

Grado en Ingeniería Informática
2019-2020

Trabajo Fin de Grado

“Reconocimiento de emociones y
comportamientos en conductores vía
Deep Learning”

Laura Álvarez Flórez

Tutor

Agapito Ismael Ledezma Espino

Leganés, Octubre 2020



Esta obra se encuentra sujeta a la licencia Creative Commons **Reconocimiento - No Comercial - Sin Obra Derivada**

RESUMEN

Los accidentes de tráfico son causa de numerosas fatalidades cada año, con vistas a reducir este número de muertes, y a mejorar la experiencia de conducción, este proyecto esta orientado al reconocimiento de emociones y distracciones en la carretera, para el desarrollo de una primera versión de un sistema de ayuda a la conducción (ADAS).

Las emociones y las distracciones en la conducción son algunos de los factores determinantes que pueden desencadenar accidentes en la carretera. En este trabajo, se desarrolla un sistema de ayuda a la conducción basado en el conductor. Este sistema, a través de técnicas de Aprendizaje Automático busca obtener información sobre el estado de ánimo del conductor, a través de sus expresiones faciales, así como de la acción que está ejecutando.

El sistema se compone de dos modelos de redes de neuronas convolucionales (CNN), capaces de monitorizar y de determinar los comportamientos y las emociones del conductor a través de imágenes. Ambos modelos han sido desarrollados utilizando técnicas de aprendizaje profundo (*Deep Learning*). El modelo de reconocimiento de emociones, ha logrado alcanzar una *accuracy* de 85.5 % en instancias de test. Mientras que el modelo de distracciones o comportamientos en el conductor, ha alcanzado un 97.7 % de *accuracy*.

Ambos modelos han sido integrados, para formar una primera versión de sistema de ayuda a la conducción, que a través de cámaras, podrá determinar el estado del conductor, para así reducir el número de accidentes y sus consecuencias.

Palabras clave: Aprendizaje Automático, Aprendizaje profundo, Deep Learning, ADAS, Redes de Neuronas Convolucionales, Computación Afectiva, Reconocimiento de emociones, Conductores distraídos.

DEDICATORIA

Con este proyecto finalizo mi grado en Ingeniería Informática.

Han sido cuatro años intensos, de los que me llevo experiencias increíbles.

Gracias a Sabri, Laura y Rafa por compartir conmigo estos años de risas, prácticas interminables, y algún que otro viaje de celebración.

Gracias a mi padre y a mi hermana por estar siempre dispuestos a ayudarme, y a mi madre por ofrecerme siempre calma y una tila en los peores momentos.

Gracias a mis compañeros de piso, por escucharme, porque se saben los detalles de este proyecto casi tanto como yo, y sobre todo, por no permitir que me desanime, incluso en medio de una pandemia.

Y finalmente, gracias a mi tutor Agapito, por permitirme dar rienda suelta a mi creatividad, y aconsejarme durante el desarrollo de este trabajo.

No hubiera sido posible sin vosotros.

ÍNDICE GENERAL

1. INTRODUCCIÓN.	1
1.1. Contexto y motivación.	1
1.2. Objetivos	3
1.3. Marco Regulator.	4
1.4. Estructura del documento	5
1.5. Acrónimos	6
2. ESTADO DEL ARTE.	7
2.1. Emociones	7
2.1.1. Reconocimiento de emociones a través de expresiones faciales	8
2.1.2. La influencia de las emociones en la conducción	9
2.2. Impacto de las distracciones en la conducción.	10
2.3. La psicología del Tráfico y la Seguridad Vial	11
2.4. Sistemas de Ayuda a la Conducción	12
2.5. Inteligencia Artificial	15
2.5.1. Computación Afectiva	15
2.5.2. Aprendizaje Automático.	16
2.5.3. Redes de Neuronas Artificiales	19
2.6. Trabajos similares	25
2.6.1. Reconocimiento de emociones vía Deep Learning	25
2.6.2. Reconocimiento de distracciones en conductores vía Deep Learning	26
2.6.3. ADAS basados en el conductor	27
2.7. Aportaciones al Estado del Arte	28
3. DISEÑO Y ANÁLISIS DEL SISTEMA	29
3.1. Arquitectura del sistema.	29
3.2. Aspectos abordados	30
3.3. Tecnologías utilizadas	31
3.3.1. Proceso de experimentación.	31

3.4. Especificación de requisitos	34
3.4.1. Requisitos Funcionales	35
3.4.2. Requisitos No Funcionales	41
3.5. Casos de uso	44
3.6. Descripción tabular de los casos de uso	44
3.6.1. Descripción gráfica de los casos de uso	47
4. RECONOCIMIENTO DE EMOCIONES.	48
4.1. Recolección de datos.	48
4.1.1. The Extended Cohn-Kanade Dataset (CK+).	48
4.1.2. Facial Expression Recognition Challenge: FER (2013)	49
4.1.3. The japanese facial expression (JAFFE)	50
4.1.4. The Karolinska Directed Emotional Faces (KDEF).	50
4.2. Unificación de los conjuntos	51
4.3. Preprocesado de imágenes.	52
4.3.1. Ajuste de dimensiones	52
4.3.2. Creación de subconjuntos de datos	54
4.4. Desarrollo del modelo	55
4.4.1. Elección de hiper-parámetros y arquitectura.	55
4.5. Experimentación y ajuste	58
4.6. Modelo final	62
4.7. Ambigüedad en los resultados	64
5. RECONOCIMIENTO DE COMPORTAMIENTO EN CONDUCTORES.	65
5.1. Descripción de datos.	65
5.2. Preparación de los datos	67
5.3. Construcción del modelo: Experimentación y ajuste	70
5.4. Modelo final	73
6. INTEGRACIÓN Y PRUEBAS DEL ADAS	75
6.1. Integración de modelos de aprendizaje automático	75
6.2. Pruebas unitarias del sistema	76
6.3. Resultados de las pruebas	85

7. GESTIÓN DEL PROYECTO	87
7.1. Planificación	87
7.2. Presupuesto	90
7.3. Impacto socio-económico	92
8. CONCLUSIONES	93
8.1. Conclusiones generales	93
8.2. Limitaciones y dificultades encontradas	93
8.3. Trabajos futuros	94
BIBLIOGRAFÍA	95
9. ANEXO A	
10. ANEXO B	

ÍNDICE DE FIGURAS

1.1	Crecimiento del número de vehículos en los próximos 25 años [2]	1
1.2	Número de muertes por causa en 2017 [3]	1
2.1	Emociones básicas de Paul Ekman	7
2.2	Expresiones faciales y sus emociones asociadas	9
2.3	Conductor irritado [16]	10
2.4	Niveles de conducción autónoma [18]	12
2.5	Línea temporal de los ADAS	14
2.6	Áreas de la Inteligencia Artificial [24]	16
2.7	Tipos de Aprendizaje Automático [25]	17
2.8	Ciclo de vida del Aprendizaje Automático [26]	18
2.9	Perceptrón simple [27]	19
2.10	Clasificación lineal y no lineal [28]	20
2.11	Modelo de Redes de Neuronas [29]	21
2.12	Aprendizaje de una red de neuronas artificial [30]	21
2.13	Ejemplo de la arquitectura de una red convolucional [31]	22
2.14	Deep Dive Afectiva Automotive AI [47]	27
3.1	Arquitectura del sistema	29
3.2	Experimentos en Comet ML	32
3.3	Almacenamiento de filtros en Comet ML	32
3.4	Instancias mal clasificadas almacenadas en Comet ML	33
3.5	Matriz de confusión en Comet ML	33
3.6	Representación gráfica de los casos de uso	47
4.1	Muestra de imágenes del set de datos CK	49
4.2	Muestra de imágenes del set de datos FER2013	50
4.3	Muestra de imágenes del set de datos JAFFE	50
4.4	Muestra de imágenes del set de datos KDEF	51

4.5	Resultados de la reducción de dimensiones para cada conjunto de datos	52
4.6	Proceso de preprocesado de imágenes	53
4.7	Comparación de un experimento válido (Experiment 1) y uno no válido	61
4.8	Representation 3D de la mejor arquitectura de CNN	62
4.9	Evolución de <i>accuracy</i> en mejor modelo	62
4.10	Matriz de confusión del mejor modelo	63
4.11	Ejemplos de posibles errores en el etiquetado	64
5.1	Vistas de las imágenes en 3MDAD	65
5.2	Muestra de las clases representadas en el conjunto de imágenes	66
5.3	Redimensionado de las imágenes	67
5.4	Ejemplo de secuencia de imágenes para eliminación	68
5.5	Evolución en el entrenamiento del modelo	73
5.6	Evolución en la matriz de confusión tras varios experimentos	74
5.7	Representación en 3D de la CNN	74
6.1	Diagrama de flujo del sistema	75
6.2	Explicación del resultado del sistema	85
6.3	Selección de resultados de las pruebas del sistema	86
7.1	Diagrama de Gantt del proyecto	89
9.1	Diagrama de Gantt del proyecto por días 1	
9.2	Diagrama de Gantt del proyecto por días 2	
9.3	Diagrama de Gantt del proyecto por días 3	
10.1	Global transport growth over the next 25 years [2]	
10.2	Preprocessing steps	
10.3	3D representation of CNN architecture	
10.4	3D representation of CNN architecture	
10.5	ADAS preview	

ÍNDICE DE TABLAS

1.2	Haddon matrix	2
3.1	Plantilla de tabla de requisitos	34
3.2	Requisito funcional RF-01	35
3.3	Requisito funcional RF-02	35
3.4	Requisito funcional RF-03	35
3.5	Requisito funcional RF-04	36
3.6	Requisito funcional RF-05	36
3.7	Requisito funcional RF-06	36
3.8	Requisito funcional RF-07	37
3.9	Requisito funcional RF-08	37
3.10	Requisito funcional RF-09	37
3.11	Requisito funcional RF-10	38
3.12	Requisito funcional RF-11	38
3.13	Requisito funcional RF-12	38
3.14	Requisito funcional RF-13	39
3.15	Requisito funcional RF-14	39
3.16	Requisito funcional RF-15	39
3.17	Requisito funcional RF-16	40
3.18	Requisito funcional RF-17	40
3.19	Requisito funcional RF-18	40
3.20	Requisito funcional RF-19	41
3.21	Requisito funcional RF-20	41
3.22	Requisito no funcional RNF-01	41
3.23	Requisito no funcional RNF-02	42
3.24	Requisito no funcional RNF-03	42
3.25	Requisito no funcional RNF-04	42
3.26	Requisito no funcional RNF-05	43

3.27	Requisito no funcional RNF-06	43
3.28	Requisito no funcional RNF-07	43
3.29	Plantilla tabla de casos de uso	44
3.30	Caso de uso CU-01	45
3.31	Caso de uso CU-02	45
3.32	Caso de uso CU-03	45
3.33	Caso de uso CU-04	45
3.34	Matriz de trazabilidad entre requisitos y casos de uso	46
4.1	Emociones representadas en los datos	51
4.2	Guía para entender los códigos asociados a los subconjuntos de imágenes	54
4.3	Descripción de subconjuntos de imágenes creados	54
4.4	Arquitectura redes convolucionales 1-5	55
4.5	Arquitectura redes convolucionales 6-10	56
4.6	Arquitectura redes convolucionales 11-15	56
4.7	Arquitectura redes convolucionales 16-20	57
4.8	Descripción y valores de hiperparámetros	58
4.9	Resultados experimentación	59
4.10	Parámetros de la búsqueda bayesiana	60
4.11	Resultados obtenidos de la búsqueda bayesiana	60
4.12	Resumen de la experimentación	61
5.1	Descripción de las etiquetas del conjunto de datos	66
5.2	Preprocesado aplicado a las imágenes de 3MDAD	68
5.3	Descripción de los subconjuntos creados a partir de 3MDAD	69
5.4	Código asignado a cada subconjunto	69
5.5	Arquitectura redes convolucionales 1-4	70
5.6	Arquitectura redes convolucionales 5-9	71
5.7	Resultados de los experimentos	72
5.8	Resumen de la experimentación	72
6.1	Plantilla de tabla de pruebas unitarias	76
6.2	Prueba unitaria PU-01	77

