



LA FORMACIÓN DE INGENIEROS:
UN COMPROMISO PARA EL
DESARROLLO Y LA SOSTENIBILIDAD

15 al 18
DE SEPTIEMBRE

20
20

www.acofi.edu.co/eiei2020

DETECCIÓN AUTOMÁTICA DE GESTOS DE CONDUCTORES COMO PARTE DE UN SISTEMA AVANZADO DE ASISTENCIA AL CONDUCTOR ADAS

Javier Chaparro, Nicolás Mosquera

**Escuela Colombiana de Ingeniería Julio Garavito
Bogotá Colombia**

Resumen

Los carros autónomos son una apuesta para el futuro del transporte en las grandes ciudades y se han convertido en un reto en varios centros de investigación del mundo. La Escuela Colombiana de Ingeniería Julio Garavito, y específicamente la Decanatura de Ingeniería Electrónica, viene apoyando la apropiación de este tipo de tecnologías en asocio con una empresa de tecnología colombiana. Por ese motivo, desarrolla actualmente el proyecto de investigación titulado "Sistema de asistencia a conductores basado en monitoreo de parámetros internos y externos de un automóvil". La problemática que se identificó está relacionada con la adaptación de los vehículos autónomos en nuestro país, ya que si bien los desarrollos de Sistemas Avanzados de Asistencia al Conductor (ADAS por sus siglas en inglés de *Advanced Driver Assistance Systems*), han alcanzado un importante grado de madurez en diferentes centros de investigación, en nuestro país es necesario hacer adaptaciones de estos dadas las particularidades en las vías y el entorno. Desde hace más de un año se viene trabajando para desarrollar un sistema electrónico de bajo costo que implemente un algoritmo para detectar automáticamente fatiga en los conductores identificando de variables fisiológicas (frecuencia cardíaca) e intrínsecas del comportamiento del conductor (número de parpadeos, estado de la boca y postura de la cabeza). Como no es posible tener conectado al conductor a un equipo para medir parámetros cardíacos, se utilizó una técnica denominada pletismografía por imágenes (PPGI) que emplea una cámara Intel RealSense sr300. En un primer trabajo de pregrado un estudiante diseñó un sistema de alarmas de fatiga analizando las variables anteriores y lo implementó en el entorno de trabajo ROS (*Robot Operating System*). Posteriormente, un estudiante de maestría evaluó el desempeño del algoritmo en diferentes tarjetas de placa reducida SBC (*Single Board Computer*) con el objetivo de determinar la que ofreciera la mejor relación *desempeño/costo*. Hasta el momento las boards analizadas han sido la Atomic Pi,

LatePanda, y Raspberry Pi 4. En este trabajo se presenta el algoritmo que se diseñó para establecer las condiciones internas del conductor y los resultados de la evaluación del desempeño de la boards.

Palabras clave: ADAS; accidentes de tránsito; parámetros internos

Abstract

The technology of the autonomous cars is pointing to the improvement of the future transportation in main cities, and it is becoming a challenge in different research centers around the world. The "Escuela Colombiana de Ingeniería Julio Garavito", more specifically the Dean of Electronic Engineering, has been supporting the appropriation of this type of technology in association with a Colombian technology company. For this reason, we are currently developing a research project entitled "Driver assistance system based on monitoring internal and external parameters of a car". The problem that was identified is related to the adaptation of autonomous vehicles in Colombia, since although the development of Advanced Driver Assistance Systems (ADAS) has reached an important level of development in different research centers, in our country it's necessary to make adjustments because of the singularities of the roads and the environment. More than a year ago, we started working in order to develop a low-cost electronic system that uses an algorithm to detect automatically driver fatigue, identifying physiological (heart rate) and intrinsic variables of the driver's behavior (number of blinks, gestures of the mouth and the head position). Since there is no way to measure the cardiac parameters of a person directly from a computer inside the car, a technique called image plethysmography (PPGI) was used with an Intel RealSense sr300 camera. In an undergraduate student project, he designed a fatigue alarm system, analyzing the previous variables and later it was implemented in ROS (Robot Operating System) environment. Subsequently, a postgraduation student evaluated the performance of the algorithm on different SBC (Single Board Computer) with the aim of determining the one that offered the best performance/cost ratio. So far, the boards analyzed have been: Atomic Pi, LatePanda, and Raspberry Pi 4. In this work, the algorithm that was designed to establish the internal conditions of the driver and the results of the performance in the board evaluation are presented.

