



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE ELÉCTRICA Y ELÉCTRONICA

**CARRERA DE INGENIERÍA EN ELECTRÓNICA E
INSTRUMENTACIÓN**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL
TÍTULO DE INGENIERO ELECTRÓNICO EN
INSTRUMENTACIÓN**

**TEMA: “SISTEMA DE VIGILANCIA AL CONDUCTOR
VEHICULAR BASADO EN TÉCNICAS DE VISIÓN ARTIFICIAL
E IMPLEMENTADO EN UN SMARTPHONE PARA LA
DETECCIÓN Y ALERTA DE SOMNOLENCIA”**

AUTOR: FABRICIO DANIEL EGAS CUNALATA

DIRECTOR: ING. EDDIE GALARZA

LATACUNGA

2017



DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN ELECTRÓNICA E INSTRUMENTACIÓN

CERTIFICACIÓN

Certifico que el trabajo de titulación, “**SISTEMA DE VIGILANCIA AL CONDUCTOR VEHICULAR BASADO EN TÉCNICAS DE VISIÓN ARTIFICIAL E IMPLEMENTADO EN UN SMARTPHONE PARA LA DETECCIÓN Y ALERTA DE SOMNOLENCIA**” realizado por **Fabricio Daniel Egas Cunalata**, ha sido revisado en su totalidad y analizado por el software anti-plagio, el mismo cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, por lo tanto me permito acreditarlo y autorizar a **Fabricio Daniel Egas Cunalata** para que lo sustente públicamente.

Latacunga, agosto de 2017



Ing. Eddie Galarza
DIRECTOR



**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN ELECTRÓNICA E INSTRUMENTACIÓN**

AUTORÍA DE RESPONSABILIDAD

Yo, **Fabricio Daniel Egas Cunalata**, con cédula de identidad N°1804463402, declaro que este trabajo de titulación **“SISTEMA DE VIGILANCIA AL CONDUCTOR VEHICULAR BASADO EN TÉCNICAS DE VISIÓN ARTIFICIAL E IMPLEMENTADO EN UN SMARTPHONE PARA LA DETECCIÓN Y ALERTA DE SOMNOLENCIA”** ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaro que este trabajo es de mi autoría, en virtud de ello me declaro responsable del contenido, veracidad y alcance de la investigación mencionada.

Latacunga, agosto de 2017

Fabricio Daniel Egas Cunalata

C.C.:1804463402



**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN ELECTRÓNICA E INSTRUMENTACIÓN**

AUTORIZACIÓN

Yo, **Fabricio Daniel Egas Cunalata**, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar en la biblioteca Virtual de la institución el presente trabajo de titulación “**SISTEMA DE VIGILANCIA AL CONDUCTOR VEHICULAR BASADO EN TÉCNICAS DE VISIÓN ARTIFICIAL E IMPLEMENTADO EN UN SMARTPHONE PARA LA DETECCIÓN Y ALERTA DE SOMNOLENCIA.**” cuyo contenido, ideas y criterios es de mi autoría y responsabilidad.

Latacunga, agosto de 2017

A handwritten signature in blue ink, which appears to read 'Daniel Egas Cunalata', is written over a horizontal line.

Fabricio Daniel Egas Cunalata

C.C.:1804463402

DEDICATORIA

Este trabajo es dedicado a Estefanía quien me motivo a asistir a clases durante la mayor parte de mi vida universitaria y me enseñó dos principios valiosos: el primero que lo esencial de la vida no se mira con los ojos y el segundo que la felicidad es el camino y no la meta.

Daniel

AGRADECIMIENTO

El presente trabajo de titulación primeramente me gustaría agradecer a Dios por darme las circunstancias necesarias para haber culminado con éxito mi vida académica.

A toda mi familia en especial a mis madres Olga y Lila y a mi padre Carlos por todo el apoyo, paciencia y ánimo que me han brindado para llegar hasta donde he llegado, porque sin ustedes no sería lo mismo.

A la UNIVERSIDAD DE LAS FUERZAS ARMADAS ESPE-L por darme la oportunidad de estudiar y ser un profesional.

A mi tutor, Ing. Eddy Galarza por su calidad tanto como docente como persona, quien con sus conocimientos, su experiencia y sus consejos me han ayudado a ser mejor.

También me gustaría agradecer a mis profesores durante toda mi carrera profesional porque todos han aportado con un granito de arena a mi formación, y en especial los Ings. Fabricio Pérez y David Rivas por sus consejos, su enseñanza y más que todo por su amistad.

Son muchas las personas que han formado parte de mi vida profesional a las que me encantaría agradecerles su amistad, consejos, apoyo, ánimo y compañía en los momentos más difíciles de mi vida. Algunas están aquí conmigo y otras en mis recuerdos y en mi corazón, sin importar en donde estén quiero darles las gracias por ser parte de mí, por el tiempo compartido y por todas sus bendiciones. Para ellos: mi eterna gratitud.

Daniel

ÍNDICE DE CONTENIDOS

TRABAJO	i
CERTIFICACIÓN	ii
AUTORÍA DE RESPONSABILIDAD	iii
AUTORIZACIÓN	iv
DEDICATORIA	v
AGRADECIMIENTO	vi
ÍNDICE DE CONTENIDOS	vii
ÍNDICE DE FIGURAS	xii
RESUMEN	xiv
ABSTRACT	xv

CAPÍTULO I

INTRODUCCIÓN	1
1.1. Planteamiento del Problema	1
1.2. Antecedentes	1
1.3. Justificación e Importancia	2
1.4. Objetivos	3
1.4.1. Objetivo general	3
1.4.2. Objetivos específicos.....	3

CAPÍTULO II

MARCO TEÓRICO	4
2.1. Antecedentes investigativos	4
2.2. Fundamentación teórica	5
2.2.1. Somnolencia.....	5
a) Definición.....	5
b) Efectos	5
c) Signos fisiológicos	6
d) Relación de la somnolencia con los accidentes de tránsito.....	6
e) Detección de somnolencia	6

2.2.2. Smartphone.....	9
2.2.3. Sistemas operativos móviles.....	10
a) Estructura de un SO móvil.....	10
b) Sistemas operativos móviles más usados.....	12
2.2.4. Android.....	12
a) Características:	12
b) Arquitectura.....	13
c) Entorno de desarrollo	15
d) Documentación	15
2.2.5. IOS	16
a) Características	16
b) Arquitectura.....	16
c) Entorno de desarrollo	17
d) Documentación	17
2.2.6. Aplicaciones Android.....	17
a) Ciclo de vida de una actividad.....	18
b) Métodos.....	19
2.2.7. Visión artificial	19
a) Etapas de un sistema de visión artificial.....	20
b) Herramientas de un sistema de visión artificial en Android	21
2.3. Variables de la investigación del proyecto en ejecución	22
2.4. Hipótesis.....	23

CAPÍTULO III

ANÁLISIS, DISEÑO E IMPLEMENTACIÓN	24
3.1.1. Descripción del sistema	24
3.1.2. Requerimientos de la aplicación	25
3.1.3. Recursos.....	28
3.1.4. Dispositivos seleccionados para las Pruebas	29
3.2.1. Diagrama de flujo principal.....	30
3.2.2. Diagrama de casos de uso.....	32
3.2.3. Diseño del algoritmo para la detección de factores visuales que indican somnolencia.....	35

3.2.4.	Movimientos de la cabeza.....	35
a)	Distracción	35
b)	Asentimiento de la cabeza	37
c)	Expresión facial.....	41
d)	Análisis ocular	43
e)	Niveles de somnolencia de acuerdo a indicadores visuales	46
3.3.2.	Configuración de la Librería Google Mobile Api Visión	46
3.3.3.	Creación de la Interface de usuario UI.....	48
3.3.4.	Adquisición de la imagen	51
3.3.5.	Detección del rostro	52
3.3.6.	Detección del estado de los ojos.....	52
3.3.7.	Detección de distracción	53
3.3.8.	Detección de bostezo.....	55
3.3.9.	Detección del asentimiento frontal de la cabeza	56

CAPÍTULO IV

	PRUEBAS Y RESULTADOS EXPERIMENTALES	57
4.1.	Metodología para las pruebas.....	57
4.1.1.	Ubicación correcta del dispositivo.....	58
4.2.	Pruebas de funcionamiento del sistema.....	59
4.2.1.	Pruebas de funcionamiento de bostezos.	59
4.2.2.	Pruebas de funcionamiento de asentimiento frontal	60
4.2.3.	Pruebas de funcionamiento de asentimiento derecho	61
4.2.4.	Pruebas de funcionamiento de asentimiento izquierdo.....	62
4.2.5.	Pruebas de funcionamiento distracción derecha.....	64
4.2.6.	Pruebas de funcionamiento de distracción izquierda.	65
4.2.7	Puebas de funcionamiento de la detección de parpadeo.....	67
4.3	Pueba de la secuencia de activación de los niveles de alarma	68
4.4.	Pruebas de usabilidad.....	76
4.5.	Verificación de hipótesis.....	79

CAPÍTULO 5**CONCLUSIONES, RECOMENDACIONES Y TRABAJOS FUTUROS**

.....	80
5.8. Conclusiones.....	80
5.9. Recomendaciones.....	81
5.10. Trabajos futuros	82

REFERENCIAS BIBLIOGRÁFICAS 83**ANEXOS..... 88**

ÍNDICE DE TABLAS

Tabla 1 Requerimientos funcionales de la aplicación	25
Tabla 2 Requerimientos no funcionales de la aplicación	26
Tabla 3 Recursos de hardware y software usados por el sistema	28
Tabla 4 Características técnicas del smartphone Huawei P8 Lite	29
Tabla 5 Subprocesos del diagrama de flujo principal.....	32
Tabla 6 Subprocesos del diagrama de caso de uso principal	34
Tabla 7 Niveles de somnolencia de acuerdo a indicadores visuales	46
Tabla 8 Porcentaje de detección de bostezo	59
Tabla 9 Porcentaje de detección de cabeceo	60
Tabla 10 Porcentaje de detección de distracción izquierda	61
Tabla 11 Porcentaje de detección de distracción derecha.....	63
Tabla 12 Porcentaje de detección de inclinacion izquierda	64
Tabla 13 Porcentaje de detección de inclinacion derecha	65
Tabla 14 Porcentaje de detección de parpadeo.....	66
Tabla 15 Porcentaje de detección total del sistema.....	67
Tabla 16 Resultados de las pruebas para los conductores con gorra.	73
Tabla 17 Resultados de las pruebas de los conductores con lentes	74
Tabla 18 Contraste de las pruebas de una conductora con cabello	76

ÍNDICE DE FIGURAS

Figura 1 Métodos para detectar somnolencia	7
Figura 2 Arquitectura de Android	13
Figura 3 Arquitectura de IOS	16
Figura 4: Etapas de un sistema de visión artificial	20
Figura 5 Esquema del sistema de vigilancia al conductor	24
Figura 6 Diagrama de flujo principal	31
Figura 7 Figura Diagrama de caso de uso principal.....	33
Figura 8 Dirección del rostro a) mirada hacia izquierda b) mirada hacia la derecha	35
Figura 9 Diagrama de flujo del algoritmo para la detección de distracción ..	36
Figura 10 Asentimiento de la cabeza de manera frontal.....	37
Figura 11 Diagrama de flujo de asentimiento frontal de la cabeza	38
Figura 12 Asentimiento de la cabeza de manera lateral	39
Figura 13 Diagrama de flujo de asentimiento lateral de la cabeza.....	40
Figura 14 Apertura de la boca en a) estado normal b) estado de bostezo ..	41
Figura 15 Diagrama de flujo de bostezo	42
Figura 16 Porcentaje del cierre del ojo PERCLOS a) ojo abierto b) ojo cerrado al 70% c) ojo cerrado al 80% d) ojo cerrado	43
Figura 17 Diagrama del algoritmo para la detección del estado de los ojos	45
Figura 18 Ubicación del archivo build.gradle	47
Figura 19 Ubicación del archivo AndroidManifest.xml	48
Figura 20 Pantalla de inicio.....	49
Figura 21 Pantalla de detección de indicadores visuales	50
Figura 22 Pantalla de alerta para alarma de conductor dormido	51
Figura 23 Imagen adquirida por la cámara frontal del smartphone.....	52
Figura 24 Detección del rostro	52
Figura 25 Detección de los ojos abiertos	53
Figura 26 Detección de los ojos cerrados	53
Figura 27 Detección del rostro del conductor mirando hacia la derecha	54
Figura 28 Detección del rostro del conductor mirando hacia la izquierda....	54

Figura 29 Detección de la boca cuando se encuentra normal	55
Figura 30 Detección de la boca cuando se encuentra bostezando.....	55
Figura 31 Detección del asentimiento frontal de la cabeza.....	56
Figura 32 Detección del rostro cuando se encuentra asintiendo de manera frontal	56
Figura 33 Posición del smartphone en el vehículo.....	58
Figura 34 Secuencia de activación del nivel de alarma bajo	68
Figura 35 Secuencia activación del nivel de alarma medio	69
Figura 36 Secuencia de activación del nivel de alarma alto	70
Figura 37 Flexibilidad al inclinar la cabeza en conductores de:	72
Figura 38 Conductores usando gorra	72
Figura 39 Conductor con lentes a) sin anti reflejante b) c) d) e) con anti reflejante	74
Figura 40 Conductora con cabello que le cubre las mejillas: a) cara no detectada b) detección errónea de los puntos de referencia	75
Figura 41 Resultado de la tabulación de la pregunta 1	76
Figura 42 Resultado de la tabulación de la pregunta 2.....	77
Figura 43 Resultado de la tabulación de la pregunta 3.....	77
Figura 44 Resultado de la tabulación de la pregunta 4.....	78
Figura 45 Resultado de la tabulación de la pregunta 5.....	78

RESUMEN

En el presente trabajo de titulación se desarrolla un sistema de vigilancia al conductor vehicular para detectar y alertar sobre la presencia de somnolencia. El sistema es implementado en un smartphone a manera de aplicación móvil bajo el sistema operativo Android. Para su desarrollo se utilizó el lenguaje de programación Java y la librería de visión artificial Google Mobile API Visión la cual proporciona métodos para la detección de rostros y características faciales. Para la detección de somnolencia se asocian los indicadores visuales más relevantes que reflejan el estado del conductor, específicamente el comportamiento de los ojos, el asentimiento lateral y frontal de la cabeza y el bostezo. El sistema funciona correctamente en condiciones de iluminación natural y sin importar el uso de accesorios por parte del conductor como son lentes de cristales transparentes, audífonos y gorra. Debido a que existe un gran número de accidentes de tránsito causados por múltiples factores ya sean estos por embriaguez, malas maniobras, distracción, fatiga, somnolencia; tomando en cuenta este último se ha desarrollado esta propuesta con el fin de prevenir dichos accidentes brindando un sistema no invasivo, de fácil uso y sin la necesidad de adquirir dispositivos especializados.

PALABRAS CLAVE:

- **SOMNOLENCIA**
- **VISIÓN ARTIFICIAL**
- **SMARTPHONE**

ABSTRACT

This work shows a surveillance system developed to detect and alert the vehicle driver about the presence of drowsiness. We used a smartphone with a mobile application that includes the Android operating system to implement the system. To develop it we used the Java programming language and the artificial vision library Google Mobile API Vision, they provide methods for the detection of faces and facial features. For the detection of drowsiness, the most relevant visual indicators that reflect the driver's condition are the behavior of the eyes, the lateral and frontal assent of the head and the yawn. The system works correctly under natural lighting conditions and without considering the use of accessories of the driver such as glasses, hearing aids and a cap. Due to a large number of traffic accidents that driver has fallen asleep this proposal was developed in order to prevent them by providing a non-invasive system, easy to use and without the necessity of purchasing specialized devices.

KEY WORDS:

- **DROWSINESS**
- **ARTIFICIAL VISIÓN**
- **SMARTPHONE**

CAPÍTULO I

1. INTRODUCCIÓN

1.1. Planteamiento del Problema

Según la Organización Mundial de la Salud (OMS) Ecuador se ubica como el segundo país de Sudamérica con el mayor índice de muertes causadas por accidentes de tránsito. Entre las causas de accidentes, la Agencia Nacional de Tránsito registra que el exceso de velocidad y las distracciones al momento de manejar se encuentran en cuarto puesto. (Ecuador Times, 2013)

Es tan común este causal que muchas marcas de autos han optado por desarrollar sistemas avanzados de asistencia al conductor (SAAC) con el fin de detectar la somnolencia y alertar automáticamente al conductor. Sin embargo, esto aumenta su costo y resultan ser inaccesibles para la mayoría de la población.

Por esta razón, existe la necesidad de implementar un sistema de vigilancia al conductor para la detección de somnolencia basado en un dispositivo usado por la mayoría de conductores como es el caso de los Smartphone, obteniendo un sistema de bajo costo, fácil de usar y de libre acceso.

1.2. Antecedentes

El sueño es una de las necesidades básicas más importantes del ser humano, la falta de sueño hace que el organismo reaccione de una manera ineficiente, reduciendo el tiempo de reacción, la vigilia, el estado de alerta y la concentración de manera que se deteriora la capacidad de realizar actividades basadas en la atención como es el caso de conducir un automóvil.

Según numerosos estudios la somnolencia se encuentra relacionada con miles de accidentes de tránsito cada año, estos tipos de accidentes tiene aproximadamente un 50% más de probabilidades de causar la muerte o lesiones graves, ya que tienden a ser impactos a alta velocidad debido a que el conductor que ha quedado dormido y no puede frenar o desviarse para evitar o reducir el impacto. Para mitigar estos accidentes se han desarrollado sistemas de detección de somnolencia del conductor que reconocen los indicios de un posible estado de somnolencia, alertando al conductor sobre su estado.

Los métodos de detección se dividen en tres grupos principales: los métodos basados en el desempeño del conductor y los métodos que se basan en el estado del conductor. Los métodos centrados en el estado del conductor se dividen en dos subgrupos: los métodos que utilizan señales fisiológicas y los métodos que utilizan técnicas de visión artificial. (Flores, Armingol, & Escalera, 2011)

Los estudios desarrollados mediante técnicas de visión artificial utilizan un sistema de video para capturar la imagen del conductor, un computador o SBC (SINGLE BOARD COMPUTER) para procesar la imagen y obtener de ella los parámetros requeridos por el algoritmo detector de somnolencia y un dispositivo que haga de pasarela entre el sistema y el usuario como es el caso de pantallas táctiles, lcd y módulos de audio.

1.3. Justificación e Importancia

La motivación principal para el desarrollo de este sistema, es asistir tecnológicamente al conductor brindándole un sistema de vigilancia al conductor vehicular basado en técnicas de visión artificial e implementado en un smartphone, dispositivo adquirido por la mayoría de conductores en el país, lo que permite al conductor trabajar con mayor confianza, tranquilidad, seguridad y evitar accidentes en las vías.

1.4. Objetivos

1.4.1. Objetivo general

Implementar un sistema de vigilancia al conductor vehicular basado en técnicas de visión artificial e implementado en un smartphone para la detección y alerta de somnolencia.

1.4.2. Objetivos específicos

- Investigar artículos relacionados con la detección de somnolencia en los conductores.
- Investigar los síntomas de somnolencia en conductores de vehículos.
- Identificar los parámetros técnicos y algoritmos que permitan procesar señales del estado de somnolencia.
- Instalar el software de programación y librerías de visión artificial.
- Desarrollar el algoritmo de detección de somnolencia.
- Desarrollar una interface en la cual se visualice el estado de somnolencia
- Determinar el correcto funcionamiento del sistema implementado.
- Elaborar el informe final y presentación del proyecto sujeto a posibles correcciones.

CAPÍTULO II

2. MARCO TEÓRICO

2.1. Antecedentes investigativos

A continuación se detallan 3 trabajos de investigación de sistemas de detección de somnolencia:

Boon-Giin Lee y Wan-Young Chung (2012), en su trabajo de investigación “Sistema de monitoreo al conductor basado en un smartphone usando fusión de datos” (Boon-Giin & Wan-Young, 2012), proponen un método para monitorear los niveles de seguridad del conductor usando un enfoque de fusión de datos como: características de los ojos, variación de las señales biológicas, temperatura dentro del vehículo y velocidad del vehículo. Este sistema es desarrollado en forma de una aplicación para un smartphone basado en Android, donde la medición de los datos relacionados con la seguridad no requiere gasto monetario o equipo adicional. El sistema tiene una eficacia del 96% para detectar que el conductor se encuentre despierto y un 97% para detectar que se encuentre dormido. Esta información permitió conocer los signos que muestra un conductor somnoliento.

He J. y su equipo de colaboradores en su trabajo de investigación “Detección de fatiga usando Smartphone” (He, Roberson, Peng, Cielocha, & Coltea, 2013) tienen como objetivo utilizar un smartphone (con sistema operativo Android o IOS) para detectar fatiga en el conductor. Usa la cámara frontal del teléfono inteligente para capturar imágenes del conductor y entonces emplea avanzados algoritmos de visión por computador para detectar su cara y ojos. La rotación e inclinación de la cabeza y el parpadeo de los ojos son detectados como indicadores de fatiga. Los algoritmos de visión por ordenador se llevaron a cabo utilizando el lenguaje de programación Java y la librería visión OpenCV. Este trabajo permitió conocer el proceso de detección de somnolencia usando un Smartphone, además de los parámetros fisiológicos utilizados para el desarrollo del algoritmo.

