso de optimización para que en lugar de tener múltiples líneas podamos generar una sola línea que trace la línea guía basado en el promedio de la pendiente de las líneas. Al aplicar este procedimiento se optimiza el dibujo de las líneas guía. Mediante este procedimiento, puede haber muchos puntos de intersección que representan líneas en XY. Lo que se hace es combinar todas esas líneas en dos líneas promedios maestras. Finalmente, cuando ya se tienen las dos líneas maestras, se dibujan en la imagen original de la carretera para tener un solapamiento suavizado.

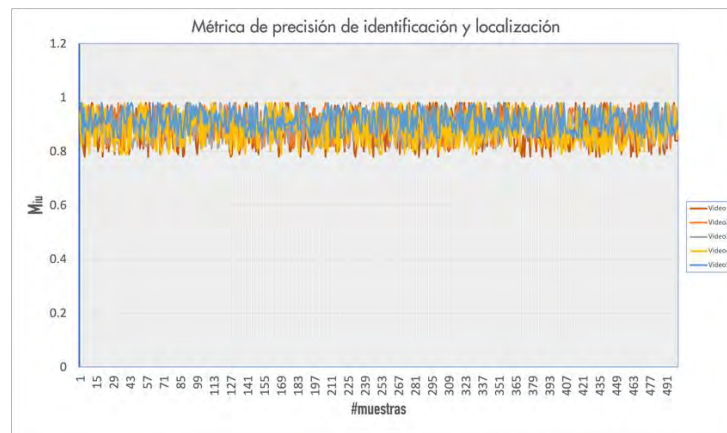
## 4. Resultados

Para la implementación del algoritmo se utilizó el lenguaje Python en conjunto con la librería de libre distribución OpenCV enfocada a la visión artificial. Para este proyecto, se utilizaron 5 diferentes videos de alrededor de 4 minutos donde se mostraba la visión de la cámara durante el proceso de conducción. Se aplicó el algoritmo y se visualizó la identificación de las líneas del carril de la carretera. La figura 4(a), muestra todas las imágenes que dan por resultado la detección de líneas del carril.

Para medir la precisión del algoritmo se realizó un comparativo de los datos reales de localización del objeto deseado con los resultados de predicción del algoritmo propuesto. Se usó la métrica de intersección sobre una unión para medir la exactitud de localización. Para esta métrica ( $M_{iu}$ ) la cual se calcula mediante una división del área de intersección entre el área de unión. De esta forma, se calcula el área de solapamiento entre la figura delimitadora real y la figura delimitadora obtenida por el algoritmo dividido entre la figura de unión de las dos regiones [12]. De acuerdo al método propuesto en [12], si el valor es mayor a 0.5 se considera un positivo verdadero. Finalmente, en la figura 5 se puede observar el resultado del proceso de la métrica  $M_{iu}$ , en la cual se observa que los valores estuvieron oscilando entre 0.78 y 0.98, con un tasas promedio entre 0.86 y 0.94. La tasa promedio de todos los cálculos fue de 0.91, lo que nos proporciona un alto grado de precisión, de acuerdo al método utilizado.



**Fig. 4.** a) diferentes resultados de las fases del proceso propuesto, b) resultado de la aplicación del algoritmo



**Fig. 5.** Resultado de la métrica de medición de exactitud y localización.

## 5. Conclusiones y trabajo futuro

Con respecto al proyecto, mediante este algoritmo propuesto pudimos realizar el primer esfuerzo para detectar carriles. Pudimos comprobar que el algoritmo fue muy preciso, ya que probamos con diferentes videos y en todos se detectó correctamente las líneas que representan la división de carriles.

Como trabajo futuro se está trabajando en aplicar este proceso para la detección de señales de tráfico y peatones para después implementarlo en un simulador que integre todas las funciones y comience a tomar decisiones en el proceso de conducción.

## References

- [1] Organización Mundial de la Salud (OMS), Informe sobre la situación mundial de la seguridad vial 2015, 2015, disponible en: [https://www.who.int/violence\\_injury\\_prevention/road\\_safety\\_status/2015/Summary\\_GSRRS\\_2015\\_SPA.pdf](https://www.who.int/violence_injury_prevention/road_safety_status/2015/Summary_GSRRS_2015_SPA.pdf)
- [2] U.S. Department of Transportation, Technical Report: Vehicle Automation and Weather: Challenges and Opportunities, 2016.
- [3] SAE, Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems, SAE On-Road Automated Vehicle Standards Committee, Hong Kong, China, Technical Report J3016\_201401, 2014.
- [4] A. Gupta, A. Anpalagan, L. Guan, A. Shaharyar Khwaja, Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues, *Array*, Volume 10, 2021.
- [5] M. Yousef, A. Hosny, W. Gamil, M. Adel, H. M. Fahmy, S. M. Darweesh, and H. Mostafa, Dual-Mode Forward Collision Avoidance Algorithm Based on Vehicle-to-Vehicle (V2V) Communication, in *IEEE International Midwest Symposium on Circuits and Systems (MWSCAS 2018)*, Windsor, Ontario, Canada, pp. 739-742, 2018.
- [6] H. Fahmy, G. Baumann, M. AbdelGhany, and H. Mostafa, "V2V-Based Vehicle Risk Assessment and Control for Lane-Keeping and Collision Avoidance", in *IEEE International Conference on Microelectronics (ICM 2017)*, Beirut, Lebanon, pp. 61-65, 2017.
- [7] Dajun Ding, Chanhoo Lee, Kwang Lee, "An Adaptive Road ROI Determination Algorithm for Lane Detection," in *Proceedings of the TENCON 2013–2013 IEEE Region 10 Conference*; pp. 22–25, 2013.
- [8] Pallavi V. Ingale and Prof. K. S. Bhagat, "Comparative Study of Lane Detection Techniques," in *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 4, no. 5, 2016.
- [9] J. Kim and C. Park, End-to-end ego lane estimation based on sequential transfer learning for self-driving cars, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 30–38.
- [10] K. Behrendt and J. Witt, "Deep learning lane marker segmentation from automatically generated labels," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 777–782.
- [11] D. Neven, B. De Brabandere, S. Georgoulis, M. Proesmans, and L. Van Gool, "Towards end-to-end lane detection: an instance segmentation approach," in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018, pp. 286–291.
- [12] Simhambhatla, Ramesh; Okiah, Kevin; Kuchkula, Shравan; and Slater, Robert (2019) "Self-Driving Cars: Evaluation of Deep Learning Techniques for Object Detection in Different Driving Conditions," *SMU Data Science Review: Vol. 2 : No. 1 , Article 23*.