Keywords: ADAS; road accidents; internal parameters

1. Introducción

La integración de los sistemas inteligentes en los diferentes escenarios de nuestra vida es una carrera que lideran diferentes empresas tecnológicas a nivel mundial, y busca extender la automatización más allá de la simple reproducción de tareas repetitivas. Las empresas fueron las primeras en aprovechar la automatización para dejar a cargo de máquinas las labores que no implicaban el uso de la inteligencia, y que, por el contrario, eran muy operativas. Con el advenimiento de nuevas tecnologías de hardware y software, relacionadas con la inteligencia artificial principalmente, se ha facilitado la automatización de procesos que involucran la toma de decisiones.

La automatización llegó entonces al transporte de personas en las ciudades. Pero antes de pasar a disponer de automóviles completamente autónomos es necesario desarrollar tecnologías que apoyen la conducción controlada por seres humanos. De esta forma, los sistemas ADAS (*Advanced Driver Assistance Systems*, por sus siglas en inglés) son una alternativa para aumentar la seguridad y es uno de los primeros pasos para llegar a la automatización completa de este proceso. Se pueden distinguir en ellos dos componentes fundamentales, el primero lo integra la información externa al vehículo y el segundo la información relativa al interior de este; generalmente centrada en las condiciones del conductor. Esta clasificación se conoce en la literatura de ADAS como monitoreo externo e interno (Felipe Jiménez, *Advanced Driver Assistance System for road environments to improve safety and efficiency*, 2016).

Teniendo en cuenta esta perspectiva, la Escuela de Ingeniería Julio Garavito y la empresa Millenium BPO desarrollaron una investigación, que tenía como objetivo principal diseñar un sistema de alarmas tanto internas como externas al vehículo, a partir del procesamiento de video en tiempo real, que asista continuamente al conductor. En este artículo se presenta la evaluación de desempeño del algoritmo de alarmas internas, basadas en la identificación del estado de atención del conductor, con diferentes tarjetas electrónicas.

En el diseño se identificaron como prioritarias las variables frecuencia cardiaca, número de parpadeos, estado de la boca y postura de la cabeza. Para medir la primera variable se utilizó una técnica denominada pletismografía por imágenes (PPGI) con el uso de una cámara Intel RealSense sr300. En el primer trabajo, con un estudiante de pregrado, diseñó el primer sistema de alarmas de fatiga analizando las variables anteriores y lo implementó en el entorno de trabajo ROS (*Robot Operating System*). Posteriormente, un estudiante de maestría evaluó el desempeño de este en diferentes boards de placa reducida SBC (*Single Board Computer*). El objetivo de este último trabajo fue identificar la mejor relación desempeño/costo en entre las boards Atomic Pi, LatePanda, y Raspberry Pi 4.

En este documento se presenta la problemática relacionada con la conducción de vehículos, se hace un resumen del estado del arte de los sistemas avanzados para la asistencia de conductores, se describe el algoritmo de detección de eventos internos, y finalmente se hace la evaluación del desempeño de diferentes SBC.

2. Problemática asociada con la conducción de vehículos

La Organización Mundial de la Salud reporta que el número de accidentes de tránsito anuales está cerca de 1.3 millones, con más de 3.000 defunciones diarias, con el atenuante que más de la mitad de estas personas no viajaban en automóvil (peatones, ciclistas y motociclistas), y el 90% de los casos eran causados por un error humano (OMS, 2020). Este mismo documento establece que entre 20 y 50 millones de personas más sufren traumatismos no mortales provocados por accidentes de tránsito; tales traumatismos constituyen una causa importante de discapacidad en todo el mundo. Como hecho importante se destaca que el 90% de las defunciones por accidentes de tránsito tienen lugar en los países de ingresos bajos y medianos, y curiosamente en estos se halla menos de la

mitad de los vehículos matriculados en todo el mundo. Finalmente se destaca como las tres causas principales de defunciones, en personas entre 5 a 44 años, los traumatismos causados por el tránsito automotor.