Yeferson Torres en su trabajo de investigación “Aplicación móvil para la detección de somnolencia de un conductor aplicando visión artificial” (Torres, 2016), desarrolla una aplicación móvil para detectar somnolencia en conductores usando visión artificial, basándose específicamente en el comportamiento de los ojos, la aplicación funciona en tiempo real y utiliza el método de Perclos para clasificar el índice de somnolencia que de ser mayor al 80% emite una alerta en forma de sonido. Para desarrollar los algoritmos de visión artificial utiliza la librería OpenCV. El sistema tiene la limitación de solo trabajar en condiciones de luminosidad natural. Este trabajo permitió conocer el método para clasificar el índice de somnolencia denominado Perclos.

2.2. Fundamentación teórica

2.2.1. Somnolencia

a) Definición

La somnolencia es un estado fisiológico con una inclinación a quedarse dormido. Técnicamente es distinta de la fatiga que es la falta de disposición de seguir realizando la misma actividad. La fatiga se produce por realizar tareas que se realizan siempre de la misma forma usando los mismos grupos musculares, su tasa de repetición es elevada y por lo general son realizadas adoptando posturas forzadas tal es el caso de monitorear una pantalla.

Una persona puede estar fatigada sin estar somnoliento pero las condiciones que producen fatiga como la conducción de automóviles en grandes distancias desenmascaran la presencia de somnolencia fisiológica, pero no la causan (Stutts, Wilkins, & Vaughn, 1999).

b) Efectos

Entre los efectos de encontrarse somnoliento tenemos:

- Disminución de la vigilia

- Disminución del tiempo de reacción
- Disminución de la coordinación psicomotora
- Disminución del procesamiento de información

Para el conductor el efecto principal es la retirada progresiva de la atención en las demandas de la carretera, tráfico y señalización, lo que provoca un bajo rendimiento de conducción provocando accidentes.

c) Signos fisiológicos

La mayoría de personas que presentan somnolencia presentan los siguientes signos:

- Parpadeo frecuente
- Frotarse los ojos
- Bostezos repetidos
- Inclinación de la cabeza
- Distracciones

d) Relación de la somnolencia con los accidentes de tránsito

Según la OMS una de las causas más frecuentes que produce accidentes de tránsito es la presencia de somnolencia en conductores. La Dirección General de Tránsito afirma que un 20% de los accidentes tienen relación con esta causa, igualando al número de accidentes producidos por el consumo de alcohol. (StopRonquidos, 2015).

e) Detección de somnolencia

El aumento de los accidentes de tránsito causados por somnolencia ha aumentado el interés de investigadores por desarrollar sistemas de asistencia a la conducción. Estos sistemas usan algunos métodos que pueden ser clasificados en tres categorías como se muestra en la figura 1:

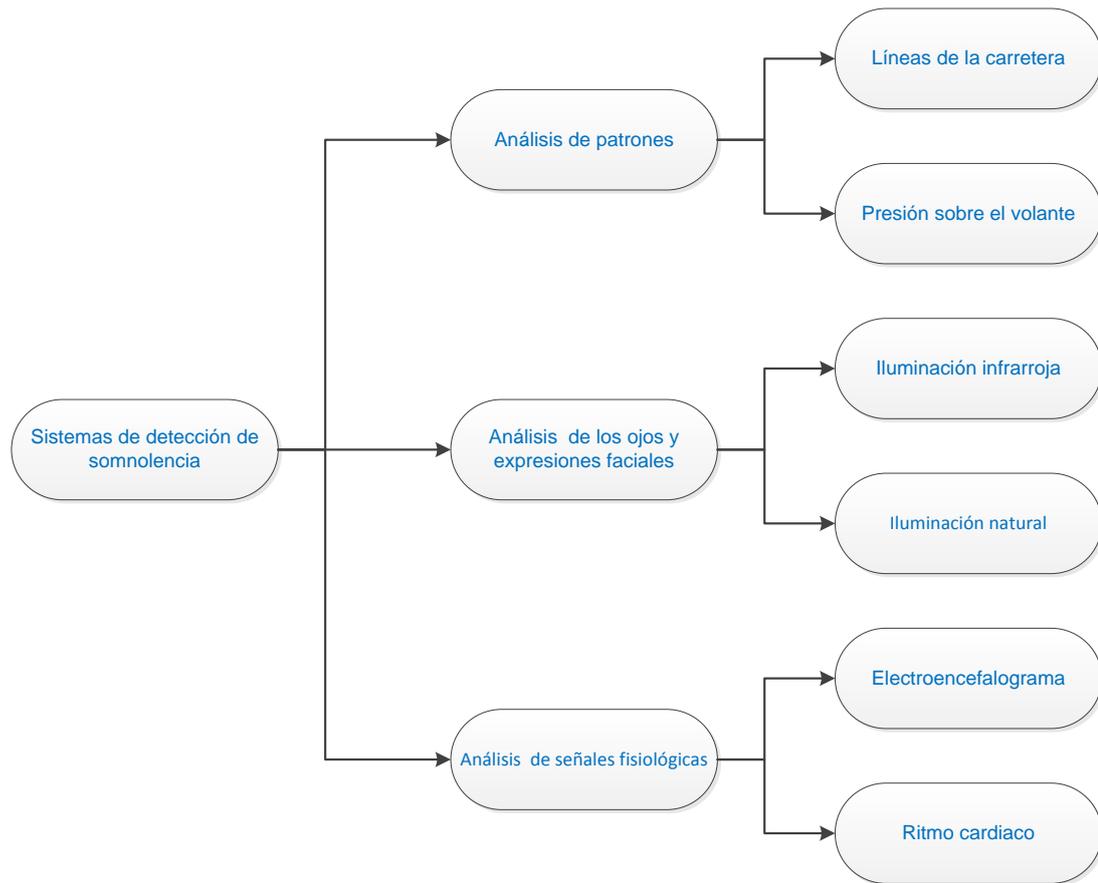


Figura 1 Métodos para detectar somnolencia

A continuación se detalla cada método:

- Análisis basado en patrones de conducción.

Los patrones de conducción son generados en base a variables medibles que son obtenidas experimentalmente. Estas variables pueden ser la velocidad, la aceleración, el frenado, el cambio de marchas, la presión de las manos en el volante y la trayectoria del automóvil en el carril.

Este método tiene la desventaja que su modelado depende de las características propias de cada automóvil y de la forma de conducir que es propia de cada conductor (Flores, Armingol, & Escalera, 2011).

- Análisis basado en cambios físicos de los ojos y las expresiones faciales usando procesamiento de imágenes.

Mediante el procesamiento de imágenes se puede determinar estados del conductor. De la imagen de la cara se puede decir si el conductor se encuentra despierto o dormido. Se puede definir la somnolencia del conductor porque si el conductor está dormido o dormitando sus ojos se encuentran cerrados. Este método tiene la ventaja de no ser intrusivo y se subdivide en 3 técnicas: (Fuletra & Bosamiya, 2013)

- **Técnica de emparejamiento de plantillas**

En este método se definen plantillas del conductor, tanto de sus ojos abiertos como cerrados, luego son comparadas con la imagen actual del conductor y se decide el estado de somnolencia.

- **Técnica del comportamiento de los ojos**

En este método se calcula la frecuencia de parpadeo y la duración del cierre de los ojos para de esta manera determinar el índice de somnolencia.

Uno de los índices más usados para calcular el nivel de somnolencia es PERCLOS (PERcentage of the time Eyelids are CLOSed), que mide el porcentaje de tiempo que los ojos de una persona se encuentran cerrados en un 80% al 100% en un periodo. Según un estudio realizado por Walter Wierwille y sus colegas (Wierwille, et al., 1994) consideran que PERCLOS está entre las más prometedoras medidas de alerta en tiempo real para sistemas de detección de somnolencia en vehículos.

- **Técnica basada en el bostezo**

Este método se basa en la frecuencia de bostezos del conductor. La apertura de la boca de una persona es mayor cuando bosteza que cuando habla normalmente. Se compara la apertura de la boca con un punto de referencia obtenido experimentalmente por el programador y se

calcula las veces que el conductor ha bostezado para generar un índice de somnolencia.

-Análisis basados en el cambio de las medidas fisiológicas

Este método se basa en usar sensores que miden variables fisiológicas del cuerpo humano para analizar estados de somnolencia. Estas variables son: frecuencia cardiaca, actividad cerebral, variabilidad del ritmo cardiaco, respiración, temperatura de la piel periférica, y presión arterial. (Flores, Armingol, & Escalera, 2011) (Jo, Lee, Kim, Jung, & Park, 2011)

2.2.2. Smartphone

Smartphone o teléfono inteligente en español es un dispositivo electrónico que reúne las funcionalidades de un teléfono móvil y las funcionalidades de una computadora. El término inteligente se usa de manera comercial para hacer referencia a la capacidad de usarse como un computador de bolsillo, llegando en algunos casos a reemplazar a una computadora personal.

El smartphone está construido sobre una plataforma móvil similar al sistema operativo de una computadora, lo que lo vuelve de igual manera vulnerable a virus. Una de las características más relevantes es que se pueden instalar aplicaciones logrando aumentar en gran medida su funcionalidad, estas aplicaciones son desarrolladas por el fabricante, el operador o por terceros.

Entre las características básicas de un smartphone tenemos:

- Incluye un sistema operativo
- Capacidad de instalar aplicaciones.
- Envío y recepción de llamadas y mensajes de texto
- Dispone de servicios multimedia.
- Incluye aplicaciones básicas como Reloj, Alarma, Calendario, Calculadora, Juegos, Agenda digital.
- Conexión a internet

- Acceso a redes sociales
- Incluye cámaras fotográficas y video frontales y posteriores
- Lectura y edición de documentos.
- Dispone de sensores biométricos.
- Cuenta con sensores de medición: giroscopio, acelerómetro, barómetro, temperatura.

2.2.3. Sistemas operativos móviles

Un sistema operativo móvil o SO móvil es el software o conjunto de programas que administran los recursos del dispositivo, tanto hardware (teclado, memoria, pantalla, cámara, etc.) como software (aplicaciones e instrucciones). Realiza la carga y ejecución de aplicaciones previamente instaladas. Todo sistema operativo ofrece una interfaz gráfica amigable compuesta por botones e imágenes que permiten al usuario comunicarse con el dispositivo.

El SO de un móvil, al igual que el SO de una computadora, viene cargado al momento de comprarlo, pero si el usuario decide y si es compatible puede actualizarlo y/o cambiarlo (GCFAprendeLibre, 2016).

a) Estructura de un SO móvil

Los SO móviles en general están contruidos por las siguientes capas:

- **Kernel**

El kernel o núcleo es el corazón del sistema operativo que proporciona acceso seguro a todos los componentes hardware del dispositivo móvil. Además es el encargado de brindar distintos servicios a las capas superiores como son los controladores o drivers para el hardware, gestión de procesos, sistemas de archivos, además del acceso y administración de la memoria del sistema.

Cuando más de una aplicación se quiere comunicar con el mismo hardware, es el kernel quien decide que aplicación se comunica y durante cuánto tiempo, esto se lo conoce como multiplexado.

Acceder de forma directa al hardware puede ser una tarea difícil por lo se implementa un conjunto de abstracciones de hardware cuya función es esconder su complejidad.

- **Middleware**

Esta capa es un conjunto de módulos que permiten que las aplicaciones puedan ser ejecutadas. Es transparente para el usuario y presta servicios para que aplicaciones en capas superiores puedan ejecutarse, estos servicios son:

- Motor de mensajería y comunicaciones
- Códec multimedia
- Intérpretes de páginas web
- Gestión del dispositivo
- Funciones de seguridad.

- **Entorno de ejecución de aplicaciones**

El entorno de ejecución de aplicaciones provee de todos los elementos necesarios para que programadores puedan crear y desarrollar aplicaciones.

- **Interfaz de usuario**

La interfaz de usuario hace posible que el operador interactúe con el diseño gráfico de la aplicación. Está compuesta por una serie de elementos como botones, pantallas, teclado y en los últimos años se han creado las pantallas táctiles que facilitan al operador controlar las funcionalidades del móvil con los dedos.

b) Sistemas operativos móviles más usados

La creciente demanda de dispositivos inteligentes ha aumentado el desarrollo de SO móviles. La cuota de mercado de los SO en el trimestre de 2016 era el siguiente (muycomputer, 2016) :

- Android 84, 1 %
- IOS 14, 8 %
- Windows Phone 0, 7 %
- BlackBerry OS 0, 2 %
- Otros 0, 2 %

2.2.4. Android

Android es un sistema operativo basado en el kernel de Linux versión 2.6 y diseñado para dispositivos con pantalla táctil como celulares inteligentes, relojes, tabletas, televisores, entre otros. Es desarrollado por la Open Handset Alliance (OHA) que es liderada por Google. La Open Handset Alliance es un consorcio compuesto por más de 80 empresas de hardware, software y telecomunicaciones, creado en 2007 y dedicado al desarrollo de estándares abiertos para dispositivos móviles.

Android Inc., es la empresa que dio origen a este SO móvil, fue apoyada económicamente por Google y años más tarde la compró. En un inicio fue desarrollado como software para cámaras digitales.

a) Características:

Las principales características del sistema operativo Android son:

- Código abierto y software libre
- Multitarea
- Adaptable a diferentes pantallas y resoluciones
- Interface personalizable
- Tiene más de un fabricante
- Tiene su propia máquina virtual Dalvik

- Usa la base de datos Sqlite.
- Posee plataformas de desarrollo en Linux, Windows y Mac
- Navegador web integrado basado en el motor WebKit
- Las aplicaciones se escriben en lenguaje Java
- Utiliza Google Play Store para descarga de aplicaciones y de contenido
- Tiene soporte tanto para la arquitectura ARM y para la arquitectura x86

b) Arquitectura

La Arquitectura del sistema operativo Android está formada por las siguientes capas que se presentan en la figura2:

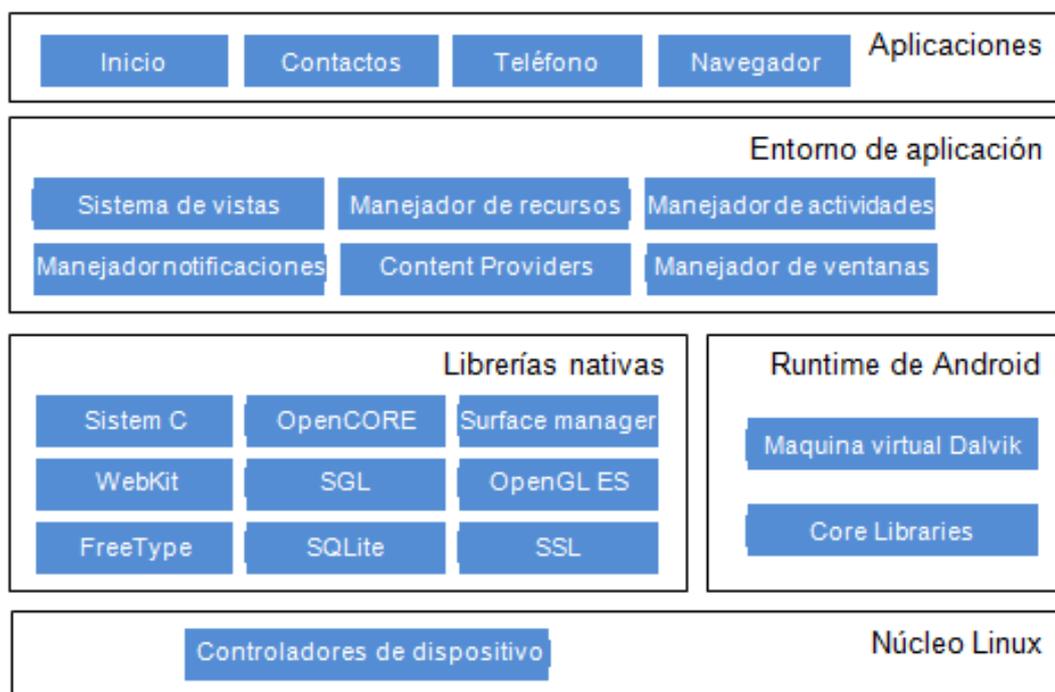


Figura 2 Arquitectura de Android

Fuente: (Gironés, 2012)

Kernel de Linux: Esta capa de la arquitectura actúa como una capa de abstracción de hardware HAL, en donde una aplicación no accede de manera directa a un componente hardware sino que lo hace a través de la

capa abstracta HAL, la cual contiene los drivers necesarios. Por lo tanto, es la única capa que es dependiente del hardware.

El SO Android está basado en los servicios del núcleo de Linux versión 2.6 como son:

- Controladores hardware
- Gestión de energía
- Gestión de procesos
- Gestión de seguridad
- Gestión de memoria
- Pila de red, etc.

Librerías Nativas: Android incluye un conjunto de librerías escritas en código nativo del procesador (C/C++) usadas por varios componentes del sistema, algunas son (Molina, Sandoval, & Toledo, 2012):

- **Libc:** Incluye cabeceras y funciones del lenguaje C.
- **Surface Manager:** Compone los diferentes elementos de navegación y ventanas en la pantalla.
- **SGL:** Soporte a gráficos 2D.
- **OpenGL ES:** Soporte a gráficos 3D.
- **Media Framework:** Proporciona códecs necesarios para múltiples formatos de audio, video e imagen
- **SQLite:** Gestor de bases de datos.
- **WebKit:** Proporciona un motor para aplicaciones tipo navegador
- **Free Type:** Trabaja con distintos tipos de fuentes.
- **SSL:** Soporte a comunicaciones seguras.

Runtime de Android: Es un entorno de ejecución constituido por dos componentes:

- **Librerías del núcleo:** Son un conjunto de librerías escritas en lenguaje java que proveen la mayor parte de las funcionalidades de las librerías nativas.

- **Dalvik VM:** Es una máquina virtual optimizada para dispositivos móviles, tomando en consideración la poca memoria, las limitaciones del procesador y la alimentación por batería. Dalvik no trabaja directamente con el bytecode de java sino que los transforma en un código más eficiente creado para procesadores pequeños, esta transformación se lleva a cabo mediante la herramienta “dx” incluida en el SDK de Android que convierte los ficheros .class de java en ficheros .dex y luego los compila en un fichero de extensión .apk (Android package) que es el formato de archivo utilizado para la instalación de aplicaciones en los dispositivos móviles. (Aranaz, 2009)

Entorno de aplicación: Es un conjunto de herramientas diseñadas para el desarrollo de aplicaciones. Esta capa ha sido diseñada para la reutilización de componentes, cuando una aplicación publica sus capacidades cualquier otra aplicación puede hacer uso de estas (Molina, Sandoval, & Toledo, 2012).

Aplicaciones Android: Las aplicaciones Android pueden ser escritas en lenguaje Java usando el kit de desarrollo de software SDK y en lenguaje c/c++ usando el kit de desarrollo Nativo o en Python a través de la capa de software para Android SL4A.

c) Entorno de desarrollo

La plataforma integral de desarrollo soportada oficialmente es Android Studio junto con el ADT (Android Development Tools plug-in), provee un emulador para poder realizar pruebas sin tener un dispositivo físico conectado en nuestro ordenador.

d) Documentación

Android tiene dedicada un web para desarrolladores que contiene documentación detallada para el desarrollador que explica cómo utilizar las distintas API del framework. Además, tiene un foro para desarrolladores, tutoriales que sirven como código de ejemplo y videos.