6.3	Prueba unitaria PU-02	77
6.4	Prueba unitaria PU-03	77
6.5	Prueba unitaria PU-04	78
6.6	Prueba unitaria PU-05	78
6.7	Prueba unitaria PU-06	78
6.8	Prueba unitaria PU-07	79
6.9	Prueba unitaria PU-08	79
6.10	Prueba unitaria PU-09	79
6.11	Prueba unitaria PU-10	80
6.12	Prueba unitaria PU-11	80
6.13	Prueba unitaria PU-12	80
6.14	Prueba unitaria PU-13	81
6.15	Prueba unitaria PU-14	81
6.16	Prueba unitaria PU-15	81
6.17	Prueba unitaria PU-16	82
6.18	Prueba unitaria PU-17	82
6.19	Prueba unitaria PU-18	82
6.20	Prueba unitaria PU-19	83
6.21	Prueba unitaria PU-20	83
6.22	Matriz de trazabilidad entre requisitos y pruebas unitarias	84
7.1	Planificación del proyecto	87
7.2	Tabla de costes de personal	90
7.3	Costes del proyecto en hardware, software e indirectos	91
7.4	Presupuesto total del proyecto	91
10.1	Summary of the experiments	
10.2	Best model hyperparameters	
10.3	Distracted drivers problem's classes	
10.4	Summary of the experiments	

1. INTRODUCCIÓN

1.1. Contexto y motivación

Hoy en día la forma más extendida de viajar, o conmutar entre ciudades, es mediante el uso de un vehículo personal. La industria del automóvil es uno de los sectores más grandes, y que más factura, siendo las ventas para el año 2020 estimadas en 59.9 millones de unidades [1].

Lo que solía ser una forma de viajar exclusiva, y no al alcance de cualquiera, a día de hoy se ha expandido alrededor de todo el mundo. En la Figura 1.1 se puede observar una interesante comparativa entre la situación aproximada que se vivió en 2015, y la situación que se prevé para el 2040. Estas predicciones estiman que el número de coches en el mundo se doblará con respecto a la cifra obtenida en 2015 antes del 2050 [2]. En uno de los posibles escenarios para el futuro, un aumento del número de coches tendrá como consecuencia un aumento en el número de víctimas en la carretera. Esta perspectiva convierte en crucial la investigación y desarrollo en la prevención de accidentes en la carretera.

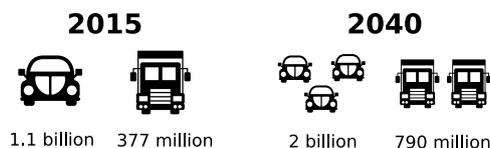


Fig. 1.1. Crecimiento del número de vehículos en los próximos 25 años [2]

De acuerdo al estudio realizado por la Organización Mundial de la Salud (OMS) en 2017, los accidentes de tráfico son causa de muerte para un total de 1.35 millones de personas al año [3]. En la Figura 1.2 se puede comparar este número con algunas de las causas de muerte más comunes a día de hoy. Como se puede observar, los accidentes de tráfico se aproximan en número de muertes al que causan algunas de las enfermedades más letales como la tuberculosis.

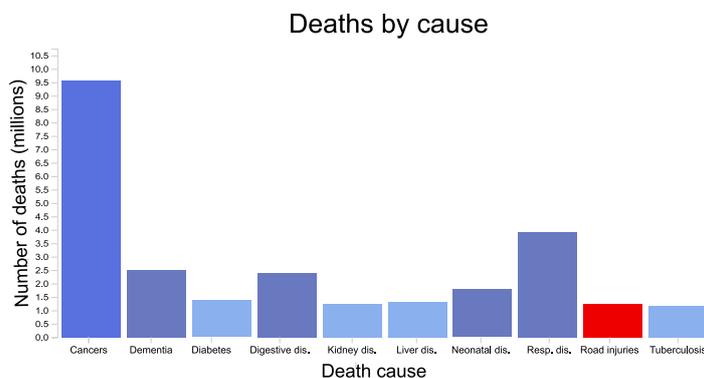


Fig. 1.2. Número de muertes por causa en 2017 [3]

Sin embargo, la alarmante cifra del número de muertes en carretera es sólo la punta del iceberg. Los accidentes de tráfico causan heridas no fatales, que pueden desencadenar una discapacidad permanente, a un total de entre 29 y 50 millones de personas al año [4].

Por otro lado, las consecuencias causadas por los accidentes en carretera, no sólo implican vidas humanas, sino que también incluyen los costes, tanto monetarios como emocionales de los familiares implicados, al igual que el impacto en la economía de cada país. De acuerdo a la OMS, los accidentes de tráfico causan, en la mayoría de los países, gastos estimados en un 3 % de su producto interno bruto [4].

Para poder lograr disminuir el número de accidentes en la carretera, es vital tener un entendimiento sobre los factores que desencadenan dichas situaciones. William Haddon ha determinado una matriz [5] en la que se pueden identificar los factores de riesgo antes, durante, y después de un accidente de tráfico. Estos factores se dividen en la matriz en tres categorías diferentes, dependiendo de si están relacionados con el conductor, el vehículo, o el entorno.

En la Tabla 1.2 se puede ver una representación de la matriz de Haddon. Este proyecto se enfocará en proponer una solución que concierne a los factores humanos que propician el accidente. Estos están remarcados en color azul en la matriz, e incluyen las actitudes e información presentes en el conductor.

PHASE		FACTORS		
		HUMAN	VEHICLES AND EQUIPMENT	ENVIROMENT
PRE-CRASH	Crash prevention	Information Attitudes Impairment Police enforcement	Road-worthiness Lighting Braking Handling Speed management	Road design and layout speed limits pedestrian facilities
CRASH	Injury prevention during the crash	Use of restraints Impairment	Occupant restraints Other safety devices Crash protective design	Crash-protective roadside objects
POST-CRASH	Life sustaining	First-aid skill Access to medics	Ease of access Fire risk	Rescue facilities Congestion

TABLA 1.2. HADDON MATRIX

De la misma manera, en un estudio realizado por Petridou et al. se determinó que los factores relativos al comportamiento del conductor están presentes en tres de cada cinco accidentes de tráfico [6]. De su investigación, se puede determinar irritabilidad y enfado como algunos de los estados de ánimo que son frecuentes en el conductor, previamente a tener un accidente de tráfico. Otros de los factores, que se determinaron como desencadenantes, fueron el estrés en los conductores y las distracciones temporales. La relación existente entre el comportamiento del conductor y la posibilidad de accidente será estudiada con más detalle a lo largo de este trabajo.

Con el fin de proponer una solución al número de fatalidades en la carretera, este proyecto propone el uso de las técnicas más pioneras en la computación para la creación de un sistema de ayuda a la conducción basado en el conductor y en sus comportamientos y emociones.

1.2. Objetivos

Siendo los accidentes en carretera un problema que afecta a la salud y el bienestar de una gran parte de la población, este proyecto está motivado por la búsqueda de una solución que consiga reducir estos sucesos, y por ende, sus consecuencias negativas. Este proyecto tiene como objetivo principal el desarrollo de un sistema capaz de reconocer emociones y comportamientos en conductores a través de técnicas de Deep Learning.

Cuando en el vehículo se presenta una situación peligrosa, es crucial alertar al conductor y a los pasajeros, para que puedan reaccionar a tiempo. Siendo el tiempo de reacción un factor clave del problema, en este proyecto se tendrá en cuenta no sólo el reconocimiento de emociones y comportamientos de riesgo al volante, sino que también su aplicación en tiempo real.

Para llevar a cabo esto, en este proyecto se han determinado los siguientes objetivos específicos:

- Estudiar el impacto de las emociones y distracciones en la conducción.
- Desarrollar un modelo de Deep Learning capaz de determinar las emociones presentes en el conductor a través de una imagen del mismo.
- Desarrollar un modelo de Deep Learning capaz de determinar los comportamientos o acciones presentes en el conductor a través de una imagen del mismo.
- Diseñar un sistema de ayuda a la conducción (ADAS), basado en el conductor, utilizando los modelos desarrollados.

1.3. Marco Regulador

Este trabajo incluye el desarrollo de un prototipo para un sistema de ayuda a la conducción, que estará centrado en el conductor. Estos sistemas forman parte de los primeros niveles de automatización, véase Figura 2.4, es por eso que la ley que rige estos sistemas se diferencia de la que ha de ser aplicada en sistemas que buscan una mayor automatización en el vehículo. En concreto, el sistema diseñado, tan solo influirá en el vehículo proporcionando una alarma o estímulo ante ciertos comportamientos. Es por esto que, los aspectos legales que ha de cumplir el sistema, se reducen a la Instrucción 15/V-113 de la DGT. Esta medida exige la autorización previa para la realización de las pruebas de un vehículo que haya sido modificado con comportamientos autónomos.

Por otro lado, consecuencia de la incorporación de una cámara en el vehículo, que tomará continuamente imágenes del conductor, será necesario la contemplación de la Ley Orgánica 3/2018, de 5 de diciembre, de Protección de Datos Personales y garantía de los derechos digitales. Esta ley busca proteger al usuario de un uso inadecuado de sus datos personales y garantizar su seguridad. Con el fin de asegurar la privacidad del usuario final del sistema, han sido establecidas las siguientes regulaciones:

- Los datos obtenidos no serán almacenados sin el consentimiento previo del usuario.
- Los datos recogidos por el sistema no serán utilizados ni facilitados a terceros sin previo consentimiento del usuario.
- El usuario podrá revocar el consentimiento ofrecido con carácter inmediato.

1.4. Estructura del documento

Este documento se compone de un total de ocho capítulos y dos anexos. En esta sección se presentarán brevemente los contenidos que incluye cada uno de estos.

- Capítulo 1. Introducción al proyecto, se exponen el problema a resolver, su contexto legal y como será resuelto. De la misma manera, se incluyen los aspectos que se han abordado durante el proyecto. Finalmente, se presenta la estructura del documento y una lista de los acrónimos que serán utilizados.
- Capítulo 2. Estado del arte, se presentan los conocimientos necesarios para entender el problema y su solución, así como los trabajos similares realizados hasta el momento.
- Capítulo 3. Se presenta un diseño para el sistema propuesto, se detallan los aspectos abordados y las tecnologías usadas. Además, se realiza la especificación de requisitos y casos de uso del proyecto.
- Capítulo 4. Este capítulo se enfoca en el modelo de reconocimiento de emociones. Se presentan los datos utilizados, los experimentos realizados, y los resultados obtenidos.
- Capítulo 5. Este capítulo se enfoca en el modelo de reconocimiento de acciones o distracciones en conductores. Se presentan los datos utilizados, los experimentos realizados, y los resultados obtenidos.
- Capítulo 6. Este capítulo abarca la integración de los modelos desarrollados en un sistema. Se detalla el funcionamiento interno del algoritmo y se presentan las pruebas realizadas. Finalmente, se incluye una demostración visual del sistema desarrollado.
- Capítulo 7. Este capítulo presenta la gestión del proyecto, incluyendo tanto la planificación realizada como el impacto socio-económico del sistema.
- Capítulo 8. El documento finaliza con unas conclusiones generales del proyecto, presentando las dificultades encontradas y los futuros trabajos.
- Anexo A. Se incluye una versión extendida del diagrama de Gantt por días.
- Anexo B. Este anexo incluye un resumen extenso en inglés del trabajo, requisito para demostrar las competencias del alumno en el idioma.