En un estudio realizado por el *National Highway Traffic Safety Administration*, se observó que la fatiga del conductor y la somnolencia o distracción son las causas del 80% de los accidentes automovilísticos (Singh, 2015). Los sistemas de asistencia al conductor ayudaran seguramente a reducir esta dramática estadística, ya que con usando una cámara apuntada hacia él, pueden establecer si el conductor accede a un teléfono móvil, o no mira la carretera durante un tiempo específico, y esto llevará a que se emita una alerta (Safety, 2019). Adicionalmente, los sistemas de detección de somnolencia y fatiga incluyen la capacidad de detectar si el conductor se ha quedado dormido y pueden alertarlo a través de una secuencia de vibraciones del cinturón de seguridad y alertas de los altavoces (Garcés, 2015).

3. Sistemas Avanzados para Asistencia de Conductores (ADAS)

La movilidad autónoma de vehículos en las calles es un tema importante para los fabricantes y para los conductores que está siendo atendido actualmente por varios centros de investigación y empresas alrededor del mundo. Actualmente, empresas líderes en el desarrollo de tecnología ya ofrecen comercialmente dispositivos específicos que se integran para conformar dispositivos ADAS. A pesar de estos esfuerzos, los accidentes relacionados con automóviles se convierten en un reto difícil de superar ya que la identificación de alarmas en un entorno incierto y variable representa una alta exigencia para cualquier sistema automático.

El sistema ADAS brinda asistencia al conductor y mejora la experiencia de manejo. Su función principal es garantizar la seguridad del vehículo, el conductor y los peatones o ciclistas. Este podría usarse para ahorrar costos de combustible al permitir un pelotón en el que los vehículos se siguen entre sí a corta distancia; puede advertir cuando un vehículo se desvía por el carril o puede aplicar un freno de emergencia para evitar colisiones, etc. Para funcionar de manera confiable, debe ser capaz de reconocer objetos, letreros, superficie de la carretera y objetos en movimiento en la carretera, así como tomar decisiones o actuar en nombre de un conductor (Zhao, 2015).

Desde el punto de vista físico, para identificar el entorno los ADAS emiten ondas que rebotan, básicamente son radares que mapean el entorno. La información obtenida se complementa gracias a la utilización de otros componentes tecnológicos como cámaras monoculares, infrarrojas y 3D, además de sensores laser tipo Lidar, entre otros. Adicionalmente, el procesamiento de la información de los sensores, incluyendo las cámaras, es fundamental para identificar las posibles alarmas que serán presentadas al conductor. Por este motivo el campo del procesamiento de imágenes y la inteligencia artificial son esenciales en este tipo de sistemas (Seunghyuk Choi, 2016).

4. Algoritmo de generación de alarmas

Los parámetros intrínsecos del conductor registrados y analizados en el proyecto fueron: estado de los ojos, estado de la boca, postura de la cabeza, y frecuencia cardiaca por métodos no invasivos. Los métodos desarrollados a partir del procesamiento de marcadores de imágenes hechos sobre el rostro de las personas son explicados por varios autores como (Yi-Ming Chan, 2013) y (Yuanyuan Nan, 2014).

El estado de la boca(bostezo) y la postura de la cabeza (ángulos de inclinación pitch yaw y roll) son características que dan información sobre el nivel de fatiga o desatención del conductor. Se definen tres estados de esta primera característica: boca cerrada, medianamente abierta, y totalmente abierta o bostezo. Para la detección de características del rostro se estimaron los llamados Facial Landmarks para lo cual existen varios algoritmos, en este caso se utilizó el desarrollado por (Vahid Kazemi, 2014). El bostezo, el parpadeo y postura de la cabeza se realizó utilizando un grupo específico de dichas marcas.