2.2.5. IOS

El sistema operativo iOS es desarrollado por la empresa Apple Inc., en un inicio fue usado en el iPhone y posteriormente en el iPod y iPad. Está basado en el sistema operativo Mac OS X y solo puede ejecutarse en dispositivos de la empresa Apple (Porrás, 2012).

a) Características

- Para ahorrar batería la multitarea se reserva específicamente para APIs definidas.
- La Actualización de su sistema se puede hacer través de una conexión wifi (vía on the air OTA), sin la utilización de un ordenador personal y sin tener que conectarlo a iTunes.
- Incorpora tecnología multi-touch, la cual es capaz de reconocer múltiples gestos y toques en la pantalla.
- Posee un sistema de seguridad infalible que inutiliza los equipos en caso de robo o pérdida.

b) Arquitectura

IOS está constituido por cuatro capas o niveles de abstracción como se observa en la figura 3:

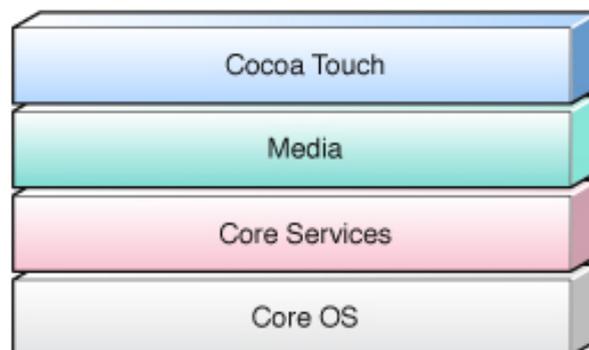


Figura 3 Arquitectura de IOS

Fuente: (Porrás, 2012)

- **Cocoa Touch:** Es el conjunto de herramientas usadas para el desarrollo de aplicaciones IOS. Los lenguajes principales de programación usados son C y Objective-C que es una extensión del lenguaje c.
- **Media:** Esta capa contiene todas las tecnologías necesarias para acceder a ficheros multimedia (imágenes, audio, video, etc.)
- **Servicios principales (Core Services):** Esta capa contiene los servicios principales del sistema y que pueden ser usados por otras aplicaciones, como: base de datos SQLite, acceso a la red, acceso a la red, ubicación del dispositivo.
- **Núcleo (Core OS):** Esta capa contiene el Kernel, los ficheros del sistema y la infraestructura de red, seguridad, manejo de la memoria y los drivers del dispositivo.

c) Entorno de desarrollo

Xcode es el IDE utilizado para escribir, compilar, ejecutar y depurar el código de aplicaciones IOS. Utiliza el lenguaje de programación Objective-C.

d) Documentación

Al igual que Android, Apple tiene dedicada una web para desarrolladores que contiene una guía detallada que explica cómo utilizar las distintas APIs del framework, guías de utilización de las herramientas y guías de programación y videos explicativos.

2.2.6. Aplicaciones Android

Los componentes básicos de una aplicación Android son (Gironés, 2012):

- **Vista (view):** Son los elementos necesarios para construir el diseño gráfico de la aplicación. Entre estos elementos tenemos: botones, cuadros de texto, listas desplegables, imágenes, etc. Las vistas son objetos descendientes de la clase View y deben ser declaradas en un fichero XML.

- **Layout:** Es el conjunto de vistas agrupadas de una manera determinada. Los layouts son objetos descendientes de la clase View y deben ser declarados en un fichero XML.
- **Actividad (Activity):** Es cada una de las pantallas que conforma la interface de usuario y refleja una determinada acción. Dentro de una aplicación puede existir más de una actividad, las cuales son independientes entre sí pero con un objetivo en común. Toda actividad pertenece a una clase descendiente de Activity y deben ser declaradas en un fichero XML.
- **Servicio (Service):** Son tareas no visibles que se ejecutan en segundo plano.
- **Intención (Intent):** Un intent es una petición para que una acción sea llevada a cabo por otro componente o aplicación.
- **Receptor de anuncios (Broadcast receiver):** Este componente es el encargado de recibir y reaccionar ante distintas notificaciones generadas por el sistema (llamada entrante, mensaje recibido, etc.) o por otras aplicaciones que puedan lanzar notificaciones del tipo broadcast. Debe ser declarado en un fichero XML.
- **Proveedores de contenido (Content Provider):** Android ha definido Content Provider como el mecanismo para compartir o modificar datos entre aplicaciones. Debe ser declarado en un fichero XML.

a) Ciclo de vida de una actividad.

Una aplicación Android corre dentro de su propio proceso Linux. Este proceso es creado para la aplicación y continuará vivo hasta que ya no sea requerido y el sistema reclame su memoria para asignársela a otra aplicación (Gironés, 2012).

El ciclo de vida de una aplicación en Android es controlado por el sistema y puede encontrarse en uno de los siguientes 3 estados:

- **Activa (Running):** La actividad tiene el foco de la acción, es visible y el usuario se encuentra interactuando con ella.

- **Detenida (Paused):** La actividad es visible aunque es posible que no tenga el foco de acción por existir otras actividades superpuestas con las que el usuario está interactuando.
- **Interrumpida (Stopped):** Cuando la actividad no es visible pero se encuentra en ejecución.

b) Métodos

Cada vez que una actividad cambia de estado se van a producir eventos que podrán ser capturados por los siguientes métodos (Invarato, 2014):

- **OnCreate (Bundle):** este método es llamado cuando la actividad es iniciada. Aquí se realiza todo tipo de inicializaciones. Además entrega datos en un Bundle si la actividad ha sido re-creada.
- **Onstart ():** este método es llamado cuando la aplicación se hace visible para el usuario.
- **OnResume ():** este método es llamado al comenzar la iteración con el usuario, en este momento la actividad se sitúa en primer plano.
- **OnPause ():** este método es llamado cuando la actividad ya no interactúa con el usuario y va a ser puesta a segundo plano para dar paso a otra actividad.
- **OnStop ():** este método es llamado cuando la actividad pasa a segundo plano y no es visible para el usuario, normalmente porque otra actividad ha sido lanzada y ha cubierto a ésta.
- **OnRestart ():** este método es llamado cuando la actividad fue pausada y vuelve a ejecutarse

2.2.7. Visión artificial

La visión artificial forma parte de la inteligencia artificial y consiste en un proceso computacional que tiene como objetivo la extracción, caracterización e interpretación de información de imágenes digitalizadas.

Entre las aplicaciones que utilizan visión artificial tenemos:

- Detección de rasgos faciales.

- Reconocimientos de rostros.
- Análisis médico por imágenes.
- Interpretación de video vigilancia.
- Inspección industrial y control de calidad.
- Visión en máquinas y robots.

a) Etapas de un sistema de visión artificial

Las etapas de un sistema de visión artificial en general se muestran en la figura 4:

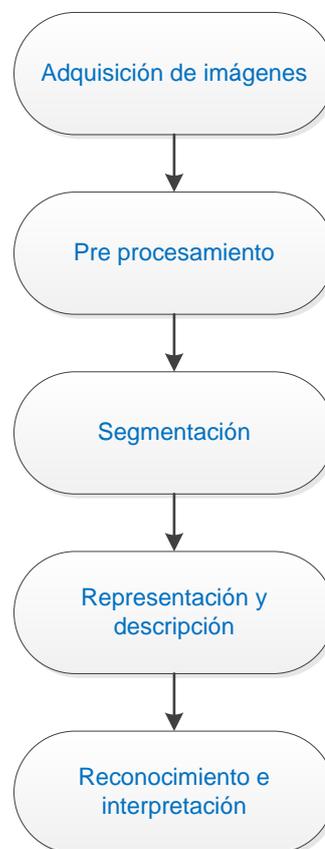


Figura 4: Etapas de un sistema de visión artificial

A continuación se detallara cada etapa (Sistemas Adaptativos y Bioinspirados en Inteligencia Artificial S.A.B.I.A, 2010) (López, 2016):

- **Adquisición de imágenes:** Es el proceso que nos lleva a la obtención de una imagen mediante cámaras o tarjetas de adquisición de imágenes.

- **Pre procesamiento:** En esta etapa se reduce características no deseadas en la imagen como el ruido y se realzan características importantes que permitan realizar de una mejor manera la etapa de segmentación.
- **Segmentación:** En esta etapa se diferencian los objetos de interés de la imagen.
- **Representación y descripción:** En esta etapa se extraen características que nos permitan identificar un objeto de otro. Las características son basadas en la textura, color, área, perímetro, entre otros.
- **Reconocimiento e interpretación:** En esta etapa se identifican los objetos y se les asigna un significado.

b) Herramientas de un sistema de visión artificial en Android

OpenCV: OpenCV (Open Source Computer Vision Library) es una librería de Fuente abierta de visión por ordenador. Es desarrollada por Intel y su primera versión apareció en 1991, se distribuye bajo licencia BSD lo que facilita a los usuarios emplear, modificar y distribuir el código con o sin modificaciones (García, 2007).

La librería cuenta con más de 2500 algoritmos optimizados que se pueden utilizar para detectar y reconocer las caras, identificar objetos, clasificar las acciones humanas en los vídeos, seguimiento de objetos en movimiento, encontrar imágenes similares de una base de datos de imágenes, eliminar los ojos rojos de las imágenes tomadas con flash, seguir los movimientos de los ojos, etc. OpenCV tiene más de 47 mil usuarios en su comunidad y el número estimado de descargas es superior a los 7 millones (OpenCV, 2016).

Esta librería es multiplataforma, dispone de versiones para Windows, Linux, Mac os y ha sido escrita en código C y C++ optimizados. Desde la versión 2.3.0 dispone de un SDK OpenCV para Android lo que permite añadir visión artificial a las aplicaciones desarrolladas para dispositivos móviles.

FastCv: FastCV es una librería de visión por ordenador optimizada para dispositivos móviles, es desarrollada por Qualcomm, puede descargarse de forma gratuita y tiene una licencia de uso muy flexible. FastCV ofrece dos versiones a los desarrolladores .Por un lado, una versión está optimizada para procesadores ARM (ARM es el procesador más extendido en dispositivos móviles), mientras que otra aprovecha las características del procesador Snapdragon. Ambas versiones comparten el mismo API (Déniz, Salido, & Bueno, 2013).

Esta librería cuenta con algoritmos para reconocimiento de gestos, detección, reconocimiento y seguimiento de rostros, reconocimiento y seguimientos de textos, realidad aumentada, etc. (Qualcomm Developer Network, 2016)

Google Mobile Visión Api: Google provee de un conjunto de interfaces de programación de visión para dispositivos móviles API de código abierto, las API que ofrece son de detección y seguimientos de rostro, scanner de código de barras, reconocimiento de texto (Echessa, 2015) (Maxwell, 2015).

El código utilizado en los algoritmos de la librería fueron creados tomando en cuenta las limitaciones que tienen los dispositivos móviles y su éxito se basa en que fueron diseñados y optimizados por la misma empresa propietaria de Android.

2.3. Variables de la investigación del proyecto en ejecución

- **Variable independiente**
Sistema de vigilancia al conductor vehicular basado en técnicas de visión artificial e implementado en un smartphone.
- **Variable dependiente**
Detección y alerta de somnolencia.

2.4. Hipótesis

- La implementación de un sistema de vigilancia al conductor vehicular basado en técnicas de visión artificial e implementado en un smartphone permitirá la detección y alerta de somnolencia.

CAPÍTULO III

3. ANÁLISIS, DISEÑO E IMPLEMENTACIÓN

3.1. ANÁLISIS

En esta sección se pretende dar respuesta a diferentes aspectos del sistema previo a su implementación, se incluirán los requerimientos funcionales así como las herramientas usadas, casos de uso y dispositivos seleccionados para las pruebas del sistema.

3.1.1. Descripción del sistema

Un esquema general del sistema se observa en la figura 5:

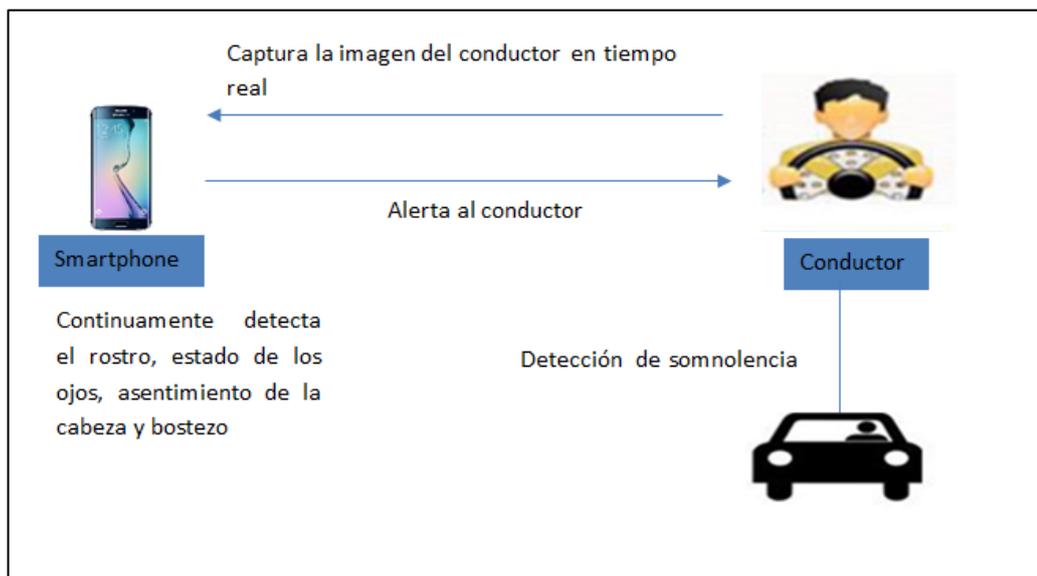


Figura 5 Esquema del sistema de vigilancia al conductor

El sistema tiene como función principal detectar y alertar al conductor sobre su estado de somnolencia. Este es implementado en un Smartphone, en el que normalmente se incluye una cámara frontal mediante la cual se captura la imagen del conductor para ser procesada y mediante métodos de visión artificial, se detectan parámetros fisiológicos que indican la presencia de somnolencia en el conductor. Dependiendo de los parámetros detectados, el sistema genera una alerta gráfica que se visualiza en la

pantalla del smartphone y una alerta sonora que es reproducida por el mismo.

3.1.2. Requerimientos de la aplicación

a) Requerimientos funcionales de la aplicación

En la tabla 1 se especifica las actividades y operaciones que nuestro sistema debería ser capaz de realizar.

Tabla 1

Requerimientos funcionales de la aplicación

Requerimiento	Descripción
Adquirir la imagen	El sistema debe capturar la imagen del conductor mediante la cámara frontal del smartphone
Detección el rostro	El sistema debe detectar el rostro del conductor en la imagen adquirida
Detección del estado de los ojos	El sistema debe detectar los ojos y clasificarlos como abiertos o cerrados
Detección de distracción	El sistema debe detectar si el foco de la mirada del conductor no se encuentra en el frente.
Detección de Bostezo	El sistema debe detectar si el conductor se encuentra bostezando

CONTINÚA 

Detección de Asentimiento de la cabeza	El sistema debe detectar si el conductor asiente la cabeza
Determinación del nivel de somnolencia	El sistema determina el nivel de somnolencia en el que se encuentra el conductor de acuerdo a la detección de varios parámetros que indiquen presencia de somnolencia.
Generación de una alarma gráfica y sonora	De acuerdo a la determinación del nivel de somnolencia, el sistema debe mostrar una alarma gráfica a manera de texto, que permita al conductor visualizar su estado y además una segunda alarma de tipo sonora, en el caso de que no se percate de la alarma gráfica.
Diseño de una interface de usuario	La Interface de usuario debe ser de fácil uso y amigable con el usuario

b) Requerimientos no funcionales de la aplicación

En la tabla 2 se especifican los requerimientos que se deben tomar en cuenta al momento de implementar la aplicación.

Tabla 2

Requerimientos no funcionales de la aplicación

Requerimiento	Descripción
IDE gratuito	Para la programación de la

CONTINÚA 

	aplicación se debe usar un IDE de programación gratuito.
Librería de visión artificial gratuita	Para la detección de parámetros fisiológicos del conductor mediante visión artificial se debe usar una librería de visión artificial gratuita.
Idioma	La UI del sistema debe estar en español, con buena ortografía y con textos entendibles para el usuario.
Posición Smartphone	La posición del teléfono dentro del vehículo debe ser la adecuada para que funcione de manera correcta. El dispositivo debe ser colocado al frente del rostro del conductor por debajo de la línea de vista (aproximadamente 20 grados) y a una distancia aproximada del largo de los brazos
Condiciones de iluminación	La aplicación debe trabajar correctamente en condiciones de iluminación natural y adecuada.
Smartphone	El smartphone donde se ejecute la aplicación para su correcto funcionamiento deberá tener los siguientes requisitos mínimos: -Sistema operativo: Android 4.1 o superior que incluya Google Play Store.

- Cámara frontal de 2 Megapíxeles.
- CPU de 8 núcleos con una velocidad 1.2 GHz
- Espacio disponible de 15 MB en memoria para instalar la aplicación
- Acceso a internet la primera vez que se instale la aplicación, esto para que se descargue los recursos de la librería de visión.

3.1.3. Recursos

Para el desarrollo del sistema se consideraron los recursos que se ilustran en la tabla 3:

Tabla 3

Recursos de hardware y software usados por el sistema

Recursos	Equipo o instrumento	Descripción
Hardware	Smartphone	Es el dispositivo en donde se instala y ejecuta la aplicación.
	Computadora	Es el equipo que nos permite desarrollar la aplicación
	Soporte para celular	Es el equipo que nos permite posicionar el celular en frente del conductor, es ajustable para distintos tamaños

CONTINÚA 

de smartphone		
Software	Android Studio	Es el entorno de desarrollo integrado oficial para la creación de aplicaciones Android el cual viene incluido con el Android SDK y es gratuito.
	API Google Mobile Visión	Es una librería gratuita que permite incluir herramientas de visión artificial en aplicaciones Android

3.1.4. Dispositivos seleccionados para las Pruebas

Para el desarrollo de las pruebas se usó un smartphone modelo Huawei P8 Lite con características especificadas en la tabla 5.

Tabla 4

Características técnicas del smartphone Huawei P8 Lite

Datos Técnicos	
Diseño	
Alto	146 mm
Ancho	72 mm
Grosor	7,5mm
Peso	147 gramos
Pantalla	

CONTINÚA 

Tamaño	5 pulgadas
Resolución	Full HD 1.920x1.080 pixeles
Hardware y rendimiento	
Procesador	MediaTek MT6753T xCortex-A53 64-bit a 1.5GHz
Núcleos	8 núcleos
Memoria RAM	2GB
Memoria Interna	16 GB
Cámara Frontal	5 Megapíxeles
Software	
Sistema operativo	Android Lollipop
Versión SO	5.0.2

Se usó este dispositivo porque era el que en ese momento se tenía disponible y cumplía con los requisitos del sistema.

3.2. DISEÑO

En esta sección se detalla el diseño del sistema mediante diagramas o esquemas, considerando la mejor alternativa al integrar aspectos técnicos y funcionales.

3.2.1. Diagrama de flujo principal

En la figura 6 se muestra el diagrama de flujo principal del sistema de vigilancia al conductor, como también en la tabla 6 se muestra una descripción de los subprocesos.

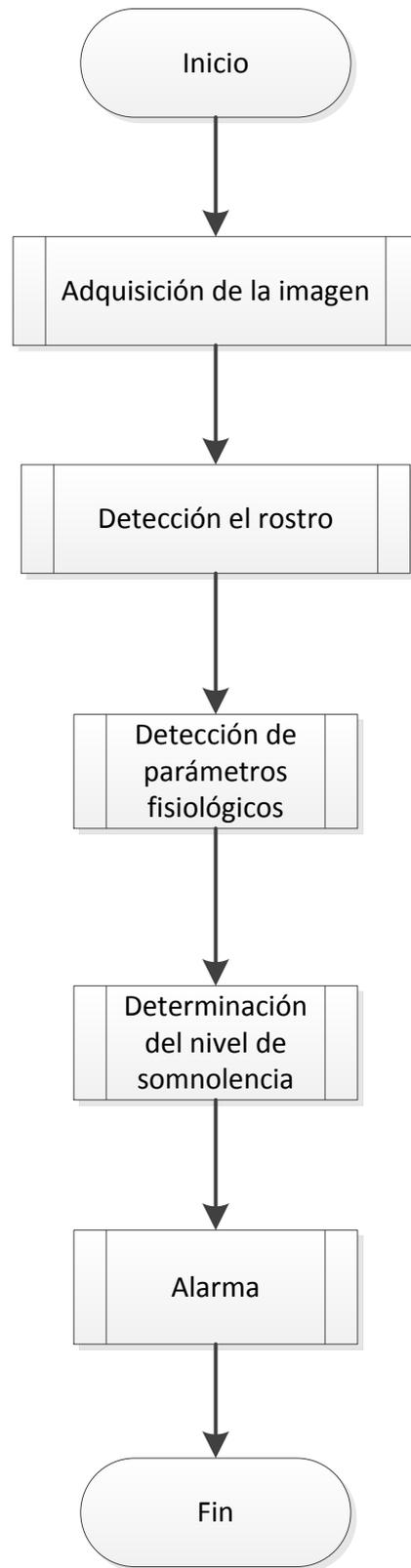


Figura 6 Diagrama de flujo principal

Tabla 5

Subprocesos del diagrama de flujo principal

Subproceso	Descripción
Adquisición de la imagen	En este subproceso se captura la imagen del conductor mediante la cámara del smartphone
Detección rostro	En la imagen adquirida se detecta el rostro del conductor
Detección de parámetros fisiológicos	Se detectan parámetros fisiológicos del conductor que nos permitan medir el estado de somnolencia del conductor.
Determinación del nivel somnolencia	Mediante un algoritmo que tiene como entrada parámetros fisiológicos se determina el nivel sobre el estado de somnolencia del conductor
Alarma	Se genera una alarma gráfica y sonora para avisar al conductor sobre su estado.

3.2.2. Diagrama de casos de uso

El siguiente diagrama de caso de uso tiene como finalidad modelar la interacción existente entre el usuario conductor y el sistema.

En la figura 7 se presenta el diagrama de flujo principal del sistema de vigilancia al conductor, como también en la tabla 7 se describen los subprocesos.