1.5. Acrónimos

- **AA** Aprendizaje Automático.
- **ADALINE** ADAptative LINEal Model.
- **ADAS** Advanced Driving Assistance System, en castellano, sistema avanzado de ayuda a la conducción.
- **CNN** Convolutional Neural Network, en castellano, red de neuronas convolucional.
- **CV** Cross Validation, en castellano, validación cruzada.
- **DGT** Dirección General de Tráfico.
- **FER** Facial Expression Recognition, en español, reconocimiento de expresiones faciales.
- **GPS** Global Positioning System, en español, sistema de posicionamiento global.
- **IA** Inteligencia Artificial.
- **LIDAR** Light Detection and Ranging
- **ML** Machine Learning, en español, Aprendizaje Automático.
- **NN** Neural Network, en español, redes de neuronas.
- **OMS** Organización Mundial de la Salud.

2. ESTADO DEL ARTE

2.1. Emociones

A día de hoy, la comunidad científica todavía no ha llegado a un consenso para definir la palabra emoción. De acuerdo al psicólogo Paul Ekman, experto en la materia, se puede definir una emoción de la siguiente manera:

“Emotions are a process, a particular kind of automatic appraisal influenced by our evolutionary and personal past, in which we sense that something important to our welfare is occurring, and a set of psychological changes and emotional behaviors begins to deal with the situation.” [7]

Cuando aparece una emoción, se experimentan una serie de cambios en el sistema nervioso. Estos cambios suelen ser percibidos a través de las expresiones faciales o el tono de voz de la persona que está experimentando dicha emoción.

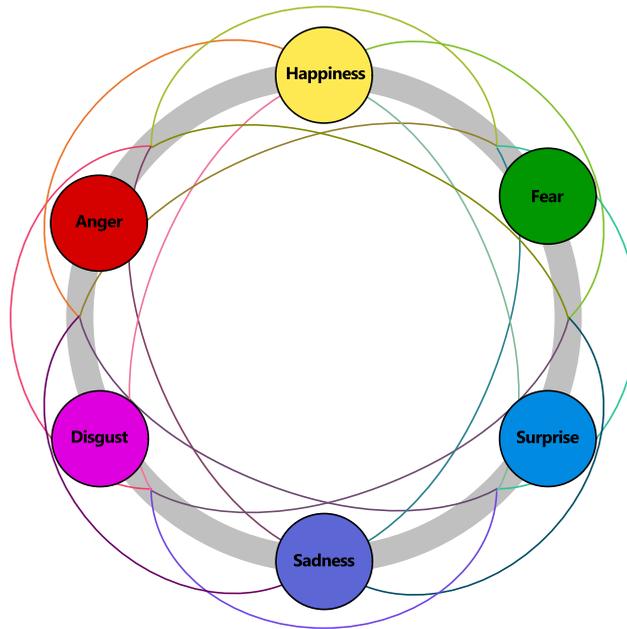


Fig. 2.1. Emociones básicas de Paul Ekman

El escritor de la definición de emoción recién mencionada, es también uno de los investigadores más renombrados en el ámbito de las emociones, y el primer autor en definir cuales son las diferentes emociones que pueden ser experimentadas por los seres humanos. En su trabajo [8], Ekman establece que existen un total de seis emociones diferentes, a las que denomina como *“the six basic human emotions”*. En la Figura 2.1 se pueden ver representadas las seis emociones definidas por Ekman. Diez años después de su publicación, sus descubrimientos aún siguen utilizándose, y han servido como base a muchos otros investigadores, para después proponer modificaciones y añadir nuevas emociones.

La estrategia propuesta por Ekman para categorizar las emociones se denomina una estrategia discreta, ya que las emociones definidas se disponen sobre una sola dimensión. Además de la estrategia discreta, existen otros enfoques para categorizar las emociones en los que se utiliza más de una dimensión para clasificar las emociones. El modelo circunplejo de las emociones de James Russel [9], es un ejemplo de uno de los modelos dimensionales más estudiados. En contraste con el modelo propuesto por Ekman, Russel utiliza dos dimensiones para clasificar cada una de las emociones en una categoría.

2.1.1. Reconocimiento de emociones a través de expresiones faciales

Cuando se presenta una emoción, esta suele ser expresada a través de la comunicación verbal y no verbal. Una de las maneras más comunes de expresar una emoción es a través del cambio en las expresiones faciales.

En 1872, Charles Darwin determinó que las expresiones de las emociones son universales en todas las culturas [10]. Esto implica la posibilidad de encontrar patrones en las expresiones producto de experimentar una emoción. Con el fin de probar que Darwin se equivocaba, el psicólogo Paul Ekman realizó numerosos viajes para estudiar las expresiones utilizadas en diferentes culturas. Sus investigaciones le llevaron a Papúa Nueva Guinea, uno de los sitios más recónditos del planeta, donde aún sin tener acceso a ningún medio de comunicación, la tribu que analizó seguía mostrando las mismas emociones y expresándolas a través de los mismos patrones [11]. Probando así que la hipótesis de Darwin era correcta, y que se existen unas características comunes para expresar cada una de las emociones básicas.

Las expresiones faciales pueden ser tanto voluntarias como involuntarias, dependiendo de la parte del cerebro que esté involucrada en dicha expresión. La parte visible al ojo humano, viene dada por el movimiento muscular que esté asociado con dicha actividad cerebral. Es por esto que existen diferentes alternativas en cuanto a la detección de emociones, incluyendo desde sensores especializados hasta imágenes.

En [12], Dzedzickis et al. estudian las diferentes técnicas que pueden ser utilizadas para llevar a cabo el reconocimiento de emociones vía sensores. Entre estos sensores se puede encontrar la medición de la actividad electro-dérmica. Estas técnicas son más precisas y fiables si se compara con el uso de imágenes, sin embargo, las imágenes resultan una alternativa más accesible y menos costosa de poder determinar de manera automática una emoción.

Algunos de los patrones presentes en las expresiones faciales, que están asociados a cada una de las emociones, pueden ser encontrados en [13]. Esta información ha sido resumida en la Figura 2.2, donde se puede observar una imagen de una representación de cada una de las emociones básicas, y una definición de los patrones presentes en dicha expresión facial. Estas características, como boca abierta o cejas levantadas, representan algunos de los patrones que el cerebro asocia con cada una de las emociones y que le

permite reconocerlas.

Esta idea recién presentada sobre el reconocimiento de patrones es lo que se busca conseguir con el reconocimiento automático de emociones, con la diferencia de que, en vez del cerebro humano, es un algoritmo el que busca poder determinar en una imagen los patrones necesarios para así asociarla con una emoción.



Fig. 2.2. Expresiones faciales y sus emociones asociadas

2.1.2. La influencia de las emociones en la conducción

Un conductor tiene el control sobre el coche que conduce, por lo tanto, un conductor concentrado en la carretera, el tráfico, y el entorno puede prever un peligro y reaccionar de manera acorde a este. Por otro lado, si el conductor se encuentra en un estado emocional comprometido, el cual le está llevando a desviar su atención de la carretera y enfocarla en sus pensamientos, entonces la capacidad de conducción se está viendo afectada por el estado emocional del conductor.

Muchos expertos han estudiado los efectos que pueden llegar a tener las emociones en la conducción. Una de las investigaciones más recientes ha demostrado que los sentimientos negativos como el enfado, la tristeza o la ansiedad pueden tener un impacto desfavorable en la capacidad de conducción de un conductor [14]. Por otro lado, los efectos de las emociones positivas han sido menos estudiados en comparación. Sin embargo,

algunos experimentos descritos en [14] indican que las emociones positivas también pueden llevar a incrementar el número de errores humanos en la conducción.

De todas las emociones presentes durante la conducción, el estado de enfado o la irritación es la más frecuente y repetida entre los conductores. Este sentimiento de irritación prevalece en la carretera influenciado por factores externos presentes en el entorno como el tráfico o el ruido [14].

Creando la simulación de un entorno de conducción, y presentando a los sujetos con estímulos, Steinhäuser et al. [15] determinaron que las emociones afectan a la conducción de manera directa (promoviendo la conducción agresiva) e indirecta (reduciendo la capacidad de realizar dos acciones al mismo tiempo). Estos resultados son evidencia de la relación existente entre las emociones y la capacidad de conducción.

El estudio realizado por Pêcher et al. [14] determina que la principal causa del estado irritado de los conductores es el tráfico. Este interesante hallazgo puede llevar a uno mismo a preguntarse cómo puede ser evitada esta situación de irritación, si no se puede evitar la causa que la provoca. En cualquier caso, la introducción de un sistema capaz de detectar el estado de irritación, puede ser un primer paso, en la elaboración de un sistema que pueda no solo detectar el estado de irritación sino actuar para disiparlo, y crear una atmósfera de conducción más relajada y menos propensa a accidentes.



Fig. 2.3. Conductor irritado [16]

2.2. Impacto de las distracciones en la conducción

Además de las emociones, existen otros factores que alteran a la conducción de una manera más directa. Estos otros factores incluyen las distracciones provocadas por comportamientos al volante como pueden ser el uso del teléfono móvil para llamar, mandar mensajes, el GPS, etc.

Más sobre estos comportamientos será desarrollado a lo largo de esta sección. No obstante, cabe destacar la relación existente entre las distracciones al volante y las emociones. Un ejemplo de esta relación puede ser un conductor en estado emocional triste,

que debido a esta sensación que experimenta, se ve tentado a coger el teléfono y realizar una llamada telefónica en busca de ayuda emocional. Con este ejemplo, se pretende ilustrar el potencial poder desencadenante de comportamientos peligrosos al volante de las emociones.

Un estudio realizado por *Virginia Tech Transportation Institute*(VTTI) reveló que alrededor del 80 % de accidentes en la carretera involucraban algún tipo de inatención por parte del conductor durante los tres segundos previos al accidente [17]. Este artículo también proporciona información interesante sobre cómo cada uno de los comportamientos de distracción típicos puede afectar la seguridad de los pasajeros. El artículo establece que un conductor que envía mensajes de texto tiene 23,2 veces más probabilidades de sufrir un accidente automovilístico, y 2,8 veces más probabilidades si el conductor está marcando un número para llamar.

De la misma manera, un estudio realizado en 2016 por *University of Iowa* afirma que el 60 % de los accidentes de adolescentes implican distracciones. Estas estadísticas ilustran la relación existente entre ciertas actividades al volante y los accidentes.

2.3. La psicología del Tráfico y la Seguridad Vial

La psicología del Tráfico y la Seguridad Vial es una rama de la psicología que nace de la necesidad de responder a preguntas relacionadas sobre cómo ciertos comportamientos afectan a la conducción. Este campo se centra en el entendimiento y descubrimiento de los comportamientos presentes en los conductores, tanto los que son planeados como los que nacen de manera impulsiva. Esta rama, mediante el estudio del conductor y su comportamiento, tiene entre sus objetivos la prevención de accidentes, así como cualquier otro inconveniente que pueda aparecer en la carretera.