Finalmente se adquirió la frecuencia cardiaca del conductor mediante una técnica no invasiva conocida como pletismografía por imágenes (PPGI) que utiliza la región de interés de la nariz y frente. Mediante el promedio de todos los puntos de las regiones se obtuvo un punto y posteriormente con la aplicación de análisis de componentes individuales (ICA) se establecieron señales, filtradas entre 0.75 y 4 Hz, cuya mayor componente frecuencia arroja la frecuencia cardiaca. El algoritmo completo se muestra en la Figura 1. La cámara utilizada en este proyecto fue una Realsense sr300 con la cual se adquirieron imágenes a una resolución de 640x480 pixeles. El algoritmo completo se presenta en el trabajo fin de carrera de (Karlo, 2019) quién lo desarrolló con la tutoría de profesores del programa de Ingeniería Electrónica y la empresa Millenium BPO.

5. Evaluación del desempeño del algoritmo

El algoritmo se implementó en varios sistemas embebidos con el propósito de identificar aquel que tenga el mejor desempeño y el menor costo. Para lograr este objetivo se podía utilizar sistemas operativos en chip (SoCs), circuitos integrados para aplicaciones específicas (ASICs), y arreglos de compuertas programables en campo (FPGAs) entre otros. En este trabajo se presentan los resultados empleando los SoCs que es un hardware donde se integran todos los componentes de un computador, e incluyen además otros componentes y conectores que facilitan su utilización gracias a la facilidad de conectar diferentes dispositivos de entrada/salidas comerciales.

Las boards de placa reducida fueron dispuestas con un sistema operativo Linux (Ubuntu LTS) sobre el cual se instaló ROS (Robot Operating System) que es un framework de desarrollo de software para robots. Sobre este se utilizó la herramienta Rosbag que permite grabar y reproducir mensajes publicados por los nodos de ROS, adicionalmente, es útil para depurar nuevos algoritmos, grabar datos para simular robots, y realizar la captura de información de varios dispositivos para realizar pruebas estandarizadas.

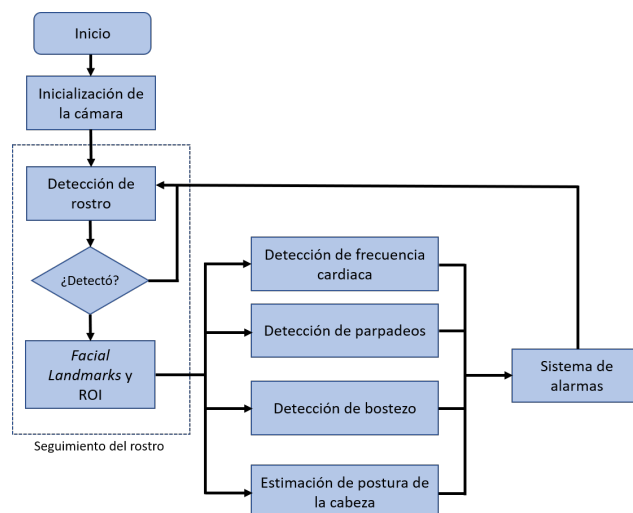


Figura 1. Diagrama de flujo del sistema de monitoreo del conductor

Los procesadores proporcionan contadores de rendimiento con los cuales se realiza la medición de eventos claves asociados con la microarquitectura como: ciclos de reloj transcurridos, número de instrucciones retiradas, predicciones erróneas de saltos, desaciertos en la memoria cache y búfer de traducción anticipada (TLB), entre otros. La caracterización de la carga de trabajo del algoritmo fue hecha mediante el análisis de estos eventos. Los contadores de dichos eventos se encuentran en el chip en la unidad de desempeño o Performance Monitoring Unit (PMU) y son integrados en los núcleos del procesador, controladores de cache de último nivel, controladores de memoria, tarjetas gráficas y dispositivos de E/S (Dimakopoulou, 2016).