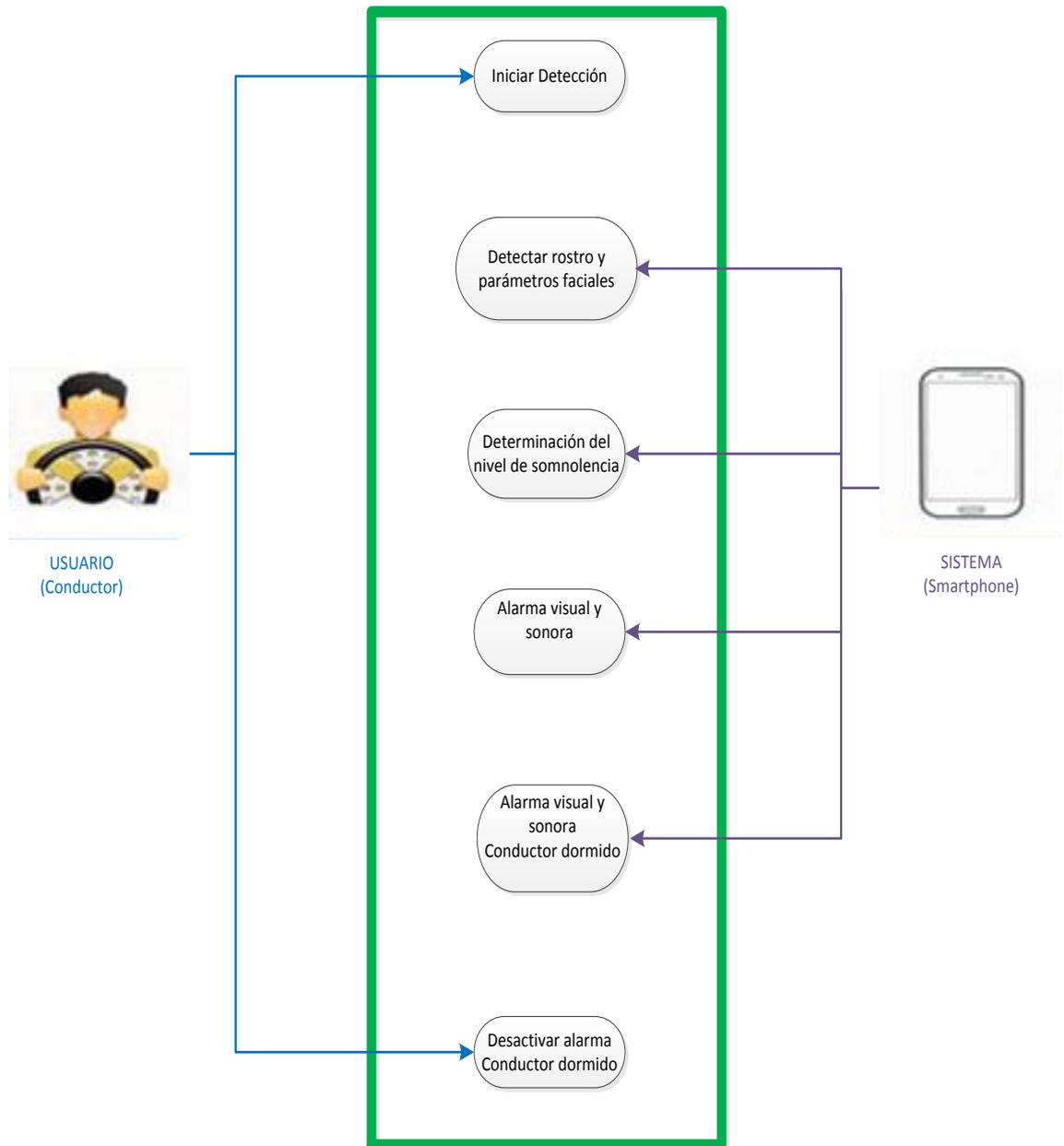


Figura 7 Figura Diagrama de caso de uso principal

Tabla 6**Subprocesos del diagrama de caso de uso principal**

Subproceso	Descripción
Iniciar Detección	El usuario inicia el proceso de detección de somnolencia cuando presiona el botón "Iniciar detección"
Detectar rostro y parámetros faciales	El sistema detecta el rostro y parámetros faciales en la imagen adquirida la cámara del dispositivo
Detección del nivel de somnolencia	El sistema genera un nivel sobre el estado del conductor
Alerta visual y sonora	El sistema alerta al conductor mediante una alerta visual y sonora
Alerta visual y sonora "conductor dormido"	El sistema generara una alarma particular cuando detecta que el conductor se ha quedado dormido y solo pueda ser desactivada mediante la pulsación de un botón
Desactivar alerta "conductor dormido"	El usuario desactiva la alarma de "conductor dormido" mediante la pulsación del botón

3.2.3. Diseño del algoritmo para la detección de factores visuales que indican somnolencia

De los factores visuales que indican somnolencia en este proyecto se va a analizar los siguientes:

De los movimientos de la cabeza:

- Distracción
- Asentimiento de la Cabeza

De la expresión facial:

- Bostezo

Del análisis acular:

- PERCLOS (Porcentaje de tiempo en el que los ojos se encuentran cerrados)
- Microsueño

3.2.4. Movimientos de la cabeza

a) Distracción

En el estado de somnolencia, la atención del conductor disminuye y empieza a distraerse fácilmente. La distracción se presenta con la particularidad de que el conductor deja de enfocar su atención en la carretera y comienza a mirar hacia otros lados como se observa en la figura 8.

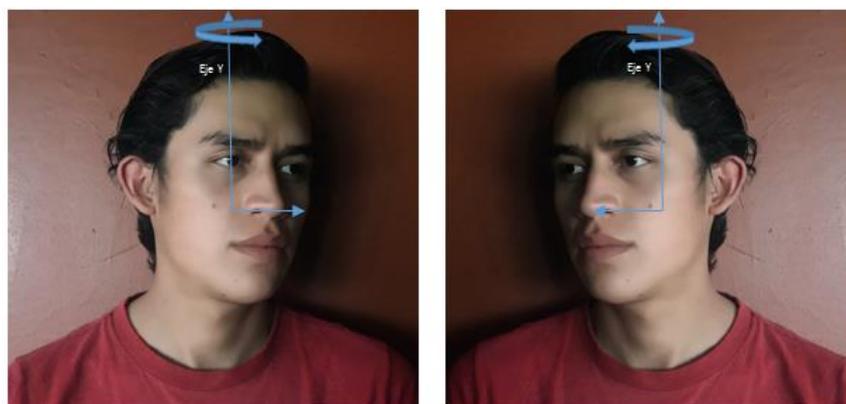


Figura 8 Dirección del rostro a) mirada hacia izquierda b) mirada hacia la derecha

Diagrama de flujo del algoritmo para de la detección de distracción

En la figura 9 se muestra el diagrama de flujo del algoritmo para la detección de distracción.

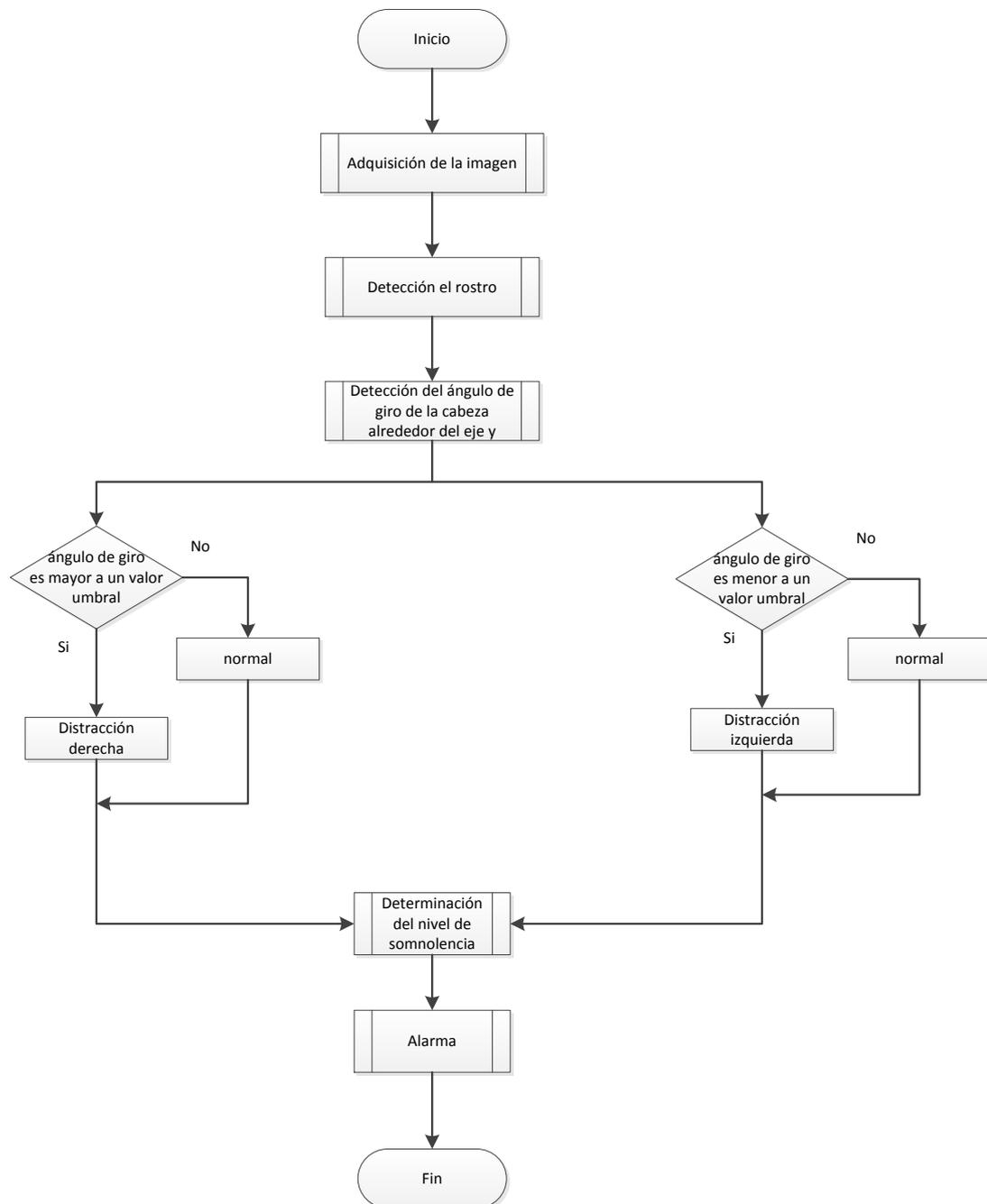


Figura 9 Diagrama de flujo del algoritmo para la detección de distracción

Una vez detectado el rostro en la imagen adquirida por el sistema, se calcula el ángulo de giro de la cabeza y se lo compara con un valor umbral, si este

es mayor, se determina como distracción derecha y si es menor como distracción izquierda.

b) Asentimiento de la cabeza

Cuando un conductor se encuentra somnoliento algunos de los músculos del cuerpo comienzan a relajarse, lo que causa asentimiento de la cabeza (Jimenez, 2011).

En un conductor somnoliento el asentimiento de la cabeza se presenta como un movimiento abrupto de la cabeza hacia abajo o hacia alguno de los lados.

- **Asentimiento frontal de la cabeza:** Para determinar el movimiento de asentir la cabeza hacia abajo se utiliza el algoritmo planteado en (Saradadev & Bajaj, 2008), donde se establece una posición relativa P1 formado por las coordenadas (P1x, P1y), dadas por el valor medio de la distancia entre el centro del ojo izquierdo y derecho, tal como se muestra en la figura 10, se toma el valor del P1 en el tiempo 0 segundos P1[0 seg] y dentro de 1 segundo P2[1 seg], este valor será la nueva posición de la cabeza. Si la diferencia entre P2 [1 seg] - P1 [0 seg] es mayor a un umbral determinado se considera como un cabeceo.

En la figura 10 se observa la variación de P2-P1 cuando existe un asentimiento frontal.

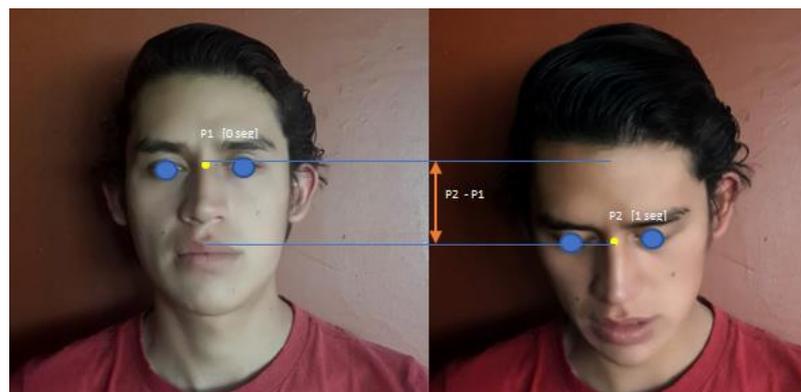


Figura 10 Asentimiento de la cabeza de manera frontal

Diagrama de flujo del algoritmo para de la detección de asentimiento frontal de la cabeza

En la figura 11 se muestra el diagrama de flujo del algoritmo para la detección de asentimiento frontal.

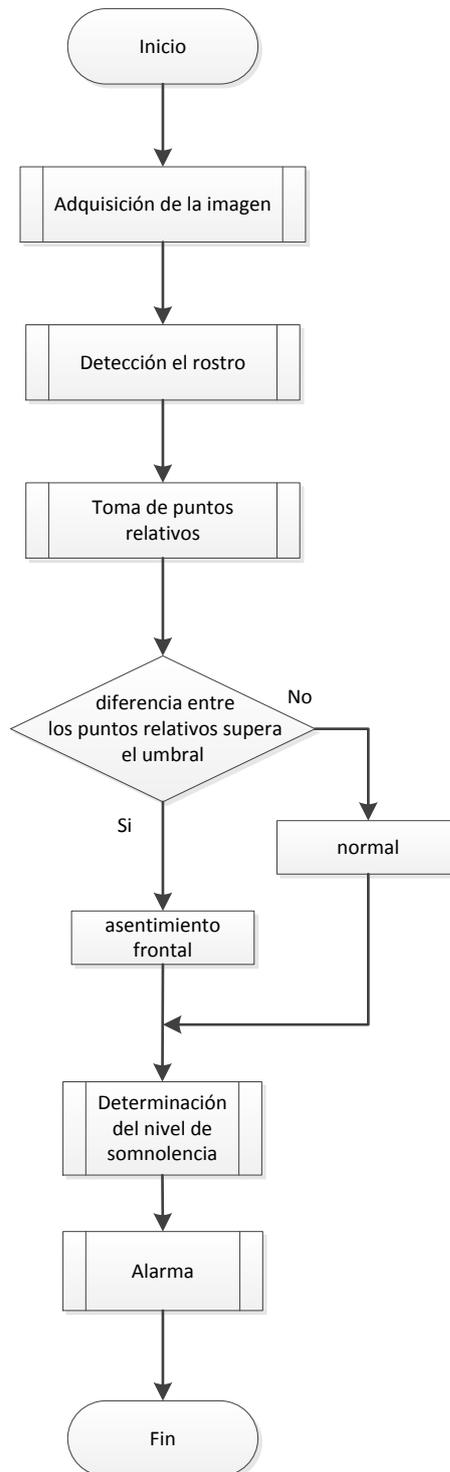


Figura 11 Diagrama de flujo para el algoritmo de detección de asentimiento frontal de la cabeza

Una vez detectado el rostro en la imagen adquirida por el sistema, se establece y se toma la posición de los puntos relativos, P1 en 0 segundos y

p2 después de 1 segundo, se procede a calcular la distancia entre estos dos puntos, si supera un valor umbral se determina como asentimiento frontal.

-Asentimiento Lateral de la cabeza: El asentir la cabeza hacia los lados se puede determinar midiendo el ángulo que se inclina la cabeza en dirección hacia la izquierda o derecha como se observa en la figura 12 , si este ángulo sobrepasa un valor umbral determinado, se considera como positivo.

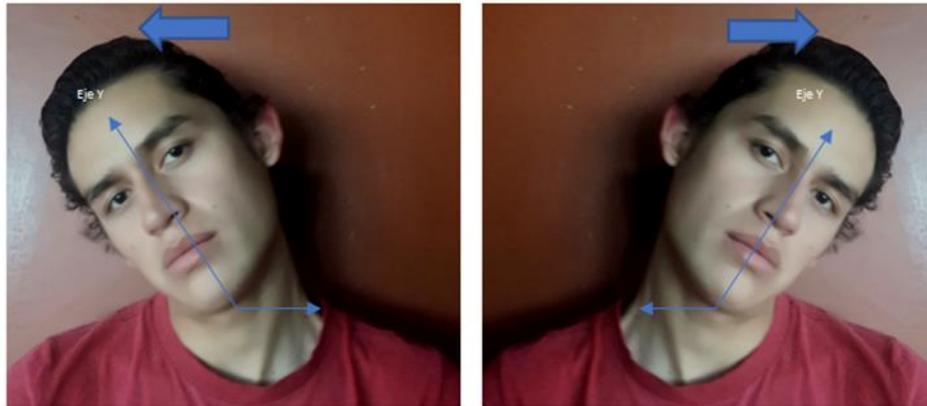


Figura 12 Asentimiento de la cabeza de manera lateral

Diagrama de flujo del algoritmo para de la detección de asentimiento lateral de la cabeza

En la figura 13 se muestra el diagrama de flujo del algoritmo para la detección de asentimiento lateral de la cabeza.

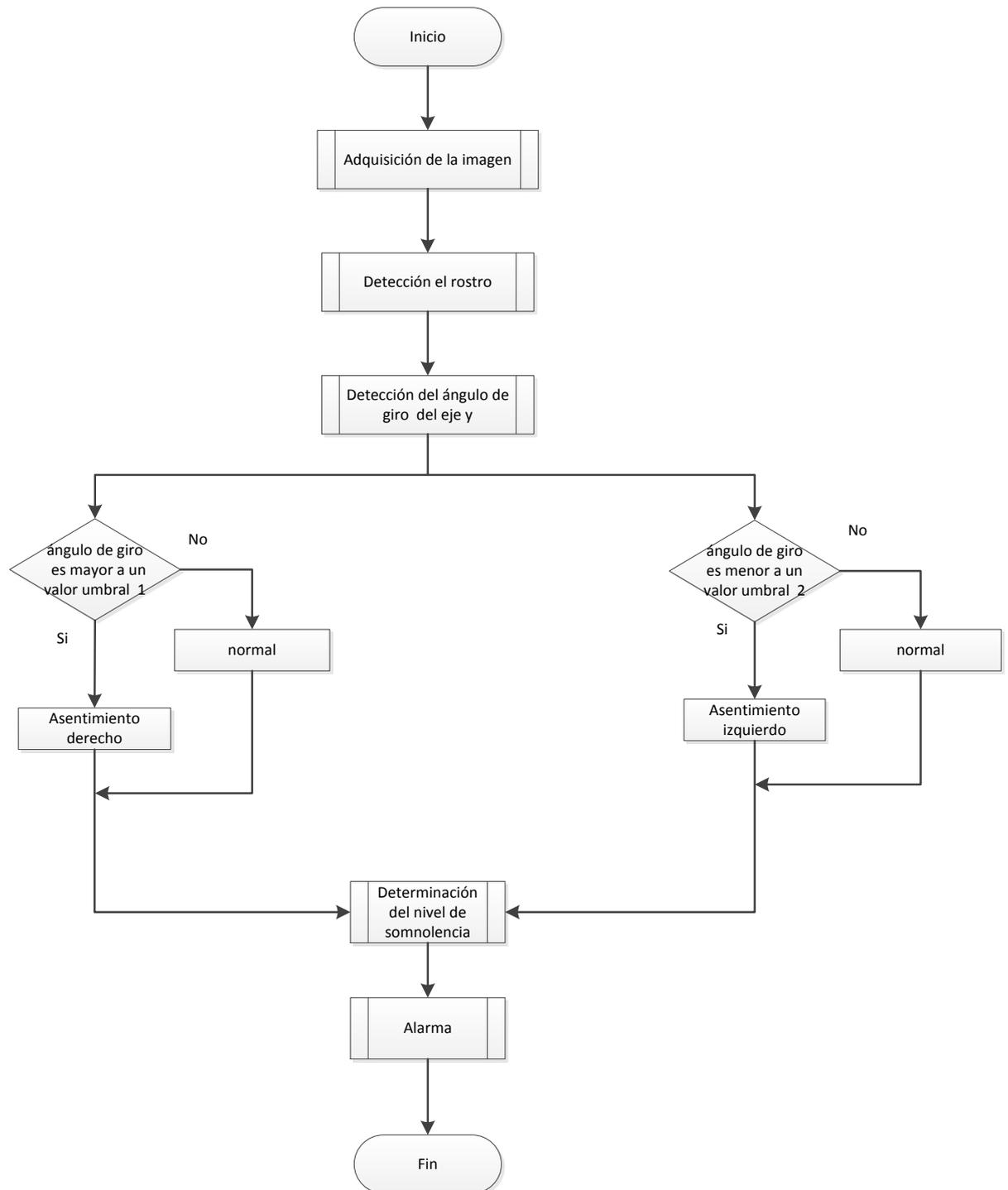


Figura 13 Diagrama de flujo para el algoritmo de detección de asentimiento lateral de la cabeza

Una vez detectado el rostro en la imagen adquirida por el sistema, mediante técnicas de visión artificial se calcula el ángulo de giro de asentimiento lateral de la cabeza y se lo compara con un valor umbral, si este es mayor a

un umbral 1 se determina como asentimiento lateral derecho y si es menor a un umbral 2 se determina como asentimiento lateral izquierdo.

c) Expresión facial

Bostezo: el estado de bostezo puede ser detectado midiendo la apertura de la boca. Cuando una persona bosteza, la apertura de su boca aumenta una cantidad específica como se observa en la figura 14. Comparando el aumento de la apertura con un valor de referencia se puede determinar si el conductor ha bostezado.