Una de las áreas dentro de la psicología del tráfico es la investigación en accidentes. En este campo, se estudian los factores relevantes como entorno del conductor, así como sus actitudes, aptitudes y contexto, con el fin de encontrar patrones y relaciones que puedan luego ser usadas para prevenir accidentes. A continuación, se listan algunos de estos factores frecuentes:

- Toma de decisiones.
- Distracción.
- Inatención.
- Comportamientos de riesgo.
- Estrés.

En este proyecto se presentará una solución capaz de detectar algunos de estos factores, sin supervisión humana. Esto les dará a los investigadores en este campo un nuevo

enfoque, para obtener y recopilar datos, que les permitirá analizar información de situaciones de la vida real de manera automática.

2.4. Sistemas de Ayuda a la Conducción

Con el fin de ayudar a los conductores a tener un viaje seguro, y reducir el número de accidentes automovilísticos que ocurren todos los días, los sistemas avanzados de asistencia al conductor (ADAS) se han introducido gradualmente en los vehículos terrestres. Mediante el uso de cámaras, sensores o radares, estos sistemas proporcionan viajes más seguros y reducen el número de lesiones en la carretera.

Estos sistemas han sido los precursores de lo que hoy se conocen como coches autónomos; sin embargo, es importante mencionar que esta tecnología se enfoca en ayudar al conductor, y no se enfoca en conducir el vehículo sin supervisión humana como harían los vehículos autónomos. Los ADAS tienen como objetivo reducir el error humano, mientras que los humanos permanecen principalmente en control del automóvil.

La principal diferencia entre ADAS y los vehículos autónomos, es que en los últimos los vehículos son controlados en gran parte, o totalmente, por un software. Pensar en un ADAS como un automóvil autónomo es un error común, para aclarar este tema en 2016 la Administración Nacional de Seguridad del Tráfico en las Carreteras (NHTSA) declaró los diferentes niveles de automatización. La Figura 2.4 [18] describe los niveles definidos de conducción autónoma.

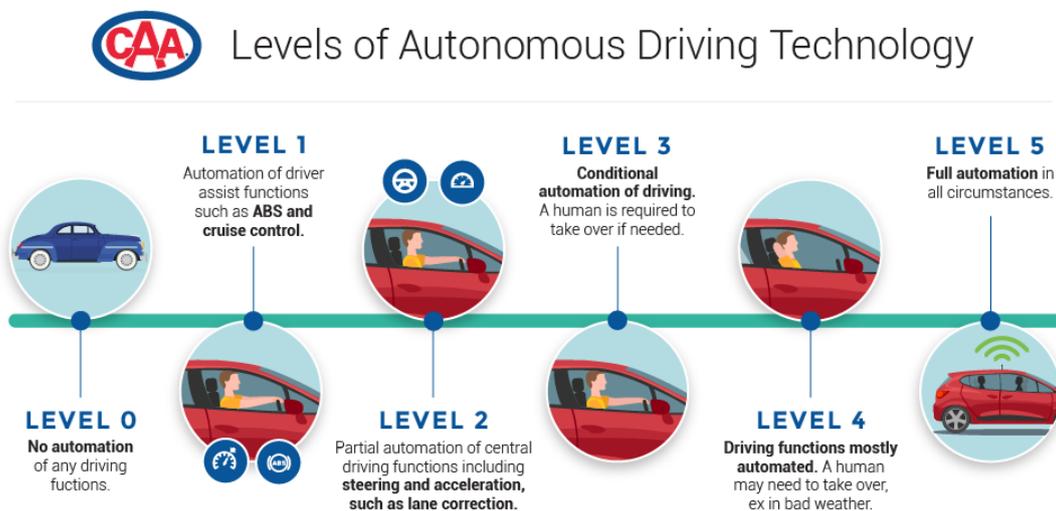


Fig. 2.4. Niveles de conducción autónoma [18]

- El Nivel 0 y el Nivel 1 incluyen la mayoría de los automóviles y otros vehículos terrestres disponibles en la actualidad. Estos no incluyen ningún sistema de ayuda, y si lo incluyen, se tratará o bien de un sistema de control de velocidad o un sistema de frenado automático.

- El Nivel 2 es donde se pueden encontrar los sistemas avanzados de asistencia al conductor (ADAS), a diferencia de los vehículos que se sitúan en la categoría de Nivel 3 de automatización, los vehículos en el Nivel 2 no tienen información ambiental con respecto al exterior, y por ende, no pueden conducir de manera autónoma. En este nivel se encuentran sistemas con una automatización parcial del vehículo, un ejemplo de estos sistemas es el asistente de mantenimiento de carril.
- El Nivel 3 incluye automóviles con tecnologías que pueden tomar decisiones informadas basadas en el entorno del automóvil, pero aún necesitan un ser humano para cuidar la situación de conducción.
- En el Nivel 4 se encuentran los vehículos que pueden operar por sí mismos, con ayuda humana, pero deben estar en un entorno controlado.
- El último de los niveles de automatización, Nivel 5, es la automatización total de un automóvil. Es decir, este puede conducir, sin supervisión humana, en cualquier entorno.

Los coches autónomos están dispuestos a sustituir al conductor en algunas o todas las tareas. Por el contrario, los sistemas avanzados de asistencia al conductor están diseñados para ayudar cuando el conductor se ve envuelto en una situación peligrosa.

Para este proyecto, algunas de las invenciones más importantes que se han logrado en los ADAS han sido recopiladas en la Figura 2.5. Esta nos muestra como estos sistemas han evolucionado a lo largo de los años, resumiendo la información encontrada en [19] y [20].

Como se muestra en la Figura 2.5, los primeros sistemas de ayuda a la conducción (ADAS), que tenían como objetivo reducir errores inesperados en la conducción humana, se centraron simplemente en alertar al conductor cada vez que el automóvil excedía el límite de velocidad de conducción.

En la década de los 90, se comenzaron a integrar en los automóviles sistemas más innovadores, capaces de medir la concentración de alcohol en el aliento del conductor antes de arrancar el automóvil. Estos fueron seguidos por sistemas que usaban sensores para detectar peatones o posibles colisiones. Esta línea temporal presente en la Figura 2.5 muestra como los ADAS han evolucionado a lo largo de las décadas para ayudar a los conductores en tareas cada vez más complejas.

Linea Temporal de los ADAS

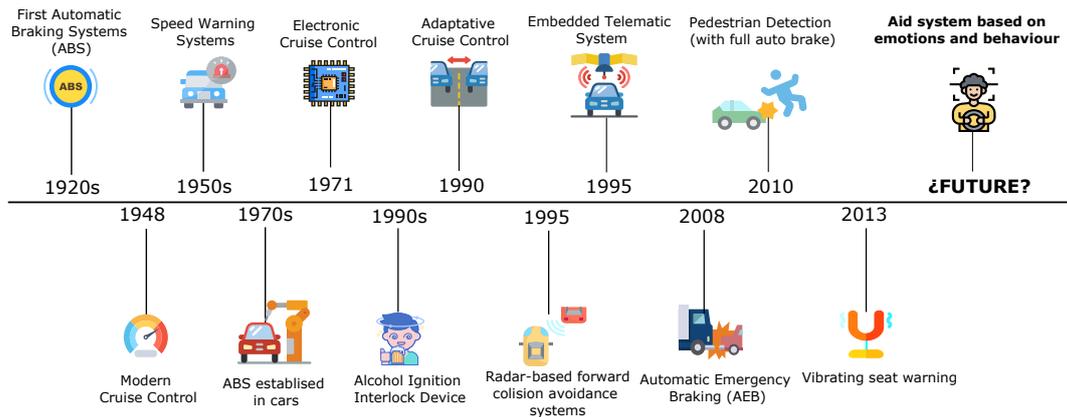


Fig. 2.5. Linea temporal de los ADAS

Hoy en día, los ADAS utilizan combinaciones de sensores y cámaras de vídeo para recopilar información relevante del automóvil y su entorno. La siguiente lista incluye componentes que se utilizan ampliamente en estos sistemas:

- **Cámaras de vídeo.** Se utilizan comúnmente para la detección y el reconocimiento de objetos. Usando cámaras y técnicas de Inteligencia Artificial, los ADAS puede detectar peatones, leer señales y moderar la velocidad en base a esa información.
- **Radares.** Pueden detectar objetos en movimiento, así como sus características relevantes como velocidad, distancia o peso.
- **Sensores ultrasónicos.** Estos sensores son capaces de detectar obstáculos alrededor del vehículo. Las ondas ultrasónicas son enviadas por el sensor. Estas ondas viajan hasta 6 metros, y cuando se detecta un objeto, envían la información de regreso al automóvil. Por lo general, se usan para el asistente de estacionamiento, ya que son increíblemente precisos cuando el automóvil circula a baja velocidad y los objetos están cerca.
- **LIDAR.** Este es el nombre que se utiliza para los sensores láser. A diferencia de los sensores ultrasónicos, pueden detectar objetos que se encuentran en un rango de hasta 200 metros. Creando así una representación en 3D del entorno. Por otro lado, las condiciones meteorológicas pueden afectar su desempeño.

Hoy en día, existen vehículos que utilizan sensores para alertar al conductor sobre casi cualquier peligro inmediato que pueda aparecer. Como consecuencia de estos avances, se puede asumir que los siguientes pasos en ADAS probablemente se centrarán en alertar al conductor sobre su propio comportamiento y temperamento. En [21], Johnson determina los resultados de la investigación realizada a cerca de las características que se espera que aparezcan en los automóviles durante los próximos cinco años. Las tendencias previstas en ADAS no solo se centran en el entorno del automóvil, sino también en el propio conductor, más específicamente en la fatiga del conductor.

2.5. Inteligencia Artificial

La inteligencia se define como la habilidad mental para llevar a cabo tareas cognitivas complejas, algunas de las cuales involucran resolución de problemas, razonamiento, aprendizaje, conocimiento emocional, planificación o comprensión.

Desde la invención de los ordenadores, los humanos han sido conscientes de la capacidad de estas máquinas para resolver problemas matemáticos y lógicos, así como también han discutido sobre la posibilidad de que los ordenadores muestren un comportamiento inteligente. Este debate comenzó con el seminario Computing Machinery and Intelligence [22] de Alan Turing en 1950, y continúa en la actualidad.

En su artículo, Turing afirma que, cuando se habla de ordenadores e inteligencia, la pregunta más crítica que uno debe hacerse ya no es si una máquina puede "pensar". Prefiere reformular la pregunta a si una máquina puede actuar de manera equivalente a la forma en que actúa un pensador. Muchos autores afirman que la Inteligencia Artificial (IA) nació cuando Turing propuso esa pregunta.

A pesar de las creencias comunes, el campo de la Inteligencia Artificial no se definió hasta 1956 cuando John McCarthy utilizó el término Inteligencia Artificial para describir "la ciencia y la ingeniería de hacer máquinas inteligentes".

El campo de la IA estudia el desarrollo de sistemas inteligentes, que imitan las funciones cognitivas presentes en los humanos, para crear máquinas capaces de actuar de la misma manera que actuaría un pensador. Como se ha comentado anteriormente, la inteligencia engloba desde el aprendizaje o la resolución de problemas hasta habilidades humanas más complejas como puede ser la capacidad de comprender y lidiar con las emociones.