Otro parámetro para establecer el rendimiento de una aplicación es analizar el uso de memoria. Con el comando "top" de Linux se puede ver tanto en desempeño del procesador como el uso de memoria. En la Figura 2 se puede ver los resultados de este comando donde se puede identificar la carga promedio y el uso de la memoria RAM y de intercambio. Estos dos parámetros fueron utilizados para establecer diferencias en el desempeño de las SBC al ejecutar el algoritmo de generación de alarmas programado en ROS.

```

top - 09:34:14 up 36 min, 3 users, load average: 0.31, 0.55, 0.42
Tasks: 140 total, 1 running, 139 sleeping, 0 stopped, 0 zombie
Cpu0  : 0.3%us, 0.8%sy, 0.0%ni, 92.8%id, 2.7%wa, 0.0%hi, 3.5%si, 0.0%st
Cpu1  : 0.2%us, 0.7%sy, 0.0%ni, 97.3%id, 1.8%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 4083276k total, 3937288k used, 145988k free, 672k buffers
Swap: 2097144k total, 156k used, 2096988k free, 3822700k cached

  PID USER   PR   NI  VIRT  RES  SHR S %CPU  %MEM    TIME+  COMMAND
    1 root     20    0  1804   760  548 S    0   0.0   0:01.19 init
    2 root     15   -5     0     0     0 S    0   0.0   0:00.00 kthreadd
  
```

Figura 2. Resultados del comando Top. Se identifican la carga promedio y la memoria utilizada por las aplicaciones.

6. Resultados

Para la prueba en las tarjetas seleccionadas se grabó un video con un voluntario simulando la acción de conducir que incluía varios eventos de relacionados con la fatiga y cansancio. Este mismo video fue reproducido desde una memoria Usb en cada una de estas tarjetas durante la evaluación del algoritmo. En la Tabla 1 se puede ver el resumen del desempeño de las tarjetas con respecto a los parámetros de evaluación establecidos.

Tabla 1. Resultados del desempeño de SBC al ejecutar el algoritmo de generación de alarmas en ROS

SBCs	Carga promedio	Promedio procesos IDLE	Uso de Memoria RAM/Intercambio
Atomic Pi	1.31 (32.8 %)	66.2%	451.3/34.1 MB
LattePanda	1.36 (34.0 %)	65.1%	529.3/2.9 MB
Raspberry Pi 4 Model B	3.70 (92.5 %)	6.1%	424.1/14.9 MB

Los resultados presentados en la tabla anterior muestran que el SBC LattePanda, logró cumplir los criterios de evaluación, ya que cuenta con un promedio de carga computacional del 34% y las CPUs estando un 65.1% del tiempo en modo espera y uso de 135 MB mientras se da la ejecución del algoritmo.

La Atomic Pi, también cumple con los criterios de evaluación. Con un promedio de carga computacional del 32.8% y las CPUs estando un 66,2% del tiempo en modo espera y uso de memoria RAM de aproximadamente 105 MB durante la ejecución del algoritmo. Este SBC presenta el mejor desempeño con respecto a los SBCs LattePanda y Raspberry Pi 4 Model B, siendo este el de menor costo comercial y menor capacidad de memoria RAM. Por último, es el SBC con mayores dimensiones volumétricas.

El SBC Raspberry Pi 4 Model B, no cumple los criterios de evaluación. Con un promedio de carga computacional del 92.8% y las CPUs estando un 5.98% del tiempo en modo espera y un uso de memoria RAM de aproximadamente 110 MB en la ejecución del algoritmo. Es posible que el resultado esté relacionado por el uso de una memoria Usb para leer las imágenes generadas por la cámara.

7. Conclusiones

La seguridad en la conducción es un tema importante dadas las altas tasas de accidentalidad que se presentan en el mundo. En particular, la relacionada con las condiciones del conductor representa una alta incidencia en las estadísticas y debe seguir siendo un tema de investigación; a pesar de que se esté pensando en estos momentos en automóviles completamente autónomos.

El algoritmo de detección de alarmas debidos a riesgos generados en el conductor, puede ser objeto de mejora ya que variables como la iluminación del entorno y el uso de gafas, pueden reducir significativamente la desempeño de este. Adicionalmente, se dispone actualmente de

nuevas técnicas de procesamiento de imágenes en tiempo real, e inteligencia artificial, que pueden ser incorporadas para mejorarlo.

La evaluación del desempeño del algoritmo de detección de alarmas probado en las tarjetas LattePanda, Atomic Pi, y Pi 4 Model B, mostró que la primera ofrece el mejor desempeño estimándolo a partir del uso del procesador y la memoria durante la ejecución. Sin embargo, hay que tener en cuenta que la información de las imágenes no fue tomada directamente de la cámara sino de una memoria Usb; esto puede producir diferencias en desempeño que pueden ser verificadas.