Figura 14 Apertura de la boca en a) estado normal b) estado de bostezo

Para obtener la apertura normal se promedia la apertura de la boca del conductor en 50 imágenes consecutivas y para obtener el valor de la apertura en estado de bostezo se utiliza la ecuación 1,2 y 3 planteadas en (Bin, Khan, & Abdullah, 2013) donde la cantidad que aumenta es un tercio más que el valor de la apertura de la boca en estado normal.

$$\text{valor umbral} = \text{Apertura boca prom de 50 imagenes} + \frac{\text{Apertura boca prom de 50 imagenes}}{3} \quad (1)$$

Las siguientes condiciones definidas por las ecuaciones (2) y (3) permiten determinar la presencia de bostezo utilizando el valor umbral calculado mediante la ecuación (1).

$$(\text{apertura actual boca} > \text{valor umbral}) \rightarrow \text{bostezo} \quad (2)$$

$$(\text{apertura actual boca} < \text{valor umbral}) \rightarrow \text{apertura normal} \quad (3)$$

Diagrama de flujo del algoritmo para la detección de bostezo

En la figura 15 se muestra el diagrama de flujo del algoritmo para la detección de bostezo

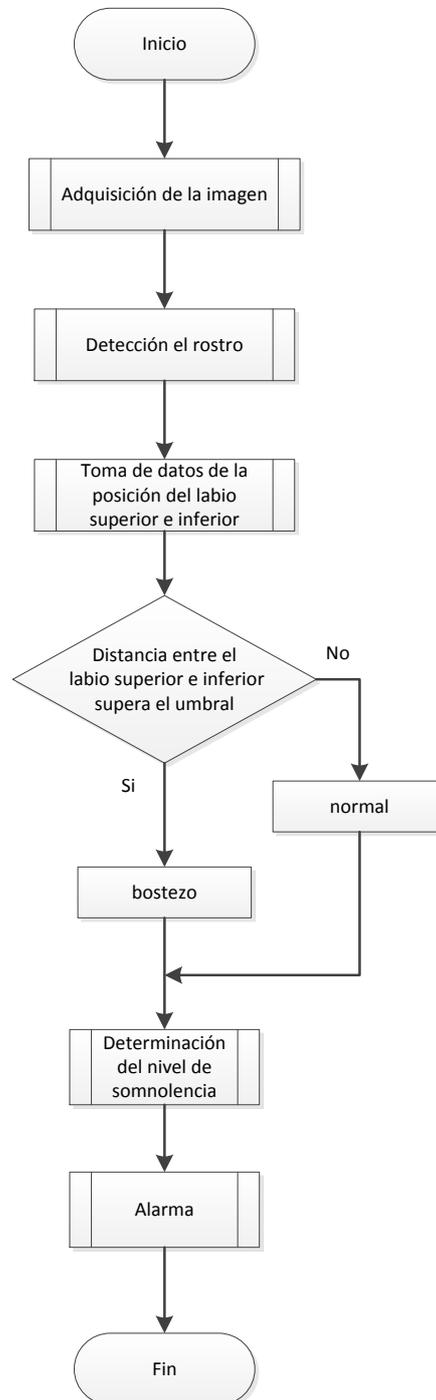


Figura 15 Diagrama de flujo del algoritmo para la detección de bostezo

Una vez detectado el rostro en la imagen adquirida por el sistema, se establece y se toma la posición de los puntos del labio superior e inferior y

se procede a calcular la distancia entre estos dos puntos, si supera un valor umbral se determina como bostezo.

d) Análisis ocular

PERCLOS: Perclos es una medida usada para la detección de somnolencia que fue establecida en 1994 por Wierwille. En un inicio fue definida como la proporción de tiempo en un minuto que el ojo está al menos el 80% cerrado. En la figura 16 se observa el comportamiento cuando el ojo es cerrado, la distancia entre el parpado superior e inferior mientras el ojo está abierto representa el 0% figura 16 a), mientras que la distancia cuando se encuentra cerrado representa el 100% figura 16 d). En la actualidad existen dos variantes de PERCLOS en uso (Svensson):

- **P70:** la proporción de tiempo que el ojo se encuentra cerrado al menos el 70% figura 16 b).
- **P80:** la proporción de tiempo que el ojo se encuentra cerrado al menos el 80% figura 16 c).

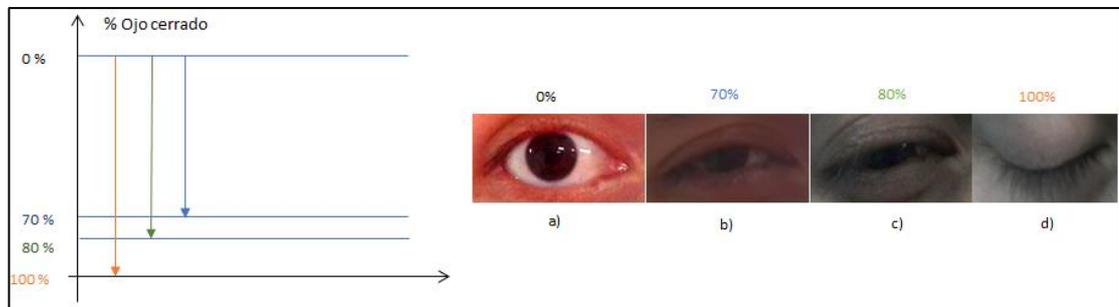


Figura 16 Porcentaje del cierre del ojo PERCLOS a) ojo abierto b) ojo cerrado al 70% c) ojo cerrado al 80% d) ojo cerrado

PERCLOS se determina mediante la ecuación 2 y su porcentaje mediante la ecuación (5):

$$\text{PERCLOS} = \frac{\text{tiempo ojos cerrados}}{\text{proporción de tiempo}} \quad (4)$$

$$\% \text{PERCLOS} = \text{PERCLOS} * 100 \quad (5)$$

Microsueño: El microsueño es un episodio temporal no intencional de pérdida de atención asociado con eventos como el asentimiento de la

cabeza y el cierre prolongado de los párpados que puede ocurrir cuando una persona esta fatigada pero tratando de permanecer despierta para realizar una tarea monótona. Este cierre prolongado de los parpados puede durar por lo general entre una fracción de segundo por lo general 0,4 hasta 30 segundos (Gregoire, 2016). Para detectar el microsueño se mide el tiempo que los ojos están cerrados y si sobrepasa un valor umbral se considera como microsueño.

Diagrama de flujo del algoritmo para la detección del estado de los ojos:

En la figura 17 se muestra el diagrama del algoritmo para la detección del estado de los ojos.

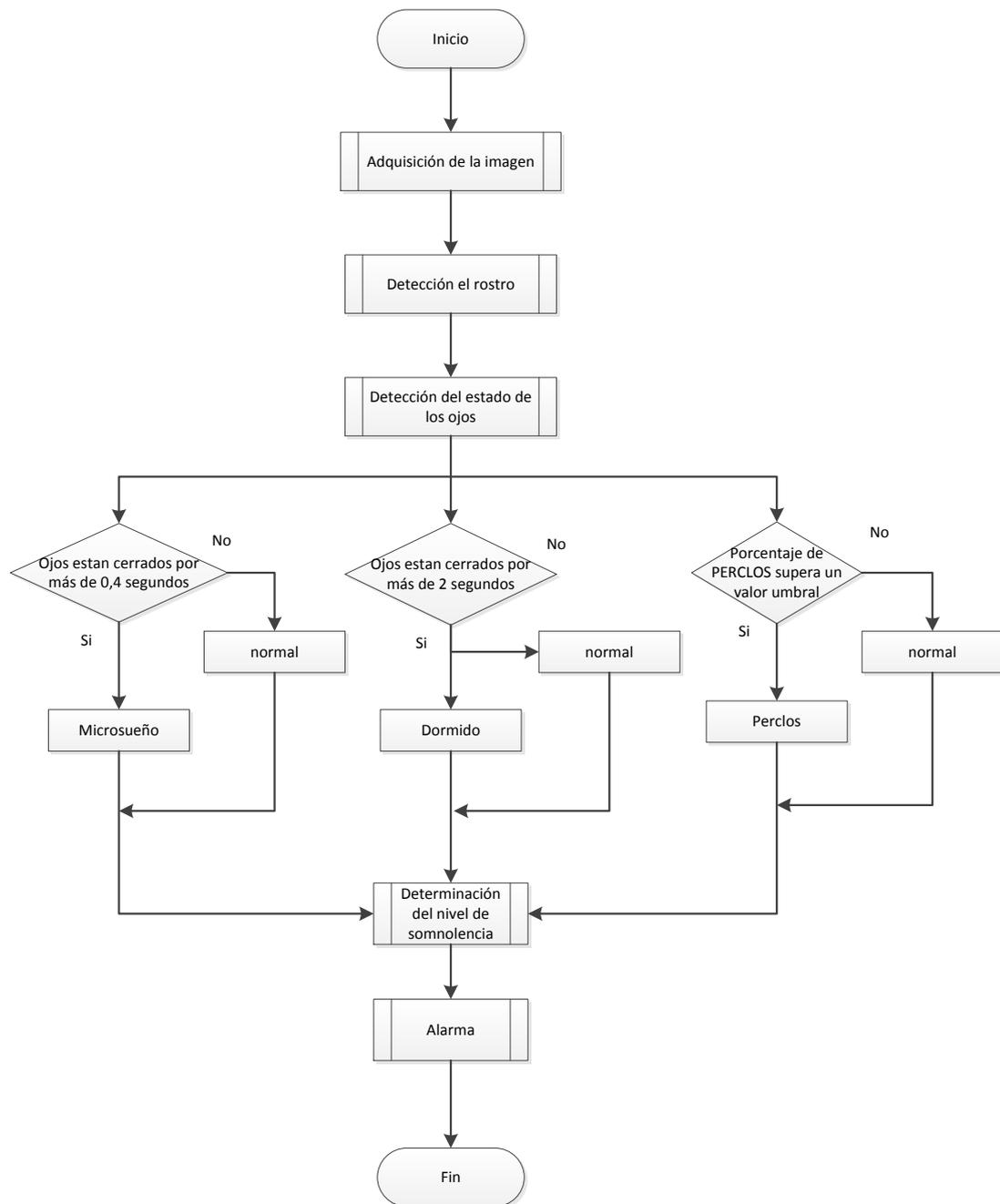


Figura 17 Diagrama del algoritmo para la detección del estado de los ojos

Una vez detectado el rostro en la imagen adquirida por el sistema, mediante técnicas de visión artificial se detecta si los ojos se encuentran cerrados y se mide el tiempo que permanecen en este estado; con este valor se calcula los siguientes parámetros:

- **PERCLOS:** Para determinar PERCLOS como indicador de somnolencia se mide el tiempo que los ojos se encuentran cerrados

en un intervalo de tiempo y se calcula este valor en porcentaje, si supera un umbral.

- **Microsueño:** si el tiempo que los ojos permanecen cerrados supera los 0,4 segundos se determina como microsueño.
- **Dormido:** si supera los 2 segundos se determina que el conductor se ha quedado dormido.

e) Niveles de somnolencia de acuerdo a indicadores visuales

Para establecer un nivel sobre el estado del conductor se consideraron los indicadores visuales que varían en el proceso de transición de despierto a somnoliento, en la tabla 8 se observa estos niveles y sus respectivos indicadores:

Tabla 7

Niveles de somnolencia de acuerdo a indicadores visuales

Nivel	Indicadores visuales
Despierto	No presencia de indicadores de somnolencia.
Somnolencia ligera	Presencia de asentimientos de la cabeza, distracción o bostezo.
Somnolencia media	Perclos80 o microsueño.
Somnolencia alta	Cierre de los ojos mayor a 2 segundos.

3.3. IMPLEMENTACIÓN

Para la elaboración de la aplicación se utiliza el lenguaje de programación java, ya que este es el lenguaje nativo del sistema operativo Android y el que usa la librería Google Mobile Api Visión, además que posee gran cantidad de soporte técnico disponible en la web.

3.3.1. Configuración de la Librería Google Mobile Api Visión

Para importar la librería Google Mobile Api Visión en nuestra aplicación es necesario añadir Google Play Services SDK que es su contenedor, como

una dependencia en el archivo de módulo de la aplicación "build.gradle". Mediante la siguiente línea de código se indica que se va a importar la versión 9.4.0 en adelante.

compile 'com.google.android.gms:play-services-vision:9.4.0+'

La ubicación de archivo build.gradle se muestra en la figura 18:

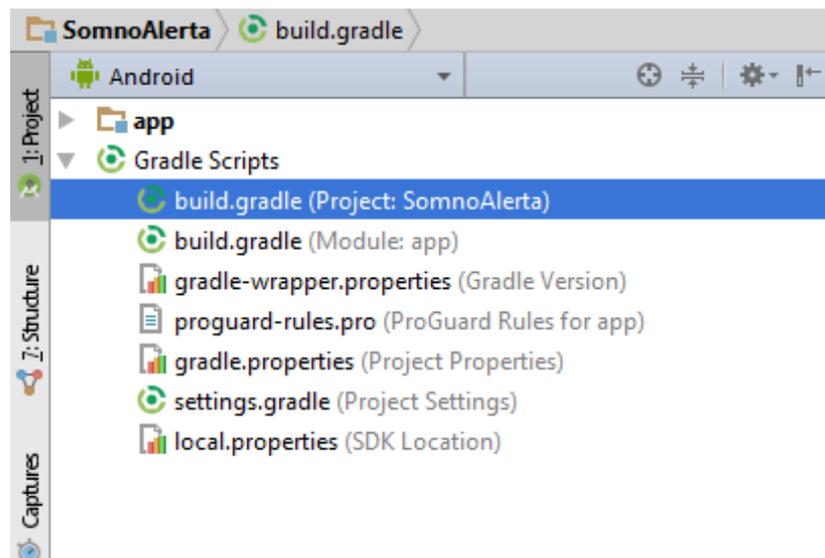


Figura 18 Ubicación del archivo build.gradle

Una vez importada la librería se debe habilitar para que esté disponible la detección del rostro, para ello añadimos el siguiente código en el archivo AndroidManifest.xml:

<meta-data

android:name="com.google.android.gms.vision.DEPENDENCIES"

android:value="face"/>

Las líneas de código anteriores permiten habilitar los recursos necesarios utilizados por la librería para la detección de rostros y detección de características faciales en la imagen.

La ubicación del archivo AndroidManifest.xml se muestra en la figura 19.

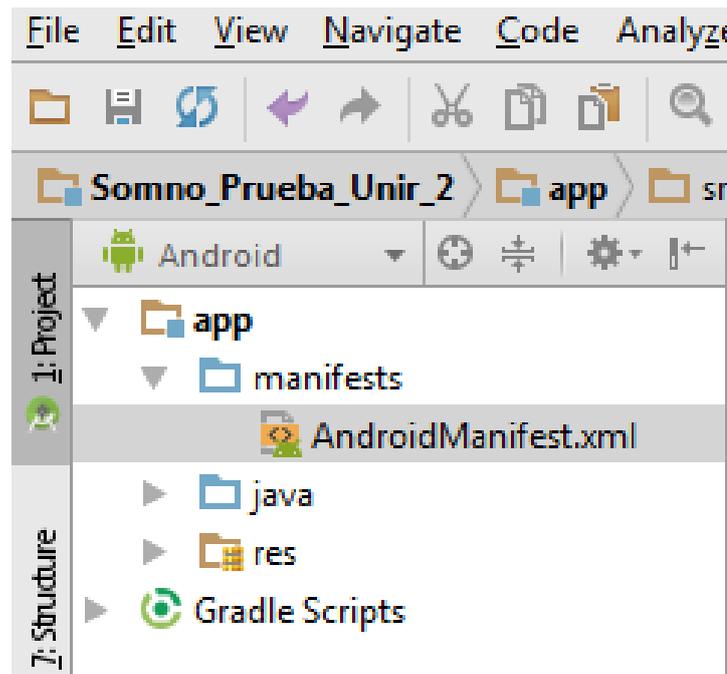


Figura 19 Ubicación del archivo AndroidManifest.xml

3.3.2. Creación de la Interface de usuario UI

El sistema cuenta con tres pantallas que son:

Pantalla de inicio: La pantalla de inicio que se muestra en la figura 20 está compuesta por los siguientes componentes:

- **Vista de texto 1:** se utiliza para visualizar el nombre de la Universidad
- **Vista de texto 2:** se utiliza para visualizar el título del sistema
- **Vista de texto 3:** se utiliza para visualizar el autor y tutor del sistema
- **Botón “DETECTAR SOMNOLENCIA”** : se utiliza para iniciar la detección del sistema
- **Botón “CERRAR”:** se utiliza para cerrar el sistema

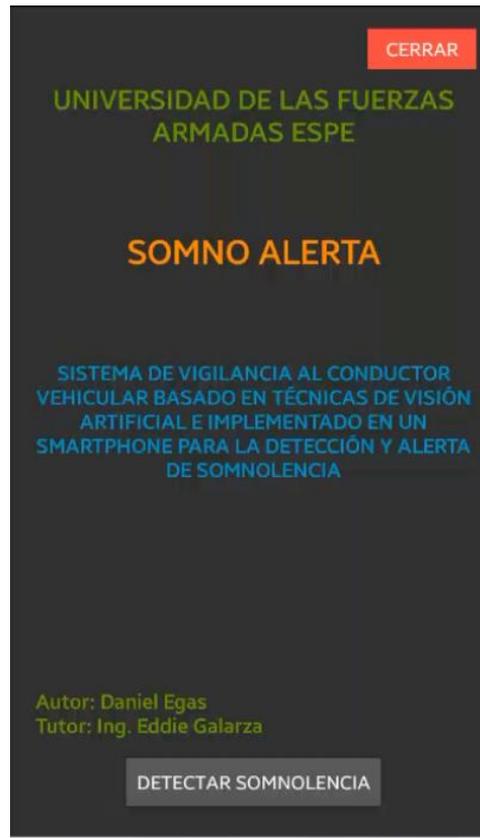


Figura 20 Pantalla de inicio

Pantalla de detección de indicadores visuales: La pantalla de indicadores visuales que se muestra en la figura 21 está compuesta por los siguientes componentes:

- **CameraSourcePreview:** se utiliza para mostrar la imagen adquirida por la cámara
- **GraphicOverlay:** se utiliza para sobreponer etiquetas sobre la imagen. Para mostrar los datos del sistema se crearon las siguientes:
- **Etiqueta 1:** mediante este componente se visualiza el estado del conductor
- **Etiqueta 2:** para la interpretación completa de los indicadores visuales se creó un cuadro de texto, el cual es de gran ayuda para comprobar que el sistema está trabajando de manera correcta.
- **Botón “OCULTAR”:** se utiliza para ocultar la imagen
- **Botón “CERRAR”:** se utiliza para cerrar el sistema
- **Slider :** se utiliza para contralar el brillo de la ventana

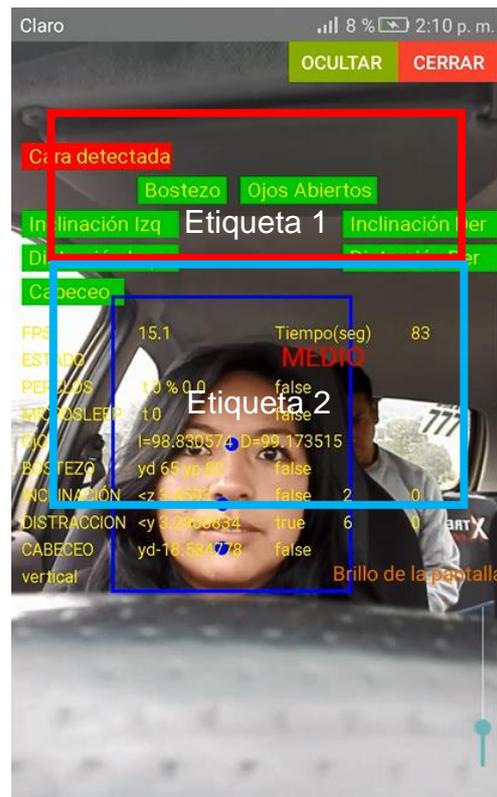


Figura 21 Pantalla de detección de indicadores visuales

Pantalla de alerta para el nivel “conductor dormido”: La pantalla de alerta para el nivel “conductor dormido” que se muestra en la figura 22 está compuesta por los siguientes componentes:

- **Vista de texto 1:** se utiliza para mostrar la alarma sobre el estado del conductor
- **Botón “CONTINUAR”:** se utiliza para desactivar la alarma especial que se genera cuando el conductor se ha quedado dormido, de esta manera se asegura de que el conductor se despierte caso contrario la alarma seguirá sonando de manera indefinida.