2.5.1. Computación Afectiva

Durante mucho tiempo, la línea que separa los ordenadores y las emociones ha sido un tema de interés en el campo de la Inteligencia Artificial. Aunque los ordenadores no pueden sentir o expresar sus propias emociones, sí presentan la capacidad de aprender a detectar esas emociones de los demás.

La Inteligencia Artificial Emocional, también conocida como Computación Afectiva, es una rama de la Inteligencia Artificial que nació en 1995 de la mano de Rosalind Picard. La Computación Afectiva se define como una rama interdisciplinaria de la Computación, Psicología y Ciencia Cognitiva enfocada en el estudio de sistemas capaces de aprender a reconocer emociones, y adaptar su comportamiento en base a ese conocimiento.

La siguiente lista, procedente de [23], describe algunas de las aplicaciones de Computación afectiva hasta el día de hoy, así como algunos ejemplos de casos de uso reales.

- Marketing: Comunicaciones, investigación y optimización de contenidos.
- Servicio al cliente: redirección inteligente de llamadas, recomendaciones de llamadas.
- Recursos humanos: contratación, formación de empleados, seguimiento de la satisfacción de los empleados.
- Cuidado de la salud: atención al paciente, diagnóstico médico, asesoramiento.
- **Asistencia a la conducción: seguridad, rendimiento de conducción autónoma.**
- Educación: medir la eficacia, apoyar a los niños autistas.
- Gobierno: comprensión de la población, seguimiento de las reacciones de los ciudadanos.

Como se puede apreciar, la asistencia en la conducción se incluye entre una de las aplicaciones de la Computación Afectiva. Las mejoras en la experiencia del conductor, así como en la seguridad de los pasajeros, es un área activa de investigación en esta rama de la Inteligencia Artificial.

2.5.2. Aprendizaje Automático

Existe una percepción generalizada de que el aprendizaje automático y la inteligencia artificial forman parte de diferentes áreas de estudio. La perspectiva correcta para comprender la relación entre estas áreas se puede inferir de la Figura 2.6.

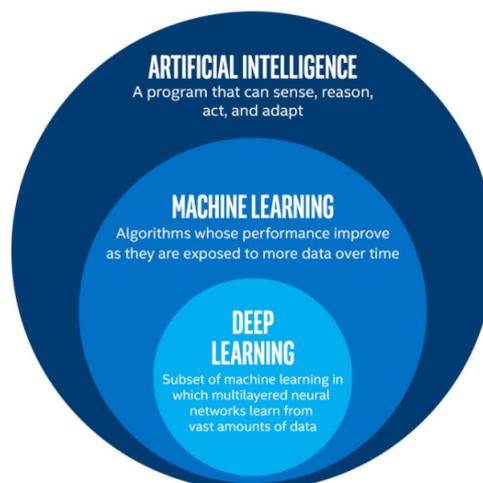


Fig. 2.6. Áreas de la Inteligencia Artificial [24]

Como se puede observar en la Figura 2.6, Machine Learning, o Aprendizaje Automático en español, es una aplicación de la Inteligencia Artificial. Este área de la IA proporciona algoritmos y métodos que pueden aprender y mejorarse por sí mismos, por lo que persigue el objetivo de la Inteligencia Artificial de crear máquinas inteligentes. Cuando esos algoritmos tienen alguna estructura característica en común, forman un clúster dentro de las técnicas de aprendizaje automático, comúnmente conocido como aprendizaje profundo o Deep Learning.

2.5.2.1. Tipos de Aprendizaje Automático

Existe una gran cantidad de algoritmos que se pueden utilizar para el aprendizaje automático. Estos se pueden dividir en tres categorías según la forma en que adquieren el conocimiento: aprendizaje supervisado, aprendizaje no supervisado y aprendizaje reforzado.

Los programadores proporcionan a los modelos supervisados instancias etiquetadas. Esto significa que los algoritmos conocen de antemano la solución del problema que están tratando, por lo que pueden ajustar y modificar sus parámetros en función de ese conocimiento. Por otro lado, al utilizar el aprendizaje por refuerzo, las máquinas no tienen esa ventaja, y no conocen la solución del problema. Por ello, realizarán un enfoque de prueba y error para llegar a la solución. El tercer tipo de aprendizaje no utiliza datos etiquetados ni un enfoque de prueba y error, el aprendizaje no supervisado tiene como objetivo encontrar patrones en los datos sin tener información explícita de ellos. La figura 2.8 [25] simplifica estos conceptos al asociarlos con ejemplos de la vida real.

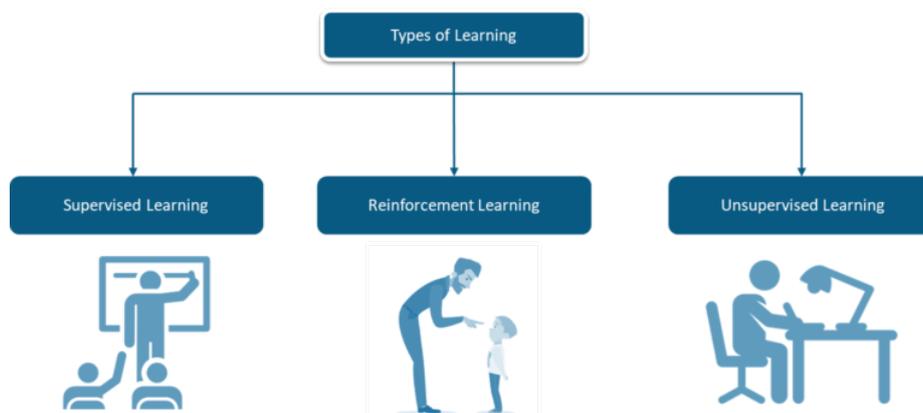


Fig. 2.7. Tipos de Aprendizaje Automático [25]

En este trabajo se utilizará un aprendizaje supervisado para entrenar los modelos de Aprendizaje Automático desarrollados. Esto implica que todos los datos utilizados, independientemente de su naturaleza, tendrán una etiqueta que contiene la solución al problema planteado, es decir, el valor de la clase a predecir.

Dentro del Aprendizaje Automático supervisado, existirán dos tipos de técnicas que se podrán aplicar sobre los datos: la clasificación y la regresión. Partiendo de un conjunto de datos, que estarán definidos por una serie de atributos, de determinará uno de estos atributos como la clase a predecir. Esta clase a predecir puede estar compuesta por datos de naturaleza categórica, o de naturaleza continua. Cuando se aplique un algoritmo de AA a los datos, se utilizará la técnica de clasificación si los datos son categóricos. En caso contrario, se utilizará la técnica de regresión.

Finalmente, una vez definido el tipo de aprendizaje a realizar, y determinado la naturaleza de la clase a predecir, lo datos deberán ser divididos en tres conjuntos diferentes. El algoritmo elegido para realizar el aprendizaje automático se entrenará con los conjuntos de entrenamiento o *train* y validación o *validation*. Una vez se ha entrenado el algoritmo, se utilizará el tercer conjunto de datos, llamado *test*, para evaluar el algoritmo. El algoritmo determinará unas predicciones para los datos de test, estas serán posteriormente comparadas con sus etiquetas, y de esta manera se determinará como de bueno es el modelo que se ha desarrollado.

2.5.2.2. Ciclo de vida del Aprendizaje Automático

En cualquier proyecto de Inteligencia Artificial, que involucre Machine Learning, se deberán seguir una serie de pasos. Estos pasos se pueden encontrar en la Figura 2.8.

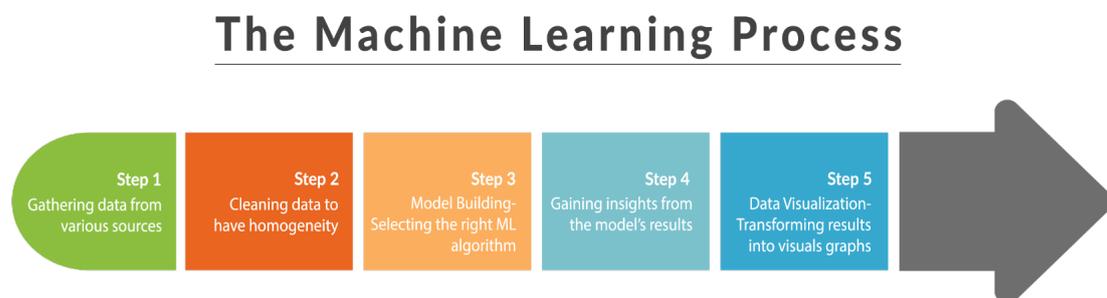


Fig. 2.8. Ciclo de vida del Aprendizaje Automático [26]

- El primer paso consiste en recopilar los datos de fuentes confiables.
- A continuación, los datos deben de limpiarse y amoldarse para cumplir con los requisitos de los algoritmos de AA que se utilizarán en el proyecto.
- En el siguiente paso el programador debe decidir qué algoritmo utilizará para resolver el problema, esto puede requerir algunos experimentos iniciales donde los algoritmos se comparen entre sí. Una vez decidido, se puede comenzar a construir el modelo y entrenarlo.
- Los resultados brindan al desarrollador información relevante sobre cómo se puede mejorar el modelo, si es posible, o si se debe probar algún otro algoritmo.
- El último paso consiste en documentar y presentar los resultados obtenidos.

2.5.3. Redes de Neuronas Artificiales

2.5.3.1. Los primeros modelos

Las redes de neuronas son un tipo de modelo de Machine Learning, en español, Aprendizaje Automático. Estas estructuras son normalmente utilizadas en el reconocimiento de patrones, donde han demostrado proporcionar resultados muy superiores a los de otros algoritmos. Gracias a los recursos computacionales de esta década, los cuales han permitido almacenar ingentes cantidades de datos tanto en memoria sólida como en memoria volátil, las redes de neuronas han experimentado un reciente auge.

La primera referencia a estas estructuras fue en 1943, cuando Warren McCulloch y Walter Pitts modelaron una red con circuitos eléctricos. Sin embargo, la investigación de estos modelos se quedó estancada durante varias décadas, fue en los años siguientes a la conferencia de Dartmouth, cuando se creó la primera y más sencilla red de neuronas, el perceptrón simple (Figura 2.9), la cual sigue en uso a día de hoy.

El perceptrón simple, invención de Frank Rosenblatt, es una red de neuronas formada por una sola neurona capaz de resolver problemas lineales de clasificación binaria. Como se puede observar en la Figura 2.9, esta estructura está formada por unas entradas, a las cuales se les aplicarán unos pesos, y a cuya suma se le aplicará una función de activación. Esto proporcionará una salida, si la salida no es la esperada, es decir, no se corresponde con la etiqueta asociada, se recalcularán los pesos.

En 1959, Bernard Widrow and Marcian Hoff de Stanford University desarrollaron el ADAPtative LINEal Model conocido como ADALINE. Este modelo, también en uso a día de hoy, difiere del Perceptrón, en que al contrario de este no sólo te indica si una instancia está mal clasificada, sino que también proporciona un valor continuo de error que indica cuánto difieren el valor obtenido y el valor real.