El trabajo interinstitucional entre la universidad y la empresa es provechoso en tanto se tengan claras las necesidades y capacidades de las dos partes. En este caso, el tema de sistemas de asistencia al conductor resultó de interés tanto para la empresa como para la universidad, pero puede mejorar el trabajo interinstitucional si se tienen problemáticas de mayor alcance donde se puedan involucran otras unidades académicas.

8. Referencias

- Dimakopoulou, M. E. (2016). Reliable and Efficient Performance Monitoring in Linux. *International Conference for High Performance Computing, Networking, Storage and Analysis*.
- Felipe Jiménez, e. a. (2016). Advanced Driver Assistance System for road environments to improve safety and efficiency. *ScienceDirect*, 14, 2245-2254.
- Felipe Jiménez, e. a. (2016). *Advanced Driver Assistance System for road environments to improve safety and efficiency*. Obtenido de Advanced Driver Assistance System for road environments to improve safety and efficiency: <https://www.sciencedirect.com/science/article/pii/S2352146516302460>
- Garcés, M. A. (2015). *Sistemas de detección de somnolencia en conductores*. Obtenido de https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwjqsY_g_onqAhUvmeAKHRf7BysQFjAAegQIARAB&url=https%3A%2F%2Fdiagonalnet.unirioja.es%2Fdescarga%2Farticulo%2F5432283.pdf&usg=AOvVaw2DZNRjvzU02DWavF6JlcY
- Karlo, F. Y. (2019). *Diseño e Implementación del Prototipo de un Sistema de Asistencia al Conductor Basado en el Monitoreo de Parámetros de un Conductor al Interior de un Vehículo*. Escuela Colombiana de Ingeniería Julio Garavito, Bogotá.
- Kukkala, V. K., & Jordan Tunnell. (2018). Advanced Driver-Assistance Systems: A Path Toward Autonomous. *IEEE Consumer Electronics Magazine*.
- OMS. (Mayo de 2020). https://www.who.int/roadsafety/decade_of_action/plan/plan_spanish.pdf?ua=1.
- Safety, I. f. (2019). <https://injuryfacts.nsc.org/motor-vehicle/occupant-protection/advanced-driver-assistance-systems/>.
- Seunghyuk Choi, F. T. (2016). *Advanced driver-assistance systems: Challenges and opportunities ahead*.

- Singh. (2015). *Critical Reasons for Crashes Investigated in the National Motor Vehicle Crash Causation Survey*. Obtenido de <https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812115>
- Vahid Kazemi, J. S. (2014). One Millisecond Face Alignment with an Ensemble of Regression Trees. *Conference: Computer Vision and Pattern Recognition, At Columbus, Ohio, USA*.
- Yi-Ming Chan, L.-C. F.-Y.-F. (2013). Pedestrian detection using histograms of Oriented Gradients of granule feature. *Conference: Intelligent Vehicles Symposium (IV)*.
- Yuanyuan Nan, C. L. (2014). Histograms of Saliency for Pedestrian Detection. *Proceedings of International Conference on Internet Multimedia Computing and Service*.
- Zhao, M. (2015). *Advanced Driver Assistant System Threats Requirements, Security Solutions*.

Sobre los Autores

- **Javier Chaparro.** Ingeniero Electrónico, Especialista en Automatización Industrial, Magister en Telecomunicaciones, Doctor en Ingeniería Biomédica. Profesor asociado de la Escuela Colombiana de Ingeniería Julio Garavito, Bogotá. Email: javier.chaparro@escuelaing.edu.co.
- **Nicolás Mosquera Seligmann.** Estudiante de la Maestría de Ingeniería Electrónica de la Escuela Colombiana de Ingeniería Julio Garavito. Ingeniero Electrónico. Email: nicolas.mosquera-s@mail.escuelaing.edu.

Los puntos de vista expresados en este artículo no reflejan necesariamente la opinión de la Asociación Colombiana de Facultades de Ingeniería.

Copyright © 2020 Asociación Colombiana de Facultades de Ingeniería (ACOFI)