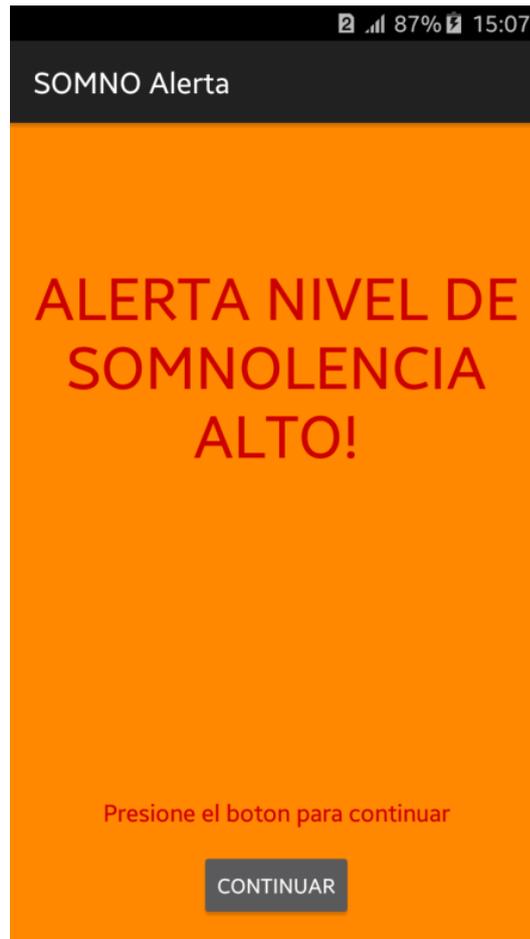


Figura 22 Pantalla de alerta para alarma de conductor dormido

3.3.3. Adquisición de la imagen

Para la adquisición de la imagen del conductor se usó la clase **mCameraSouce**, su función es gestionar la cámara del smartphone en combinación a un detector de rostro asociado. Recibe las imágenes de vista previa de la cámara y las envía tan rápido como sea posible al detector.

La imagen es capturada mediante la cámara frontal a una velocidad de 30 imágenes por segundo y tiene un tamaño de 320x240 pixeles.

En la figura 23 se observa una imagen adquirida por el sistema:



Figura 23 Imagen adquirida por la cámara frontal del smartphone

3.3.4. Detección del rostro

La detección del rostro se realizó mediante la clase detector (), la cual recibe una imagen como entrada y como salida nos entrega la posición de la cara.

Resultados de la detección de rostros: Cuando el rostro es detectado se crea un rectángulo sobre este y una etiqueta de color rojo con el texto “Cara Detectada” se muestra como se observa en la figura 24.



Figura 24 Detección del rostro

3.3.5. Detección del estado de los ojos

La detección del estado de los ojos se realizó mediante la clase **getIsLeftEyeOpenProbability()**, la cual mediante un procesamiento digital en la región del rostro analiza si los ojos se encuentran abiertos o cerrados.

Resultados de la detección del estado de los ojos: Cuando los ojos detectados se encuentran cerrados, una etiqueta de color verde se muestra en la pantalla con el color de la etiqueta texto “Ojos Abiertos” (figura 25), si los ojos se encuentran cerrados el color de la etiqueta cambia a rojo y el texto a “Ojos Cerrados” figura 26.

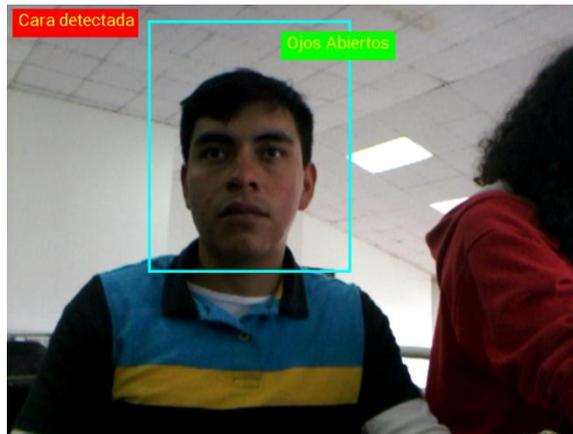


Figura 25 Detección de los ojos abiertos



Figura 26 Detección de los ojos cerrados

3.3.6. Detección de distracción

El proceso de distracción se realizó mediante la clase **getEulerY()**. Esta clase realiza un procesamiento sobre el rostro del conductor para determinar el ángulo que gira la cabeza alrededor del eje Y, el rango del ángulo de giro va desde los 90 grados cuando se encuentra la cabeza girada totalmente hacia la derecha y -90 grados cuando se encuentra la cabeza girada totalmente hacia la izquierda.

Resultados de la detección de distracción: Los resultados del algoritmo implementado para la detección de distracción se observan en la figura 27, donde el foco de la mirada del conductor no se encuentra en el frente sino hacia el lado derecho o izquierdo (figura 28).

Cuando el foco de la mirada se encuentra al lado derecho o izquierdo, el color de la etiqueta “Distracción Der” y “Distracción Izq” cambian de color verde a rojo respectivamente.

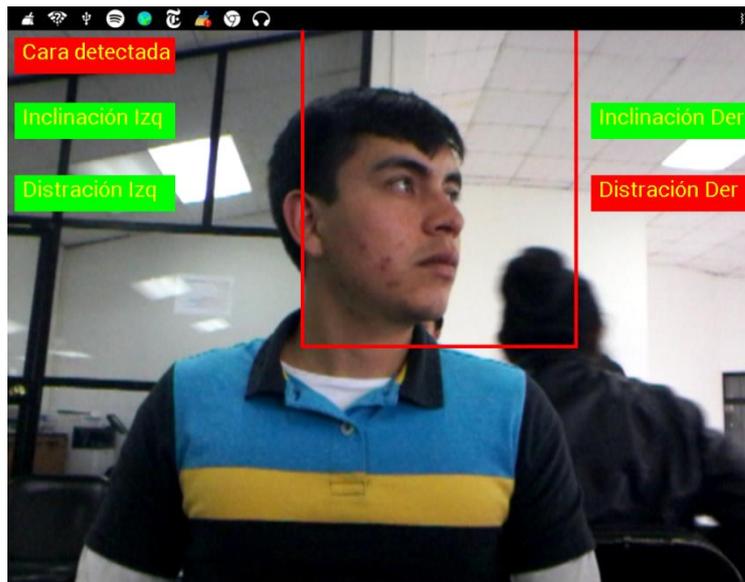


Figura 27 Detección del rostro del conductor mirando hacia la derecha

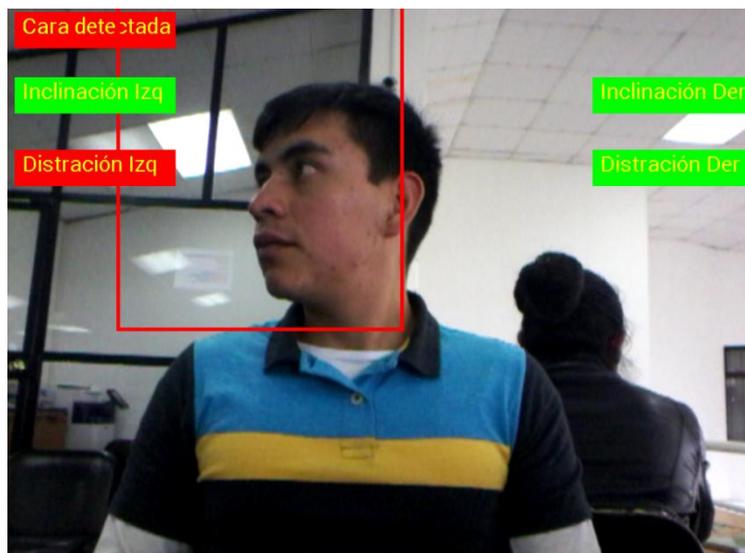


Figura 28 Detección del rostro del conductor mirando hacia la izquierda

3.3.7. Detección de bostezo

Para la detección de bostezo, se implementó el algoritmo analizado en la sección anterior, el cual establece la presencia de un bostezo analizando la distancia entre dos puntos característicos de la boca como son el labio superior y el inferior, tal como se observa en la figura 29.

Resultados de la detección de bostezo: Cuando un bostezo es detectado, la etiqueta de color verde y de texto “Bostezo” cambia a color rojo como se observa en la figura 30.

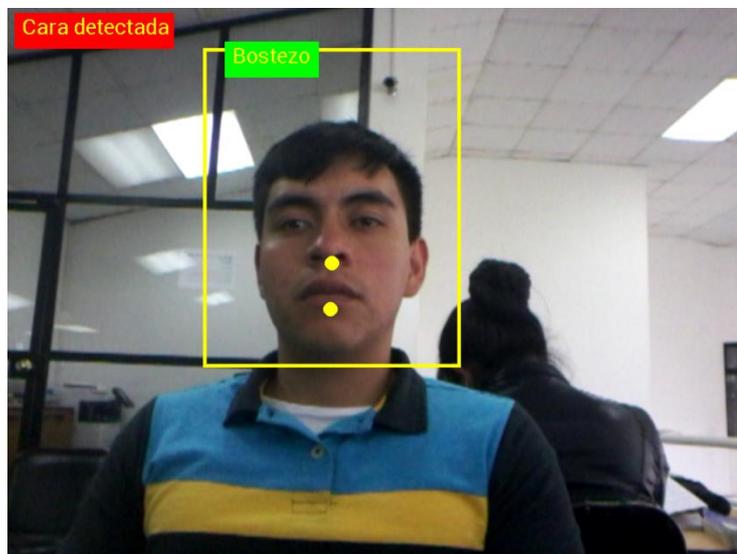


Figura 29 Detección de la boca cuando se encuentra normal

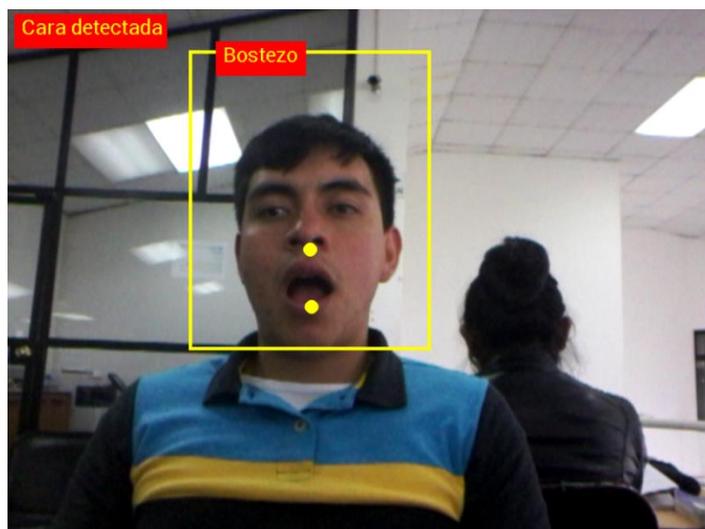


Figura 30 Detección de la boca cuando se encuentra bostezando

3.3.8. Detección del asentimiento frontal de la cabeza

Resultados de la detección del asentimiento frontal de la cabeza:
Cuando un asentimiento frontal es detectado la etiqueta de color verde y de texto “Cabeceo” cambia a color rojo como se observa en la figura 32.



Figura 31 Detección del asentimiento frontal de la cabeza



Figura 32 Detección del rostro cuando se encuentra asintiendo de manera frontal

CAPÍTULO IV

4. PRUEBAS Y RESULTADOS EXPERIMENTALES

En esta sección, probaremos el sistema bajo condiciones reales de conducción, además se presentan los resultados sobre la detección de los indicadores visuales de somnolencia.

4.1. Metodología para las pruebas

La recopilación del conjunto de datos para evaluar adecuadamente el sistema es un reto. Esto se debe a que los peligrosos eventos de somnolencia no están garantizados que se produzcan durante la conducción diaria para los ensayos a pequeña escala de la aplicación. Por el contrario, los fabricantes de autos que desarrollan sistemas de vigilancia al conductor se esfuerzan en probar sus productos en ambientes lo más parecidos a los reales, por ejemplo, los científicos que trabajan para la marca Mercedes - Benz (Taylor, 2008) disponen de coches especializados para evaluar, donde los participantes de estos estudios son físicamente fatigados antes del inicio del experimento o simulación, consiguiendo que aumente la probabilidad de que se presenten estos eventos.

Tal enfoque va más allá de nuestros recursos. La recolección de estos datos no es viable porque no podemos acumular suficientes ejemplos de conducción somnolienta para evaluar completamente el sistema. Además, sería irresponsable realizar este experimento sin tomar el tipo de medidas preventivas que toman los fabricantes de automóviles. Por estas razones evaluamos el sistema utilizando un conjunto de datos creado a partir de eventos de somnolencia controlados. Para ello se van a utilizar 20 conductores, 10 hombres y 10 mujeres de distintas edades, donde cada uno es acompañado por un “copiloto” que dirige los eventos controlados de somnolencia solo cuando las condiciones externas en la carretera y alrededor del vehículo son seguras (por ejemplo, el copiloto ordena al conductor que cierre por unos segundos los ojos cuando la carretera se

preste para un movimiento rectilíneo, no existe presencia de vehículos en un amplio tramo y no se visualice peatones).

Los eventos para validar el sistema son los siguientes:

- Bostezo
- Asentimiento frontal y lateral de la cabeza
- Distracción izquierda y derecha
- Parpadeo

Para validar el sistema, el copiloto dirigirá estos eventos, haciendo que se repitan 10 veces. Se analizará el número de eventos detectados correctamente, los falsos positivos, falsos negativos y se determinará la eficiencia del sistema implementado.

4.1.1. Ubicación correcta del dispositivo

La ubicación del dispositivo para probar el sistema se observa en la figura 33, en donde se encuentra ubicado cumpliendo los requisitos mencionados en el capítulo 3.



Figura 33 Posición del smartphone en el vehículo

4.2. Pruebas de funcionamiento del sistema de detección de somnolencia

Las siguientes pruebas fueron realizadas bajo las condiciones establecidas en el capítulo 3, con la intervención de 20 conductores de vehículos entre ellas 10 hombres y 10 mujeres; al inicio de cada sesión de pruebas con el conductor, se le hizo una introducción explicándole los movimientos que debía reproducir, la mayoría habían experimentado de alguna manera estos movimientos como son bostezos, asentimiento de la cabeza frontal y lateral, distracción y parpadeo, sin embargo no eran conscientes de cómo realizarlos a voluntad propia, para ello fue necesario demostrarle visualmente como debían realizarlos.

4.2.1. Pruebas de funcionamiento de la detección de bostezos bajo condiciones establecidas.

En la tabla 8 se presentan los resultados para esta primera prueba.

Tabla 8

Porcentaje de detección calculado con 200 muestras pertenecientes a 20 conductores bajo condiciones establecidas

PORCENTAJE DE DETECCIÓN DE BOSTEZOS		
CONDUCTOR	NÚMERO DE ACIERTOS	% DE RECONOCIMIENTO
1	10	100
2	8	80
3	6	60
4	8	80
5	8	80
6	10	100
7	9	90
8	8	80
9	5	50
10	10	100

CONTINÚA 

11	10	100
12	8	80
13	9	90
14	7	70
15	9	90
16	10	100
17	8	80
18	8	80
19	5	50
20	10	100
% TOTAL DE DETECCIÓN		83

La tabla 8 muestra el porcentaje total de detección de bostezos, el resultado indica que el sistema posee un 83 % de acierto en la detección.

4.2.2. Pruebas de funcionamiento de la detección de asentimiento frontal de la cabeza bajo condiciones establecidas.

En la tabla 9 se presentan los resultados para esta prueba.

Tabla 9

Porcentaje de detección calculado con 200 muestras pertenecientes a 20 conductores bajo condiciones establecidas

PORCENTAJE DE DETECCIÓN DE ASENTIMIENTO FRONTAL DE LA CABEZA		
CONDUCTOR	NÚMERO DE ACIERTOS	% DE RECONOCIMIENTO
1	9	90
2	10	100
3	9	90
4	10	100
5	10	100
6	9	90
7	9	90

CONTINÚA 

8	9	90
9	9	90
10	10	100
11	8	80
12	10	100
13	10	100
14	10	100
15	10	100
16	10	100
17	10	100
18	6	60
19	10	100
20	6	60
% TOTAL DE DETECCIÓN		92

La tabla 9 muestra el porcentaje total de detección de asentimiento frontal de la cabeza, el resultado indica que el sistema posee un 92 % de acierto en la detección.

4.2.3. Pruebas de funcionamiento de la detección de asentimiento derecho de la cabeza bajo condiciones establecidas.

En la tabla 10 se presentan los resultados para esta prueba.

Tabla 10

Porcentaje de detección calculado con 200 muestras pertenecientes a 20 conductores bajo condiciones establecidas

PORCENTAJE DE DETECCIÓN DE ASENTIMIENTO DERECHO DE LA CABEZA		
CONDUCTOR	NÚMERO DE ACIERTOS	% DE RECONOCIMIENTO
1	10	100
2	10	100
3	9	90

CONTINÚA 

4	10	100
5	9	90
6	10	100
7	8	80
8	9	90
9	10	100
10	9	90
11	10	100
12	10	100
13	9	90
14	9	90
15	10	100
16	10	100
17	10	100
18	10	100
19	10	100
20	8	80
% TOTAL DE DETECCIÓN		95

La tabla 10 muestra el porcentaje total de detección de asentimiento derecho de la cabeza, el resultado indica que el sistema posee un 95 % de acierto en la detección.

4.2.4. Pruebas de funcionamiento de la detección de asentimiento izquierdo de la cabeza bajo condiciones establecidas.

En la tabla 11 se presentan los resultados para esta prueba.

Tabla 11

Porcentaje de detección calculado con 200 muestras pertenecientes a 20 conductores bajo condiciones establecidas

PORCENTAJE DE DETECCIÓN DE ASENTIMIENTO IZQUERDO DE LA CABEZA		
CONDUCTOR	NÚMERO DE ACIERTOS	% DE RECONOCIMIENTO
1	9	90
2	9	90
3	9	90
4	10	100
5	8	80
6	10	100
7	10	100
8	8	80
9	10	100
10	9	90
11	10	100
12	10	100
13	10	100
14	6	60
15	9	90
16	10	100
17	10	100
18	9	90
19	8	80
20	7	70
% TOTAL DE DETECCIÓN		90,5

La tabla 11 muestra el porcentaje total de detección de asentimiento izquierdo de la cabeza, el resultado indica que el sistema posee un 90,5 % de acierto en la detección.

4.2.5. Pruebas de funcionamiento de la detección de distracción derecha bajo condiciones establecidas.

En la tabla 12 se presentan los resultados para esta prueba.

Tabla 12

Porcentaje de detección calculado con 200 muestras pertenecientes a 20 conductores bajo condiciones establecidas

PORCENTAJE DE DETECCIÓN DE LA DISTRACCIÓN DERECHA		
CONDUCTOR	NÚMERO DE ACIERTOS	% DE RECONOCIMIENTO
1	10	100
2	9	90
3	8	80
4	7	70
5	10	100
6	8	80
7	8	80
8	8	80
9	9	90
10	10	100
11	10	100
12	9	90
13	10	100
14	9	90
15	10	100
16	10	100
17	10	100
18	10	100
19	9	90
20	10	100
% TOTAL DE DETECCIÓN		92

La tabla 12 muestra el porcentaje total de la detección de la distracción derecha, el resultado indica que el sistema posee un 92 % de acierto en la detección.

4.2.6. Pruebas de funcionamiento de la distracción izquierda bajo condiciones establecidas.

En la tabla 13 se presentan los resultados para esta prueba.

Tabla 13

Porcentaje de detección calculado con 200 muestras pertenecientes a 20 conductores bajo condiciones establecidas

PORCENTAJE DE DETECCIÓN DE LA DISTRACCIÓN IZQUERDA		
CONDUCTOR	NÚMERO DE ACIERTOS	% DE RECONOCIMIENTO
1	10	100
2	9	90
3	10	100
4	9	90
5	10	100
6	10	100
7	8	80
8	9	90
9	10	100
10	10	100
11	10	100
12	8	80
13	10	100
14	10	100
15	10	100
16	10	100
17	10	100
18	10	100

CONTINÚA 

19	10	100
20	10	100
% TOTAL DE DETECCIÓN		96,5

La tabla 13 muestra el porcentaje total de detección de la distracción izquierda, el resultado indica que el sistema posee un 96,5 % de acierto en la detección.