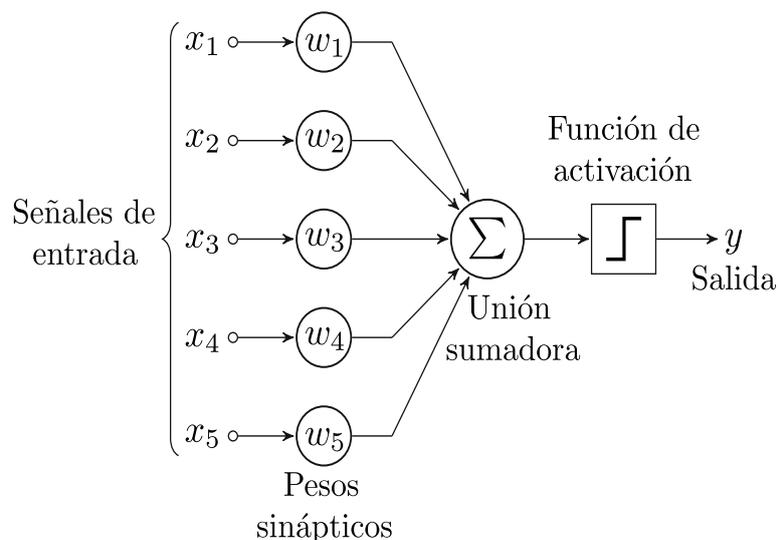


Fig. 2.9. Perceptrón simple [27]

2.5.3.2. Clasificación y regresión no lineal

No fue hasta 1975, cuando se desarrolló la primera red multicapa. Esta junto con otras innovaciones, como la retro-propagación del error, y la resolución de problemas tanto lineales como no lineales, han provocado una evolución en las redes de neuronas hasta formar las estructuras que se utilizan a día de hoy.

En la Figura 2.10 podemos observar dos ejemplos de un clasificador, es decir, un modelo de AA que utilizará los atributos presentes en los datos para predecir una clase. El de la imagen izquierda realiza una operación lineal, será capaz de realizar clasificaciones utilizando un hiper-plano. Para poder utilizar estos clasificadores, es necesario que los datos sean linealmente separables, sin embargo, en el caso de la imagen de la derecha se puede observar que el modelo se adapta y proporciona una solución no lineal al problema.

De esta manera, un modelo no-lineal podrá resolver una gama de problemas considerablemente más amplia que un modelo lineal. Gracias a la incorporación de funciones de activación no lineales en las redes de neuronas, se produjo un auge, y estos modelos se convirtieron en uno de los algoritmos más usados para resolver problemas de Aprendizaje Automático.

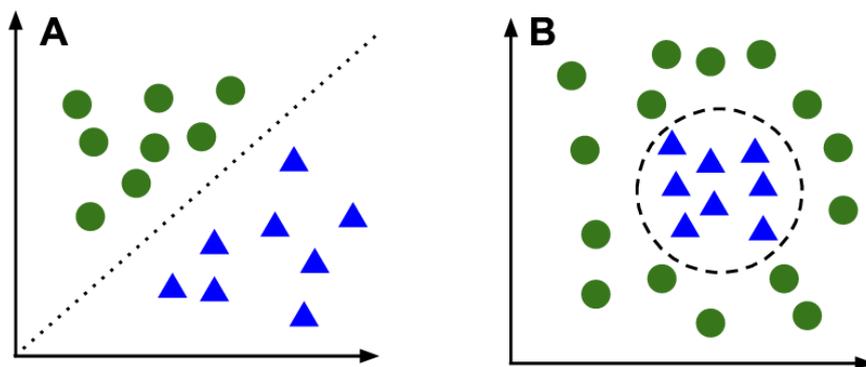


Fig. 2.10. Clasificación lineal y no lineal [28]

2.5.3.3. Aprendizaje en Redes de Neuronas Artificiales

Las redes de neuronas artificiales se inspiran en la imitación del funcionamiento del cerebro humano. El cerebro está formado por billones de neuronas interconectadas entre sí, que se activan en base a señales recibidas. De la misma manera, las redes de neuronas artificiales están compuestas por capas de neuronas conectadas, estas conexiones están definidas por un peso. Además de los pesos, en la Figura 2.11 se encuentran las diferentes características que definen una red de neuronas. Estas serán definidas con detalle más adelante, en el apartado relativo a redes de neuronas convolucionales de este trabajo.

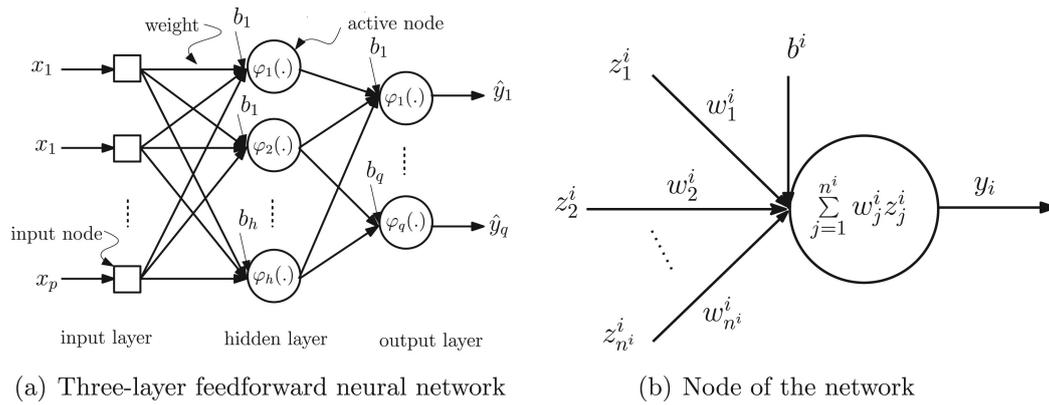


Fig. 2.11. Modelo de Redes de Neuronas [29]

El funcionamiento interno general de una red es sencillo y cambia dependiendo de la técnica que se utilice. En las redes de neuronas artificiales, se podrán utilizar tanto aprendizaje supervisado como no supervisado.

En caso de que el problema se resuelva utilizando una estrategia de aprendizaje supervisado, la red recibe instancias previamente etiquetadas, estas instancias están definidas por atributos y cada uno de ellos será la entrada a una de las neuronas de entrada presentes en la primera capa.

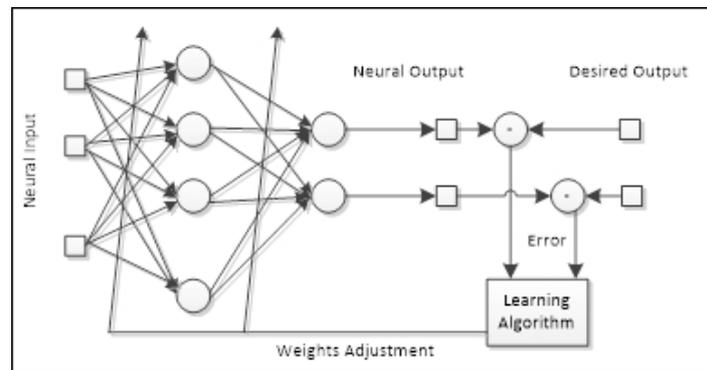


Fig. 2.12. Aprendizaje de una red de neuronas artificial [30]

Un ejemplo de este proceso se puede encontrar en la Figura 2.12. Cuantas más iteraciones se realicen sobre los ejemplos, la red ajustará sus pesos de manera más precisa a los ejemplos que se le han propuesto en el aprendizaje. Si se continúa enseñando indefinidamente a una red un conjunto de datos, cuando ya ha aprendido previamente la definición del resultado, podría producirse un sobre-ajuste de la red. Una red esta sobre-ajustada cuando no consigue generalizar un problema, es decir, cuando define sus resultados en base a patrones que no forman parte del problema a resolver, sino que son características inherentes en los datos que ha aprendido.

2.5.3.4. Redes de Neuronas Convolucionales: Deep Learning

En la actualidad, a la hora de abordar problemas de aprendizaje automático basados en imágenes, siempre es interesante tener presente el uso de técnicas de Deep Learning. Estas redes de neuronas complejas han demostrado ser muy eficaces en el reconocimiento, tanto de objetos, como de personas.

Se entiende como Deep Learning o aprendizaje profundo a los modelos de aprendizaje automático, refiriéndonos generalmente a las redes de neuronas, los cuales poseen la capacidad de realizar aprendizaje de características. Estos permiten introducir al modelo directamente una imagen, sin tener que llevar a cabo una extracción de características como en las redes de neuronas artificiales tradicionales.

Las Redes de Neuronas Convolucionales, o Convolutional Neural Networks (CNN) en inglés, es una de las técnicas más utilizadas en los últimos años para abordar problemas con imágenes. Al contrario de otras arquitecturas de redes de neuronas más simples, las CNN forman parte del conjunto de modelos de aprendizaje profundo, realizando de manera interna la extracción de características de los datos. La extracción de características es un proceso que incluye a un experto en la materia, el cual asesorará sobre qué atributos son los más relevantes en la imagen y cómo extraerlos. Gracias al uso de las redes de neuronas convolucionales, no se necesitará de dicho perfil para desarrollar el modelo.

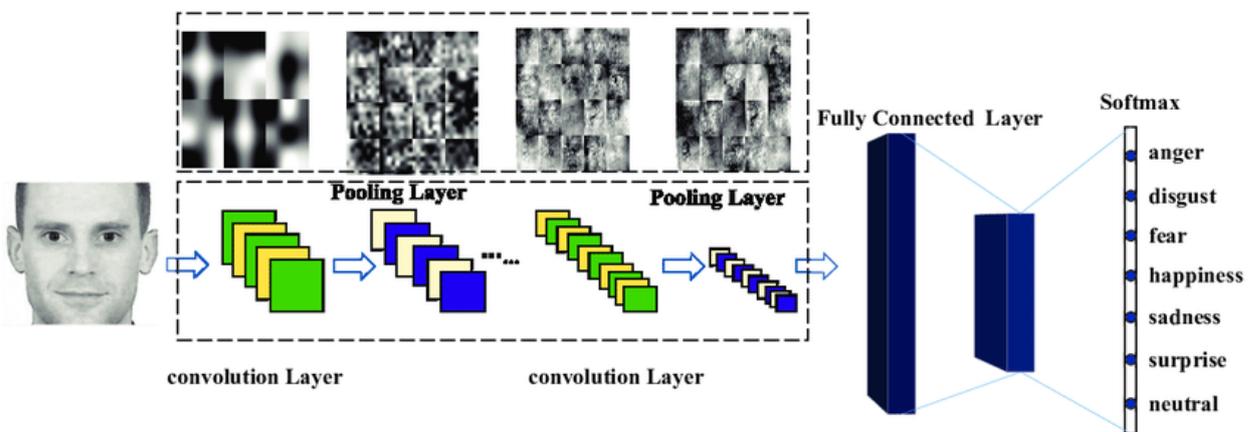


Fig. 2.13. Ejemplo de la arquitectura de una red convolucional [31]

La Figura 2.13 muestra una arquitectura básica de una red convolucional. Las primeras capas de la red, están formadas por capas de convolución, las cuales aplicarán diferentes filtros a la imagen, estos filtros representan el equivalente a la extracción de características. Tras cada capa de convolución, se encuentra una capa de muestreo, la cual se encarga de reducir el tamaño de cada una de las características que se han producido en la capa de convolución.

Como se puede observar en la Figura 2.13, al final de las capas de convolución, se introduce una red de neuronas tradicional, la que recibirá como entrada las características que han sido extraídas de manera automática por las capas de convolución, y determinará

cuál es la clase a la que pertenece la imagen introducida.

2.5.3.5. Selección de hiper-parámetros y arquitectura

Si bien estas redes proporcionan muchos aspectos positivos, dado que se hace una extracción de las características mucho más compleja y precisa que de la manera tradicional, también presentan algunas desventajas respecto a las redes de neuronas simples. Algunos de los inconvenientes de las redes convolucionales rebosa en la elección de la arquitectura de la red.

La elección de los parámetros que definen un modelo de red de neuronas convolucionales, es un problema donde su dificultad recae en lo amplio que es el espacio de posibilidades. Se puede dividir las características de una red convolucional en sus hiper-parámetros y su arquitectura.

La arquitectura de una red convolucional está definida por los siguientes componentes:

- Número y posición de capas convolucionales
- Función de activación de las capas convolucionales
- Número de filtros en cada capa convolucional
- Tamaño del filtro de convolución
- Número y posición de capas de sampling
- Número y posición de capas de dropout
- Número de capas ocultas de la red de neuronas
- Número de neuronas en las capas ocultas

Como se puede observar, las posibles combinaciones de parámetros son infinitas, por lo tanto, también lo será el número de posibles arquitecturas. Dado que no existe una regla para determinar qué arquitectura debe ser utilizada, el proceso a seguir consiste en la elección de una o varias arquitecturas base, y la variación informada de los componentes que la describen.

En las redes tradicionales, la elección del número de capas ocultas de la red sigue siendo un problema con un espacio de posibilidades infinito acotado por los recursos computacionales utilizados, donde la prueba y error es la única técnica que se puede utilizar para elegir la estructura de la red. En las redes convolucionales, este problema se multiplica. No solo se tendrá que elegir un número de capas convolucionales para la red, sino que además cada una de ellas podrá tener un número de filtros diferentes.

La elección de la arquitectura correcta es una de las limitaciones encontradas en este tipo de redes de neuronas. A día de hoy, se han aplicado técnicas de búsqueda, así como

algoritmos evolutivos con el fin de proponer una alternativa a la búsqueda a ciegas de una arquitectura funcional. No obstante, la complejidad del espacio de posibilidades, y el tiempo de entrenamiento en estas redes, hace que a día de hoy siga siendo un desafío.

Para este proyecto, se han utilizado técnicas de búsqueda para explorar posibles arquitecturas. Los resultados obtenidos de dichas búsquedas, son en este caso menos precisos si se comparan con los obtenidos a través de la búsqueda a ciegas basada en la literatura de otros proyectos similares.

Junto con la arquitectura, cada experimento tendrá una variación en los hiper-parámetros que definen el modelo. Estas modificaciones forman parte del proceso de la búsqueda del modelo que mejor defina el problema. Generalmente, en un problema de redes de neuronas convolucionales, y de la misma manera en este trabajo, se modifican las siguientes características:

- **Función de activación.** Estará presente en cada una de las neuronas de la red, y definirá no sólo si la neurona ha de activarse, sino también el tipo de problema que podrá ser resuelto. El uso de una función de activación no lineal implicará la capacidad de la red de resolver problemas no lineales.
- **Optimizador.** Este definirá la fórmula utilizada para modificar los pesos y la razón de aprendizaje en la red.
- **Razón de aprendizaje.** Este parámetro controlará cuánto se están modificando los pesos de la red de neuronas artificial.
- **Batch size.** Este concepto indica el número de instancias que serán utilizadas en entrenamiento a la vez. Las instancias se dividirán en grupos de este tamaño y serán introducidas en el modelo por turnos.
- **Épocas.** La introducción de todos los subgrupos creados, en función al batch size, se le denomina época.

Finalmente, como en cualquier modelo de Aprendizaje Automático, es necesario determinar una métrica para poder evaluar el modelo. Una métrica evalúa cómo de bueno es el modelo que está siendo entrenado, y proporciona información sobre cómo evoluciona el proceso de entrenamiento. Para este trabajo, se utilizará la *accuracy* como métrica de evaluación. A lo largo de este trabajo, se nombrará como exactitud o *accuracy* a la métrica que sigue la siguiente fórmula:

$$accuracy = \frac{\text{instancias correctamente clasificadas}}{\text{instancias totales}} * 100 \quad (2.1)$$

2.6. Trabajos similares

Este proyecto, cubre muchos aspectos y temas que otros trabajos abordan de manera individual. Es por esto, que para poder tener una visión general del estado del arte de este trabajo hay que dividir en tres secciones los trabajos similares, ya que en ningún otro trabajo hasta la fecha se ha abordado en conjunto el desarrollo de ambos modelos, tanto de emociones como de comportamiento, para formar un sistema.

2.6.1. Reconocimiento de emociones vía Deep Learning

Si bien existen otros modelos alternativos para resolver el problema del reconocimiento de patrones en imágenes, en la mayoría de los trabajos realizados en este campo los investigadores practican el uso de técnicas de aprendizaje profundo. En este apartado, se hará una revisión sobre los experimentos y resultados obtenidos por otros autores en este campo hasta la fecha.

Uno de los trabajos más recientes en el campo de la detección de emociones utilizando técnicas de Deep Learning, se realizó en 2017 por Shan et al. [32]. En este proyecto, se utilizaron los conjuntos de datos CK+[33] y JAFFE [34], los cuales serán utilizados también para este proyecto, junto con otros conjuntos de imágenes. El modelo propuesto en este trabajo utiliza una red de neuronas convolucional, que será aplicada independientemente a cada uno de los conjuntos de datos. Al igual que en este proyecto, previamente al entrenamiento con la red convolucional, se realiza un preprocesado que incluye la detección de cara, además se realiza una ecualización del histograma en las imágenes. Esta técnica busca unificar los niveles de contraste en una imagen. Finalmente, la *accuracy* alcanzada fue de 76.7 % (JAFFE) y 80.3 % (CK+) respectivamente. La principal diferencia que reside entre el trabajo realizado por Shan et al. y este proyecto, es que para el entrenamiento de la red se unificarán distintas bases de datos y se buscará crear un modelo único que pueda resolver el problema de la detección de emociones para cualquier conjunto de datos.

En [35], se realizó una comparación de algunas de las arquitecturas más famosas en el campo de las redes convolucionales, sobre el conjunto de datos FER2013 [36]. El mejor resultado, obtenido con la arquitectura AlexNet [37] logró conseguir un 64.2 % de *accuracy*.

Al igual que en [35], Ramdhani et al. [38] también proponen el uso del dataset FER2013 [36] para la creación de un modelo de reconocimiento de expresiones faciales. Este trabajo realiza un estudio utilizando no sólo FER2013 sino también su propio conjunto de datos creado específicamente para el proyecto. El estudio realizado investiga el uso de diferentes batch size, comparando la *accuracy* obtenida alternando dicho parámetro para dos arquitecturas diferentes. Los mejores resultados del trabajo, se obtuvieron con un batch size de tamaño ocho, utilizado en su propio dataset, con un valor de *accuracy* de 73.8 %. Para FER2013 su mejor resultado es de 69.1 % utilizando un batch size de 128.

Otro trabajo referente fue realizado en 2019 por Pranav et al. [39]. En este se utilizó únicamente el conjunto de datos FER2013, y se realizó una comparativa entre varios modelos de Aprendizaje Automático. Entre los modelos analizados se encuentran las redes de neuronas y dos arquitecturas de redes convolucionales. El mejor modelo, el cual presentó mejores resultados de *accuracy*, fue la red convolucional propuesta, esta red contaba con dos capas convolucionales de 32 y 64 filtros respectivamente. La *accuracy* obtenida fue de un 55.6 %. En comparación con el modelo que se ha obtenido en este trabajo, al igual que en [32], tan solo se crea un modelo para un conjunto concreto de datos.

Uno de los proyectos que logró obtener resultados remarcables fue desarrollado por Nwosu et al. [40] en 2017. Para este trabajo se desarrolló una red convolucional formada por dos canales, donde el primero incluye una imagen de los ojos de una persona, mientras que el segundo canal introduce la expresión producida por la boca. Los resultados obtenidos, sobre el conjunto de datos es de 97.7 % para el conjunto JAFFE y de 95.7 % en CK+.

La principal diferencia que existe entre los proyectos comentados en este apartado, y el modelo que ha sido creado para este trabajo, es que en este trabajo se han incluido un total de cuatro conjuntos de datos diferentes, y se ha evaluado la agregación de todos los conjuntos previamente preprocesados. Esto permite tener más instancias de entrenamiento, y conseguir un modelo versátil que puede detectar imágenes con diferentes patrones independientemente del contexto de la foto analizada.

2.6.2. Reconocimiento de distracciones en conductores vía Deep Learning

De la misma manera que en el apartado anterior, en esta se hará una revisión de los trabajos que han abordado el problema del reconocimiento de distracciones en conductores, utilizando técnicas de redes de neuronas convolucionales. Al contrario que en el estudio de las emociones, existe un número inferior de investigadores que han tratado estos datos, y por lo tanto, las publicaciones son menos frecuentes.

Uno de los proyectos a destacar es el realizado por Torres et al. en 2017 [41]. Su modelo de redes convolucionales estaba formado por tres capas de convolución de 32, 32 y 64 filtros respectivamente, y una red de neuronas al final del modelo con una capa oculta de 64 neuronas. Este proyecto alcanzó un 99 % de *accuracy* en el conjunto de datos de conductores distraídos State Farm [42], reducido a cinco clases de las nueve que contiene originalmente el conjunto de datos.

Por otro lado, el estudio realizado por Baheti et al. [43] presenta una reducción de la arquitectura VGG-16 [44]. Esta reducción consigue limitar a 15 millones el número de parámetros que contiene la red convolucional, una reducción considerable dado que la arquitectura original consta de 140 millones de parámetros. Esto permite una mayor velocidad a la hora de entrenar y de predecir la clase, permitiendo realizar estudios en tiempo real. La *accuracy* alcanzada por esta arquitectura es de 95.5 % en el conjunto de

datos de conductores distraídos State Farm.

Finalmente, el trabajo realizado en 2018 por Sathe et al. [45] propone un sistema de redes convolucionales, que además usa la técnica de *data augmentation* para la preparación previa de los datos. Gracias a esta técnica, el proyecto consigue aumentar y variar las imágenes utilizadas procedentes del conjunto de datos de conductores distraídos State Farm [42] la *accuracy* obtenida por Sathe y su equipo en este proyecto es de 96.7 %.

2.6.3. ADAS basados en el conductor

En este trabajo se desarrollarán dos modelos que formarán un prototipo para un sistema de ayuda a la conducción basado en la monitorización del conductor. En este apartado se comentarán algunos sistemas existentes de ayuda que también se basan en el análisis del conductor a través de imágenes.

El sistema más valorado, y similar al propuesto en este trabajo es el desarrollado por la empresa Afectiva [46]. Este proyecto es un sistema de ayuda a la conducción que es capaz de detectar emociones como el nivel de distracción, así como aspectos relevantes del conductor como el nivel de somnolencia. En la Figura 2.14 [47] se puede ver un ejemplo del resultado del sistema de Inteligencia Artificial creado por la empresa.

Este sistema ha sido creado utilizando más de 6.5 millones de vídeos espontáneos de conductores, y aunque inicialmente reconocía tan solo aspectos de las emociones del conductor, a día de hoy esta evolucionando para ofrecer aspecto más avanzados como el nivel de somnolencia o el estado de los ojos del conductor.

El proyecto de la empresa Afectiva, propone la creación un sistema de ayuda a la conducción que combina el estado anímico del conductor con otros aspectos relativos al comportamiento. Este sistema es el más similar, a día de hoy, al que se propone en este proyecto. Sin embargo, en este trabajo se va un paso más allá, y no solo se detectará la somnolencia, sino que también se incluyen otros comportamientos que puedan poner en estado de riesgo el coche.