4.2.7. Pruebas de funcionamiento de la detección de parpadeo bajo condiciones establecidas.

En la tabla 14 se presentan los resultados para esta prueba.

Tabla 14

Porcentaje de detección de parpadeo calculado con 200 muestras pertenecientes a 20 conductores bajo condiciones establecidas

PORCENTAJE DE DETECCIÓN DE PARPADEO		
CONDUCTOR	NÚMERO DE ACIERTOS	% DE RECONOCIMIENTO
1	9	90
2	10	100
3	10	100
4	10	100
5	10	100
6	10	100
7	10	100
8	10	100
9	10	100
10	9	90
11	10	100
12	10	100
13	10	100
14	10	100

CONTINÚA 

15	9	90
16	10	100
17	10	100
18	10	100
19	10	100
20	10	100
% TOTAL DE DETECCIÓN		98,5

La tabla 14 muestra el porcentaje total de detección de parpadeo, el resultado indica que el sistema posee un 98,5 % de acierto.

4.2.8. Precisión total del nivel de somnolencia

Para determinar el porcentaje de precisión total del sistema se promedia los porcentajes de precisión de los eventos individuales, el resultado se muestra en la tabla 15.

Tabla 15

Porcentaje de detección total del sistema

EVENTOS	NÚMERO DE ACIERTOS
BOSTEZO	83
ASENTIMIENTO FRONTAL	92
ASENTIMIENTO DERECHO	95
ASENTIMIENTO IZQUIERDO	90,5
DISTRACCIÓN DERECHO	92
DISTRACCIÓN IZQUIERDO	96,5
PARAPDEO	98,5
% TOTAL DE DETECCIÓN	92,5

Se obtuvo una precisión total del sistema del 92,5 % para detectar los indicadores de somnolencia.

4.3.2. Activación del nivel de alarma medio

En la figura 35 se muestra la activación del nivel de alarma medio la que se realizó con éxito.

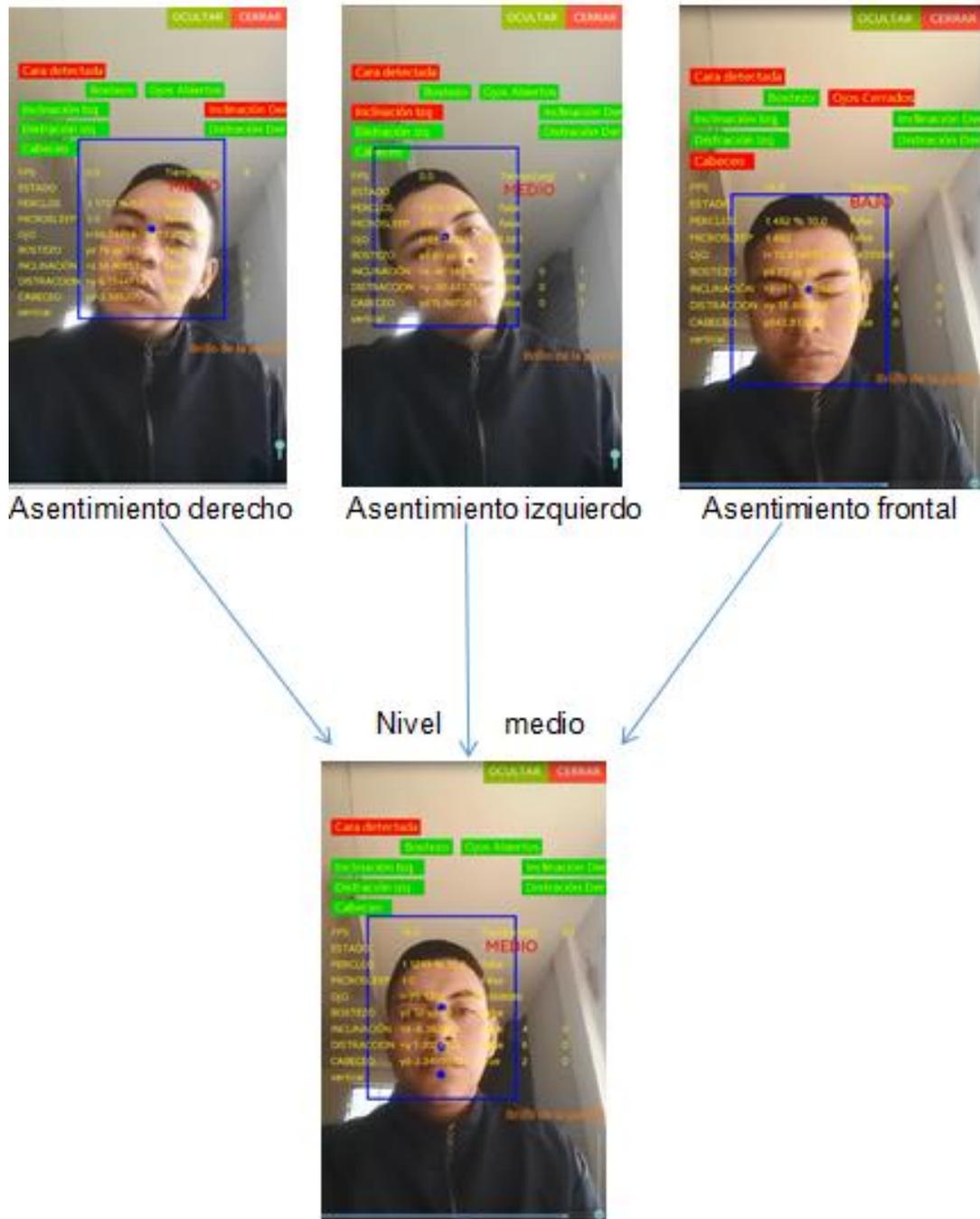


Figura 35 secuencia activación del nivel de alarma medio

4.3.3. Activación del nivel de alarma alto

En la figura 36 se muestra la activación del nivel de alarma alto la que se realizó con éxito.

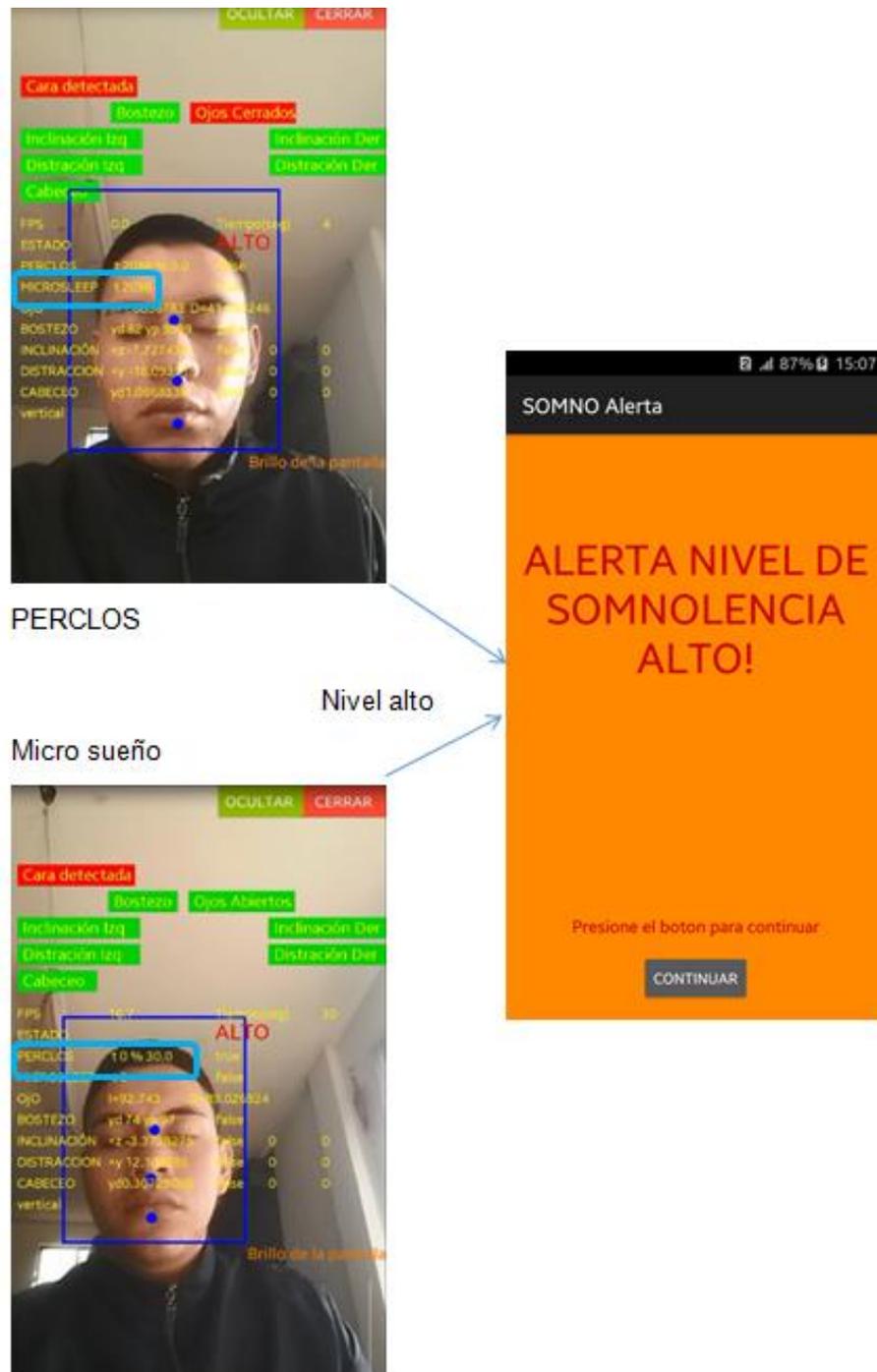


Figura 36 secuencia de activación del nivel de alarma alto

4.3.4. Análisis general

El sistema denominado “sistema de vigilancia al conductor vehicular basado en técnicas de visión artificial e implementado en un smartphone para la detección y alerta de somnolencia” tiene una precisión promedio de 92,5%.

En el momento de realizar las pruebas de cabeceo, el sistema detectó erróneamente bostezos, esto se debe a que tanto el algoritmo implementado para bostezo y asentimiento frontal usan puntos de referencia del rostro, como son el labio superior e inferior en el caso de bostezo y el centro de los dos ojos en el caso del asentimiento frontal. Al momento de reproducir un movimiento de asentimiento, la cabeza se desplaza hacia abajo impidiendo que se visualice correctamente el labio inferior y en algunos casos perdiéndolo de vista, esto hace que se tomen valores erróneos de su posición y se active el estado de bostezo detectado. Esto no representa un problema tomando en cuenta la prioridad y el orden de aparición de los eventos, es decir, es más importante detectar un cabeceo a un bostezo ya que este se encuentra en el nivel 2 y el bostezo en el nivel 1, tomando como consideración este hecho se optó por implementar un bloqueo que evita que se detecten bostezos falsos después de un cierto periodo de tiempo de haberse producido un asentimiento frontal.

En lo referente a la detección de asentimiento de la cabeza lateral, se observó que este es un movimiento que depende en gran medida de la flexibilidad de las personas, conductores de edad adulta como se muestra en la figura 37 a), no giran la cabeza en la medida como los de edad media figura 37 b) y baja figura 37 c), esto provoca que el umbral que se utiliza para activar la alarma no sea el suficiente para las personas cuya flexibilidad se ha visto disminuida con el pasar de los años.

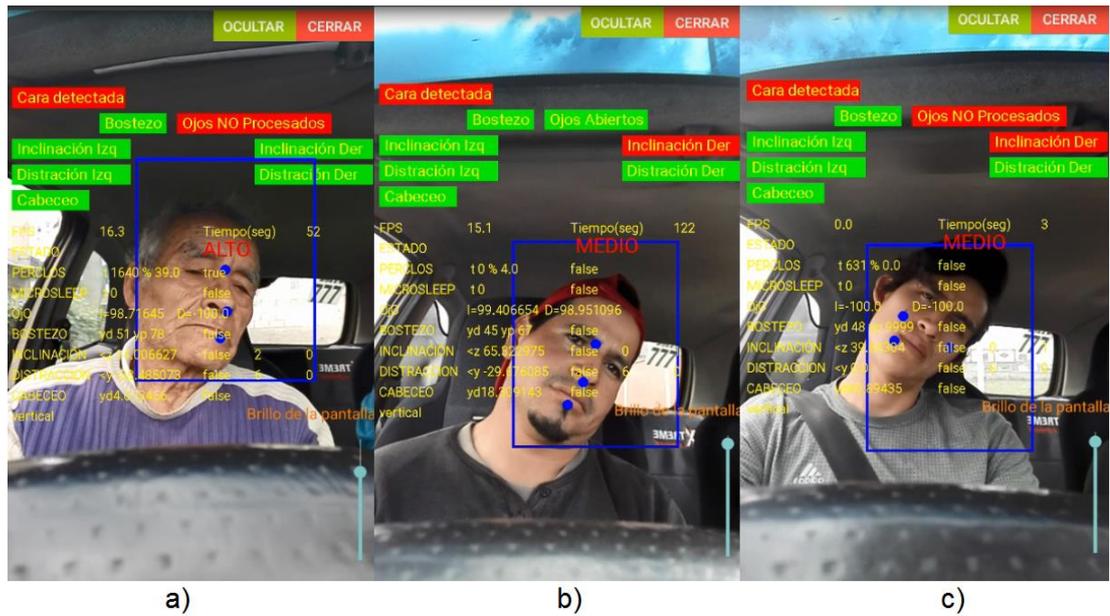


Figura 37 Flexibilidad al inclinar la cabeza en conductores de:
a) Edad adulta b) edad media c) edad baja

En cuanto a la vestimenta y apariencia se destacan los siguientes casos:

- El uso de gorra como se observa en la figura 38, no afectó el rendimiento del sistema.

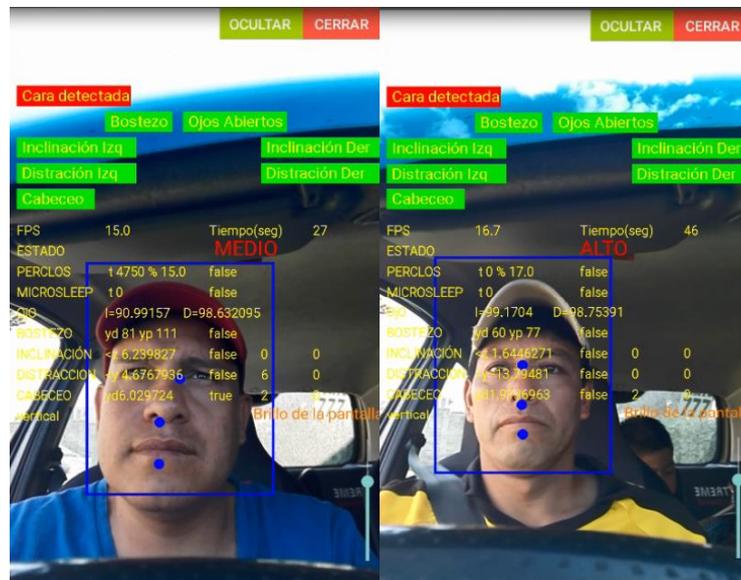


Figura 38 Conductores usando gorra

Los conductores 16 y 17 que usaron gorra presentaron valores similares a los que no usaban tal como se puede observar en la tabla 16.

Tabla 16

Resultados de las pruebas para los conductores que usaron gorra.

	Bostezo	Cabeceo	Dis. derecha	Dis. izquierda	Inc. derecha	Inc. izquierda	Parpadeo
1	10	9	10	10	10	9	9
2	8	10	9	9	10	9	10
3	6	9	8	10	9	9	10
4	8	10	7	9	10	10	10
5	8	10	10	10	9	8	10
6	10	9	8	10	10	10	10
7	9	9	8	8	8	10	10
8	8	9	8	9	9	8	10
9	5	9	9	10	10	10	10
10	10	10	10	10	9	9	9
11	10	8	10	10	10	10	10
12	8	10	9	8	10	10	10
13	9	10	10	10	9	10	10
14	7	10	9	10	9	6	10
15	9	10	10	10	10	9	9
16	10	10	10	10	10	10	10
17	8	10	10	10	10	10	10
18	8	6	10	10	10	9	10
19	5	10	9	10	10	8	10
20	10	6	10	10	8	7	10

- De los conductores que usaron lentes, el sistema detecto el 100% de los parpadeos en el caso del conductor que usaba lentes sin anti reflejante figura a), mientras que en el caso de 4 conductores que usaron lentes con anti reflejante el sistema detecto el 100% de uno de ellos figura d) y el 90% a los otros tres figura b),c) y e). Esta mínima diferencia en los aciertos se debe a la variación del reflejo en los lentes, en ocasiones este reflejo no permite visualizar de una manera correcta el ojo para que se realice una detección acertada sobre su estado.



Figura 39 Conductor con lentes a) sin anti reflejante b) c) d) e) con anti reflejante

Los resultados de las pruebas de los conductores que usaron lentes se observan en la tabla 17:

Tabla 17

Resultados de las pruebas de los conductores que usaron lentes

	Bos.	Cabeceo	Dis. derecha	Dis. izquierda	Inc. derecha	Inc. izquierda	Parpadeo
1 b)	10	9	10	10	10	9	9
2 a)	8	10	9	9	10	9	10
3	6	9	8	10	9	9	10
4	8	10	7	9	10	10	10
5 d)	8	10	10	10	9	8	10
6	10	9	8	10	10	10	10
7	9	9	8	8	8	10	10
8	8	9	8	9	9	8	10
9	5	9	9	10	10	10	10
10 e)	10	10	10	10	9	9	9
11	10	8	10	10	10	10	10
12	8	10	9	8	10	10	10
13	9	10	10	10	9	10	10
14	7	10	9	10	9	6	10
15 c)	9	10	10	10	10	9	9
16	10	10	10	10	10	10	10
17	8	10	10	10	10	10	10
18	8	6	10	10	10	9	

19	5	10	9	10	10	8	10
20	10	6	10	10	8	7	10

- El sistema se vio afectado en la detección de distracción y asentimiento lateral cuando la conductora lucía un estilo de cabello que le cubría la mayor parte de las mejillas, impidiendo que se visualice el rostro completamente al girarlo, esto causaba que no se detecte el rostro como se observa en la figura 40 a) o que se detecte el rostro pero se tenga valores erróneos de los puntos de referencia figura 40 b), los demás eventos no se vieron afectados.



Figura 40 Conductora con cabello que le cubre las mejillas: a) cara no detectada b) detección errónea de los puntos de referencia

En la tabla 18 se observa que los valores de los aciertos en la detección de bostezo, cabeceo, inclinación derecha, inclinación izquierda y parpadeo son similares en ambos casos, a diferencia de los aciertos, en la detección de distracción derecha e izquierda, en donde el sistema detectó el 0% y 10% respectivamente en el caso de la conductora con las mejillas cubiertas de cabello y un 100% para ambas distracciones en el caso de la conductora sin las mejillas cubiertas.

Tabla 18

Contraste de las pruebas de una conductora con cabello que cubre las mejillas y sin cabello que cubre las mejillas

Conductor	Bostezo	Cabeceo	Dis.		Inc.		Parpadeo
			Der.	. izq.	Der.	izq.	
mejillas descubiertas							
	8	10	10	10	9	8	10
mejillas cubiertas							
	10	10	0	1	8	10	10

4.4. Pruebas de usabilidad

Esta prueba se realizó con el fin de tener una apreciación del sistema por parte del usuario final, para ello se realizaron 5 preguntas (ver anexo 1) cuyos resultados son los siguientes:

1. ¿Considera que la alerta emitida es oportuna?

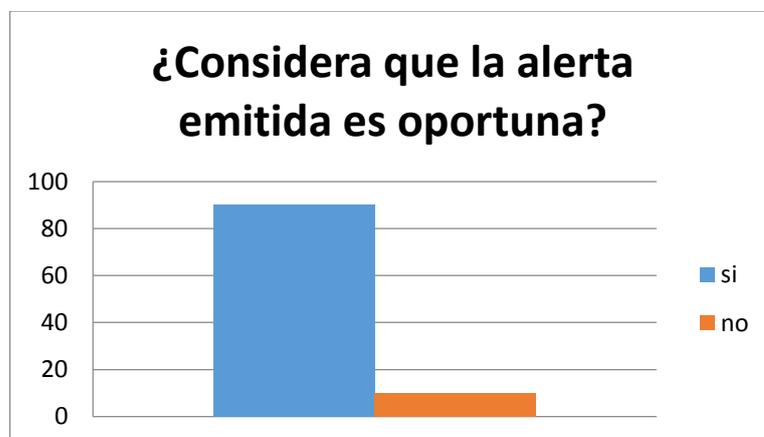


Figura 41 Resultado de la tabulación de la pregunta 1

En el gráfico 1 se puede observar que el 90 % de los conductores consideran que el tono de la alarma es oportuno.

2. ¿Considera que el sistema es fácil de usar?

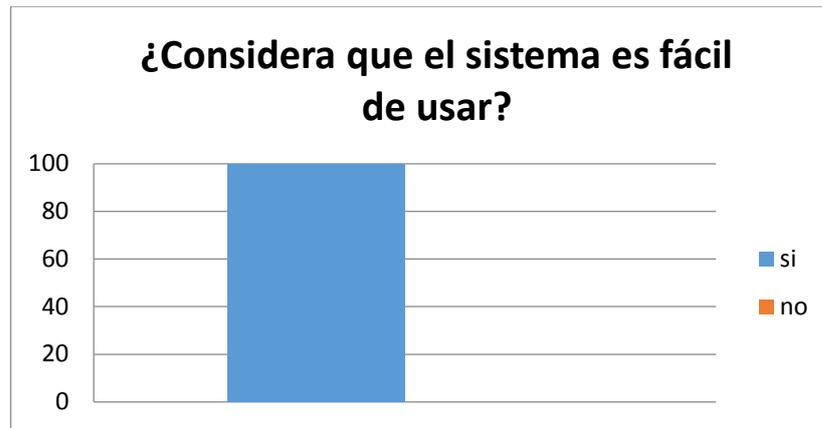


Figura 42 Resultado de la tabulación de la pregunta 2

En el gráfico 3 se puede observar que el 100 % de los conductores consideran que el sistema es fácil de usar.

3. ¿Usaría el sistema en su actividad de conducción?

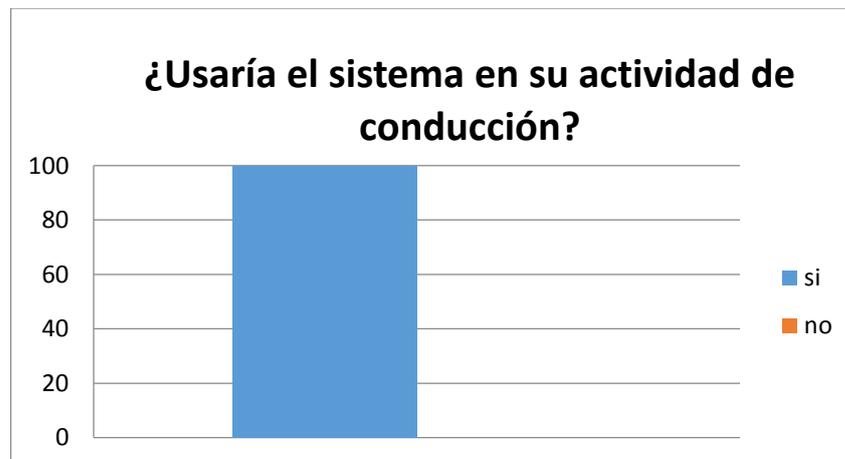


Figura 43 Resultado de la tabulación de la pregunta 3

En el gráfico 3 se puede observar que el 100 % de los conductores usarían el sistema en su actividad de conducción.

4. ¿Sugeriría a otros conductores que usen la aplicación?

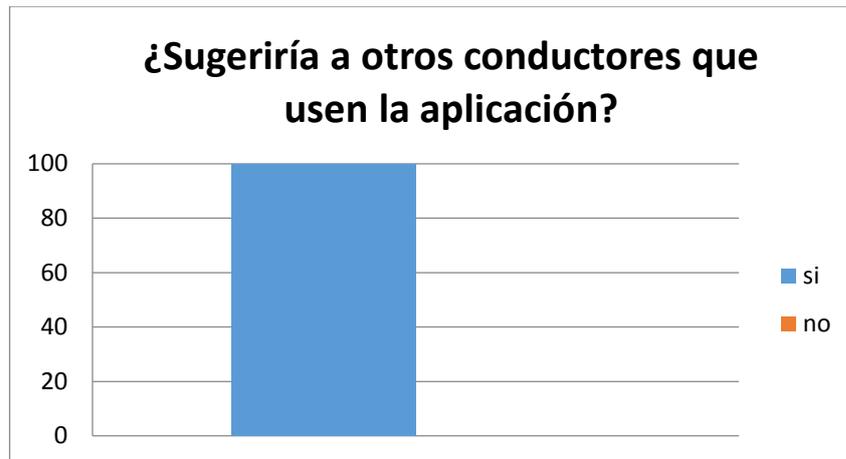


Figura 44 Resultado de la tabulación de la pregunta 4

En el gráfico 4 se puede observar que el 100 % de los conductores sugerirían a otros que usen la aplicación

5. ¿Estaría dispuesto a pagar un costo por la utilización del sistema?

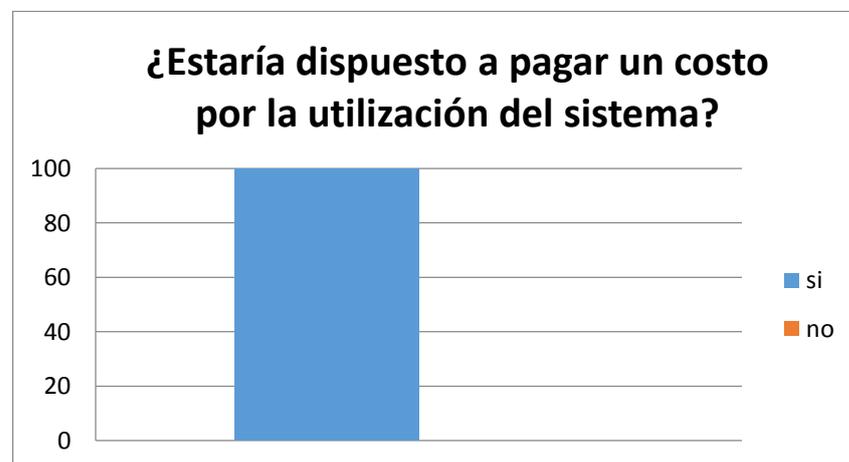


Figura 45 Resultado de la tabulación de la pregunta 5

En el gráfico 5 se puede observar que el 100 % de los conductores consideran que estarían dispuestos a pagar un costo por la utilización del sistema.

En base a las pruebas de usabilidad se obtuvo una respuesta favorable por parte de los conductores, la mayoría de ellos piensan que es fácil de usar, que el tono de alarma es el adecuado, que usarían la aplicación mientras conducen, que sugerirían a otros conductores usarla y que pagarían un costo por su adquisición.

4.5. Verificación de hipótesis

La hipótesis planteada en este proyecto de investigación es: “La implementación de un sistema de vigilancia al conductor vehicular basado en técnicas de visión artificial e implementado en un smartphone permitirá la detección y alerta de somnolencia”

Una vez finalizado el proyecto de investigación y realizado las pruebas necesarias para evaluar la eficiencia de la detección de somnolencia se obtuvo un 92.5% de acierto en la detección de los indicadores de somnolencia con un error del 7.5%, En la revisión sobre las investigaciones relacionadas con el tema realizada por (Garcés, 2015), se tiene que los sistemas de detección de somnolencia desarrollados tienen una eficiencia de alrededor del 80% con un error de alrededor del 20 %, de acuerdo a estos datos, se comprueba la hipótesis habiendo desarrollado un sistema con una eficiencia por encima de la del promedio y con un menor porcentaje de error.

CAPÍTULO 5

5. CONCLUSIONES, RECOMENDACIONES Y TRABAJOS FUTUROS

5.1. Conclusiones

- El sistema de “vigilancia al conductor vehicular basado en técnicas de visión artificial e implementado en un smartphone”, permite la eficiente detección de los indicadores que se presentan en la somnolencia.
- El aumento de las características de procesamiento en los smartphone hizo posible que se desarrolle una aplicación de visión artificial, capaz de detectar el rostro e indicadores visuales presentes en una persona que padece de somnolencia como son: bostezo, movimientos de la cabeza y el estado de los ojos.
- Para validar el sistema se realizaron pruebas de los eventos de forma individual bajo las condiciones establecidas, obteniendo como resultados los siguientes porcentajes: la precisión con que el sistema detectó los bostezos es del 83%, la precisión con que el sistema detectó los asentimientos frontales es del 92%, la precisión con que el sistema detectó los asentimientos a la derecha es del 95%, la precisión con que el sistema detectó los asentimientos a la izquierda es del 90,5% la precisión con que el sistema detectó las distracciones a la derecha es del 92%, la precisión con que el sistema detectó las distracciones a la izquierda es del 96,5%, la precisión con que el sistema detectó los parpadeos es del 98,5%, obteniendo una precisión total del sistema de un 92,5%.
- Existen 3 métodos para la detección de somnolencia, entre ellos el método de detección por medio de indicadores visuales. La implementación de nuestro sistema bajo este método, nos permitió desarrollar un sistema no intrusivo que no interfiere en el desempeño cotidiano de conducción.
- Mediante la librería de visión artificial denominada Google Mobile Api Visión y el IDE Android Studio, ambos programas de uso gratuito

creados por la empresa Google, se desarrolló el sistema de detección de somnolencia el cual fue escrito completamente en el lenguaje de programación java.

- La implementación de un botón para que se oculte la pantalla de procesamiento de la imagen permite que el conductor no se distraiga mirando su smartphone mientras conduce.
- PERCLOS es una medida del estado de somnolencia ampliamente investigado y experimentado desde hace décadas, que permite determinar con un alto valor de precisión la presencia de somnolencia, la implementación de esta medida en nuestro sistema hace este sea muy robusto y preciso al determinar la presencia de somnolencia.
- Los síntomas que presentan las personas durante la transición entre despierta y dormida van apareciendo conforme aumenta la intensidad de la somnolencia, a una mayor intensidad de somnolencia mayor es la pérdida de concentración y capacidad de reacción por parte del conductor
- La implementación de 3 niveles de somnolencia permite que el sistema alerte al conductor sobre su estado no necesariamente en un nivel crítico donde posiblemente se tenga repercusiones graves, sino en niveles tempranos donde la somnolencia recién está apareciendo.
- Como en todo sistema de visión artificial, nuestro sistema también se ve afectado cuando los rayos de luz inciden directamente sobre la cámara, ocasionando que en la imagen no se pueda procesar con éxito los algoritmos de detección del rostro y detección de características faciales.

5.2. Recomendaciones

- La ubicación del smartphone en el vehículo es muy importante ya que de esto depende la correcta detección de los indicadores visuales que permiten generar un nivel de somnolencia, por lo tanto al momento de

posicionar el dispositivo se debe tomar en cuenta los requisitos mencionados en el capítulo 3.

- Para el correcto funcionamiento, el sistema requiere la descarga de recursos de la librería de visión, por lo tanto, la primera vez que el usuario instale la aplicación debe tener acceso a internet, si es exitosa la descarga el sistema enmarca el rostro detectado del conductor.
- Es necesario que el conductor que opte por usar este sistema no lleve puesto gafas de sol, ya que esto impide que el sistema detecte los ojos, que son usados por el algoritmo para calcular PERCLOS80 y microsueño; y que además no luzca un estilo de cabello que le cubra las mejillas puesto que esto impide que se detecte correctamente las distracciones.
- Se recomienda instalar la aplicación en un dispositivo con características similares o superiores al dispositivo usado para la validación, entre más potente sea este, más eficiente será la detección de somnolencia.
- Se recomienda que se verifique el nivel de volumen del smartphone, de ser posible se ajuste en lo más alto, esto para que los niveles de alerta sonoro surjan efecto en el conductor.
- Se recomienda que una vez inicializado el sistema, se oculte la pantalla donde se procesa la imagen mediante el botón “ocultar”, esto con el fin de evitar posibles distracciones por parte del conductor.

5.3. Trabajos futuros

En este trabajo se ha presentado un sistema de detección de somnolencia implementado en un smartphone que trabaja en condiciones de iluminación natural diurna, con el auge de nuevas características en los dispositivos móviles se plantea que se traslade este sistema a un smartphone que cuente con una cámara nocturna y se brinde de un sistema de vigilancia al conductor para las 24 horas del día.

REFERENCIAS BIBLIOGRÁFICAS

- Aranaz, J. (Enero de 2009). *DESARROLLO DE APLICACIONES PARA DISPOSITIVOS MÓVILES SOBRE LA PLATAFORMA ANDROID DE GOOGLE*. Recuperado el 5 de Agosto de 2016, de <http://e-archivo.uc3m.es/handle/10016/6506>
- Bin, M., Khan, I., & Abdullah, H. (27 de Agosto de 2013). *Efficient Drowsiness Detection by Facial Features Monitoring*. Recuperado el 5 de Agosto de 2016, de <http://maxwellsci.com/print/rjaset/v7-2376-2380.pdf>
- Boon-Giin, L., & Wan-Young, C. (2012). *A Smartphone-Based Driver Safety Monitoring System Using*. Recuperado el 5 de Agosto de 2016, de <http://www.mdpi.com/1424-8220/12/12/17536>
- Déniz, Ó., Salido, J., & Bueno, G. (2013). *PROGRAMACIÓN DE APPS DE VISIÓN ARTIFICIAL*. España: Editor Bubok Publishing S.L.
- Echessa, J. (9 de Septiembre de 2015). *SitePoint*. Recuperado el 5 de Agosto de 2016, de Face Detection in Android with Google Play Services: <https://www.sitepoint.com/face-detection-in-android-with-google-play-services/>
- Ecuador Times. (4 de Noviembre de 2013). *Ecuadortimes.net*. Recuperado el 6 de Agosto de 2016, de <http://www.ecuadortimes.net/es/2013/11/04/ecuador-registra-altos-indices-de-accidentes-de-transito/>
- Flores, M., Armingol, J., & Escalera, A. (2011). Sistema Avanzado de Asistencia a la Conduccion para la Detección de la Somnolencia. *ScienceDirect*, 13.
- Fuletra, J., & Bosamiya, D. (Noviembre de 2013). *Academia*. Recuperado el 6 de Agosto de 2016, de A Survey on Driver's Drowsiness Detection Techniques:

https://www.academia.edu/5666888/A_Survey_on_Driver_s_Drowsiness_Detection_Techniques

Garcés, M. A. (2015). Sistemas de detección de somnolencia en conductores: *Revista Ingeniería y Región*, 10.

García, L. (21 de Noviembre de 2007). *Licencias de Software*. Recuperado el 6 de Agosto de 2016, de <http://www.monografias.com/trabajos55/licencias-de-software/licencias-de-software2.shtml#ixzz4SO4ILaBq>

GCFAprendeLibre. (7 de Agosto de 2016). *GCFAprendeLibre*. Recuperado el 3 de Julio de 2016, de https://www.gcfaprendelibre.org/tecnologia/curso/informatica_basica/sistemas_operativos/2.do

Gironés, J. (2012). *El gran libro de Android*. México: Alfaomega.

Gregoire, C. (3 de Diciembre de 2016). *What's Happening In Your Brain During 'Microsleep'*. Recuperado el 7 de Agosto de 2016, de http://www.huffingtonpost.com/entry/microsleep-brain-sleep-deprivation_us_56d9ed81e4b0ffe6f8e958e6

He, J., Roberson, S., Peng, J., Cielocha, S., & Coltea, J. (2013). *Fatigue Detection using Smartphones*. Recuperado el 8 de Agosto de 2016, de <http://www.omicsgroup.org/journals/fatigue-detection-using-smartphones-2165-7556.1000120.pdf>

Invarato, R. (2014). *Android 100%*.

Jimenez, R. (2011). *Sistema de detección de nivel de cansancio en conductores mediante técnicas de visión por computador*. Recuperado el 12 de Agosto de 2016, de <http://www.bdigital.unal.edu.co/6239/1/299970-2011.pdf>

Jo, J., Lee, S. J., Kim, J., Jung, H. G., & Park, K. R. (2011). *SPIE*. Recuperado el 13 de Agosto de 2016, de Vision-based method for detecting driver drowsiness and distraction in driver monitoring system: <http://spie.org/Publications/Journal/10.1117/1.3657506>

- Leira, A. (18 de Abril de 2015). *AndroidStudioFAQs*. Recuperado el 13 de Agosto de 2016, de ¿Qué es gradle en Android Studio?: <http://androidstudiofaqs.com/conceptos/que-es-gradle-en-android-studio2015>
- López, W. (Enero de 2016). *SISTEMA DE CONTROL DEL ESTADO DE SOMNOLENCIA EN CONDUCTORES DE VEHÍCULOS*. Recuperado el 1 de Septiembre de 2016, de <http://repositorio.uta.edu.ec/handle/123456789/16/simple-search?filterquery=L%C3%B3pez+Romero%2C+Wilfrido+Leonel&filtername=author&filtertype>equals>
- macprogramadores. (s.f.). *iOS*. Recuperado el 1 de Septiembre de 2016, de <http://macprogramadores.org/?q=content/ios>
- Maxwell, T. (13 de Agosto de 2015). *9to5google*. Recuperado el 1 de Septiembre de 2016, de Google's new Mobile Vision API makes it easy for Android developers to tell if you're smiling: <https://9to5google.com/2015/08/13/google-mobile-vision-api-android/>
- Molina, Y., Sandoval, J., & Toledo, S. (2012). *SISTEMA OPERATIVO ANDROID: CARACTERÍSTICAS Y FUNCIONALIDAD*. Recuperado el 2 de Septiembre de 2016, de <http://repositorio.utp.edu.co/dspace/handle/11059/2687>
- muycomputer. (23 de Mayo de 2016). *muycomputer*. Recuperado el 12 de Septiembre de 2016, de <http://www.muycomputer.com/2016/05/23/windows-phone-hundido-mercado/>
- OpenCV. (2016). *OpenCV*. Recuperado el 13 de Septiembre de 2016, de <http://opencv.org/about.html>
- Porras, E. (11 de Abril de 2012). *Ingeniería de sistemas*. Recuperado el 16 de Septiembre de 2016, de <http://eve-ingsistemas-u.blogspot.com/2012/04/sistemas-operativos-moviles-ios.html>

- Qualcomm Developer Network. (2016). *FastCV Computer Vision SDK*. Recuperado el 21 de Septiembre de 2016, de <https://developer.qualcomm.com/software/fastcv-sdk>
- Saradadev, M., & Bajaj, P. (Junio de 2008). *Driver Fatigue Detection Using Mouth and Yawning Analysis*. Recuperado el 19 de Septiembre de 2016, de http://paper.ijcsns.org/07_book/200806/20080624.pdf
- Sistemas Adaptativos y Bioinspirados en Inteligencia Artificial S.A.B.I.A. (Diciembre de 2010). *Sistemas Adaptativos y Bioinspirados en Inteligencia Artificial S.A.B.I.A*. Recuperado el 19 de Septiembre de 2016, de <http://sabia.tic.udc.es/gc/Contenidos%20adicionales/trabajos/3D/Visio nArtificial/>
- StopRonquidos. (12 de Marzo de 2015). *StopRonquidos*. Recuperado el 26 de Septiembre de 2016, de <http://www.stopronquidos.com/quedarse-dormido-al-volante-es-una-de-las-primeras-causas-de-accidentes-de-trafico/>
- Stutts, J., Wilkins, J., & Vaughn, B. (1999). *Why Do People Have Drowsy Driving Crashes? Input from Drivers Who Just Did*. Recuperado el 2 de Octubre de 2016, de <https://www.aaafoundation.org/why-do-people-have-drowsy-driving-crashes-input-drivers-who-just-did>
- Svensson, U. (s.f.). *Blink behaviour based drowsiness detection*. Recuperado el 18 de Octubre de 2016, de <http://www.diva-portal.org/smash/get/diva2:673983/FULLTEXT01.pdf>
- Taylor, M. (23 de Diciembre de 2008). *No Doze: Mercedes E-Class alerts drowsy drivers*. Recuperado el 18 de Octubre de 2016, de <http://autoweek.com/article/car-news/no-doze-mercedes-e-class-alerts-drowsy-drivers>
- Torres, Y. (2016). *Repositorio Universidad Nacional de Loja*. Recuperado el 17 de Octubre de 2016, de <https://dspace.unl.edu.ec/jspui/handle/123456789/10861>

Wierwille, W., Tijerina, L., Glecker, M., Duane, S., Johnston, S., & Goodman, M. (1994). *PERCLOS: A Valid Psychophysiological Measure of Alertness As Assessed by Psychomotor Vigilance*. Recuperado el 28 de Octubre de 2016, de <http://ntl.bts.gov/lib/10000/10100/10114/tb98-006.pdf>

ANEXOS



**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN ELECTRÓNICA E INSTRUMENTACIÓN**

CERTIFICACIÓN

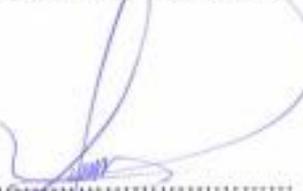
Se certifica que el presente trabajo fue desarrollado por el señor: FABRICIO DANIEL EGAS CUNALATA.

En la ciudad de Latacunga a los 22 días del mes de Agosto de 2017.

Aprobado por:


.....
Ing. Eddie Galarza
DIRECTOR DEL PROYECTO


.....
Ing. Franklin Silva
DIRECTOR DE LA CARRERA


.....
Dr. Rodrigo Vaca
SECRETARIO ACADÉMICO