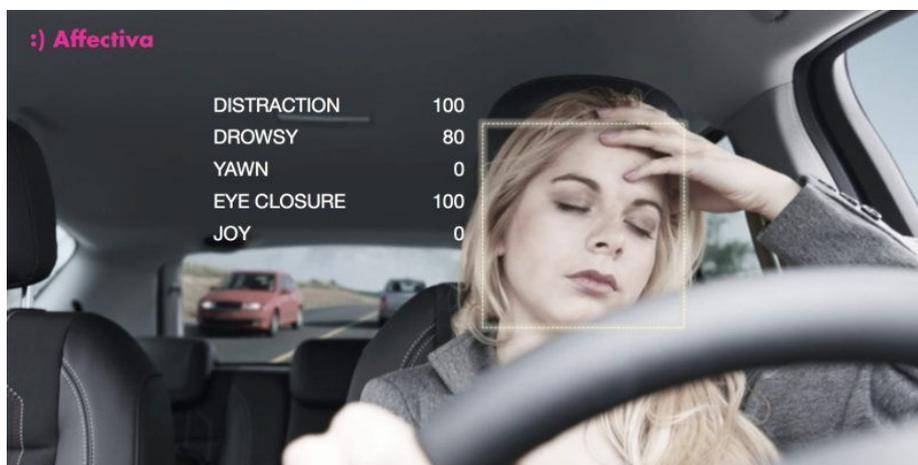


Fig. 2.14. Deep Dive Afectiva Automotive AI [47]

2.7. Aportaciones al Estado del Arte

Esta sección pretende situar este trabajo en el estado del arte que se ha desarrollado durante el capítulo. Teniendo en cuenta los avances actuales, y los sistemas existentes, se presenta cómo el sistema propuesto resulta ser innovador y una posible solución para un problema del mundo real.

Numerosos estudios han sido realizados en cuanto a las emociones, y a su capacidad de ser detectadas mediante patrones visuales. Actualmente, existen modelos de inteligencia artificial capaces de detectar dichos patrones, y por lo tanto predecir una emoción en base a una imagen. Sin embargo, aunque si se han estudiado aplicaciones de estos modelos, nunca han sido combinados para crear un sistema más complejo e informado.

En este trabajo, se da un paso más allá, y se propone la combinación de las emociones, con la combinación de las acciones en un conductor. Dado que estos dos aspectos han sido estudiados y han demostrado ser clave a la hora de determinar la posibilidad de un accidente en la carretera, la creación de un sistema que combine en conjunto dicha información podría resultar en una ayuda crucial a la hora de evitar accidentes de tráfico.

La combinación de ambos modelos y su desarrollo, consigue crear un sistema de ayuda a la conducción basado en el conductor, que va más allá de los sistemas que se encuentran actualmente en el mercado, o incluso de los prototipos surgidos producto de investigación. Se propone una aplicación directa de la computación afectiva y el Deep Learning a la resolución de un problema del mundo real.

Además de las aplicaciones del sistema en el mundo real para la prevención de accidentes, este también supondrá un avance en la forma de trabajar en áreas como el de la psicología de tráfico, ya que permitirá a los investigadores una nueva forma de reunir datos y de analizar situaciones. Teniendo la información de comportamientos típicos en un sujeto, obtenidos mediante el sistema, se podrían realizar estudios extensos y automatizados en la búsqueda de patrones y de comportamientos conflictivos.

Finalmente, la posibilidad de detectar comportamientos también supone una oportunidad para empresas de taxistas o conductores como UBER. Gracias a la detección de la acción realizada, se puede proveer un servicio mucho más seguro y eficaz si se incluye un sistema de detección como el propuesto. El sistema puede alertar al mismo conductor cuando su comportamiento o sus acciones no sean acordes a los valores de la empresa, para evitar tanto situaciones desagradables a los pasajeros o clientes del servicio como accidentes no intencionados.

3. DISEÑO Y ANÁLISIS DEL SISTEMA

3.1. Arquitectura del sistema

El diseño consta de cuatro componentes principales. Entre ellos se encuentran la cámara que tomará las imágenes del conductor, los dos modelos de Deep Learning, que serán encargados de determinar tanto la emoción presente en el conductor como la acción que esta ejecutando en cada momento, y finalmente, el sistema de decisión que se encargará de determinar si se debe tomar una acción ante la situación presente en el vehículo.

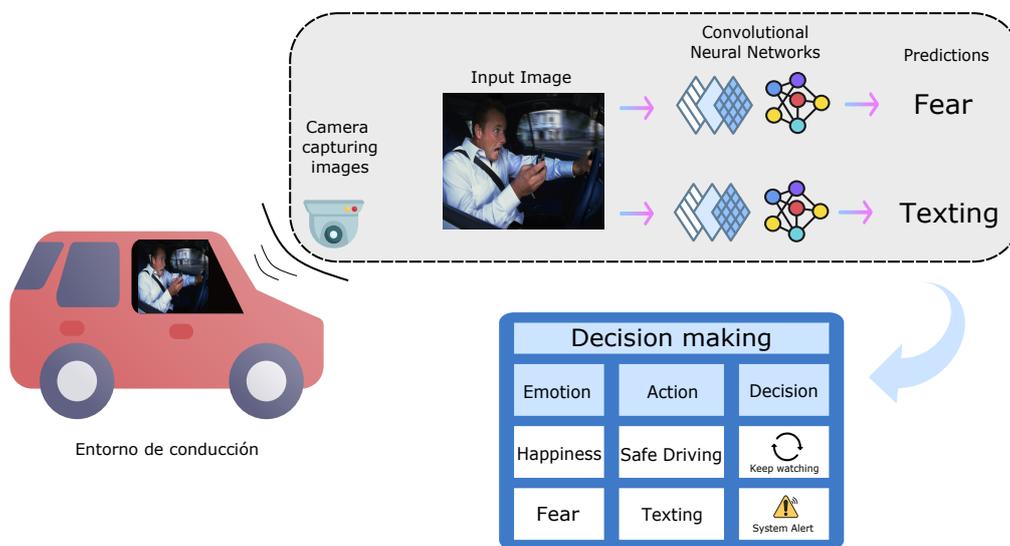


Fig. 3.1. Arquitectura del sistema

En la Figura 3.1 se puede observar un ejemplo de cómo actuaría el sistema en un entorno del mundo real. A través de una cámara, situada en el interior del coche, se toman continuamente imágenes que serán procesadas e introducidas como entrada a ambos modelos de Aprendizaje Automático.

Los modelos producirán, independientemente uno del otro, un resultado respecto a la imagen. El primer modelo se encargará de analizar y predecir la posible emoción del conductor, mientras que el segundo se encargará de la acción que este realizando. Una vez se han obtenido los resultados, estos se procesan mediante un sistema de decisión basado en reglas.

Este sistema contemplará todas las posibilidades respecto a las variables de las clases a predecir en ambos modelos. En la Figura 3.1 se pueden ver dos ejemplos concretos del sistema, si se detecta una acción que podría calificarse como peligrosa, en el ejemplo el conductor muestra temor y está distraído, entonces se propone que el coche tome una acción, en este caso una alerta. Por el otro lado, si las variables dan lugar a una acción que

no contempla ningún peligro, acompañado por un estado anímico favorable, entonces el sistema no realiza ninguna acción y continuará monitorizando al conductor.

3.2. Aspectos abordados

Para este proyecto, se propone el diseño definido en la Figura 3.1, habiéndose desarrollado únicamente la parte del sistema que está encuadrada en el recuadro gris de la Figura 3.1.

El desarrollo del sistema de toma de decisiones requiere conocimiento extra por parte de expertos, que puedan valorar y determinar situaciones de riesgo, y no se encuentra en el alcance de este proyecto. Sin embargo, la investigación realizada sobre la conducción, los comportamientos, y las emociones, sí que permiten realizar una propuesta informada sobre aspectos que potencialmente se podrían incluir en el diseño del sistema. Aunque en este trabajo no se determinen cuales son combinaciones de emoción-acción que deban desencadenar una acción por parte del sistema, sí se propone el uso de alertas sonoras, cada vez que se dé comportamiento peligroso.

En este trabajo se han abordado las tres primeras fases de la arquitectura del sistema. Los componentes desarrollados son los siguientes:

- Modelo para el reconocimiento de emociones: Capítulo 4
- Modelo para el reconocimiento de distracciones: Capítulo 5
- Integración para pruebas en tiempo real con una cámara: Capítulo 6

Con el fin de desarrollar el sistema propuesto, capaz de detectar emociones y comportamientos en un conductor, es necesario crear dos modelos diferentes, entrenados de manera independiente. El primer modelo abarcará únicamente el problema del reconocimiento de emociones, mientras que el segundo modelo será capaz de detectar ciertas acciones, ambos están explicados de manera independiente en los siguientes capítulos del proyecto.

Es importante mencionar que, aunque los modelos sean creados y evaluados por separado, ambos han sido pensados para formar parte del mismo sistema. Es por eso que muchas decisiones de diseño que se encuentran en algunos de los siguientes apartados, respecto a la elección de las imágenes o las arquitecturas evaluadas, se han tomado viendo el problema como una unidad y siempre teniendo en cuenta que el modelo formará parte de un sistema.

3.3. Tecnologías utilizadas

Para la elaboración de este proyecto, se han hecho uso de librerías, propias del lenguaje Python, para el desarrollo de Machine Learning y el análisis de datos. A continuación, se comentarán los diferentes softwares utilizados para el desarrollo del sistema:

- Keras y Tensorflow. Estas distribuciones han sido utilizadas para la implementación de las redes de neuronas convolucionales. Las librerías proporcionan la infraestructura necesaria para poder crear diferentes arquitecturas de redes convolucionales en Python.
- Pandas. Esta librería en Python permite el manejo de grandes cantidades de datos, permitiendo crear tablas y estructuras intermedias para poder almacenarlos.
- OpenCV. Esta librería ha sido utilizada para el preprocesado de las imágenes, pudiendo utilizar a través de ella funciones como la de corte o redimensionado de imágenes. Además, también ha sido utilizada para llevar a cabo la conexión entre la cámara y el sistema en Python. Por otro lado, para la detección de cara en una imagen se ha utilizado el modelo de detección de caras implementado en esta librería.
- Comet ML. Esta herramienta ha sido utilizada para almacenar todos los experimentos realizados durante este trabajo. Este software ha proporcionado una interfaz que al conectarse con el código del sistema facilita el proceso de experimentación.

3.3.1. Proceso de experimentación

El desarrollo de un modelo de Machine Learning conlleva un proceso de experimentación muy elaborado en el que varias técnicas y estrategias son evaluadas y analizadas. Para poder llevar a cabo dicho proceso de experimentación se ha utilizado la herramienta Comet ML. En este apartado se desarrollará cómo ha sido esta herramienta usada, así como la organización utilizada para evaluar los experimentos y almacenarlos.

Para este trabajo, dado que se desarrollarán dos modelos diferentes, se han creado dos proyectos independientes en la plataforma, uno para cada modelo. En la Figura 3.2 se puede observar la interfaz de la plataforma Comet ML. Cada experimento realizado ha sido almacenado, al igual que sus características más importantes. Entre estas características estarán los datos utilizados para cada experimento, los hiperparámetros o el tiempo de duración. Además, en la parte superior se pueden observar dos gráficos con la evolución de todos los experimentos, para tener así una visión global de la evolución del modelo.

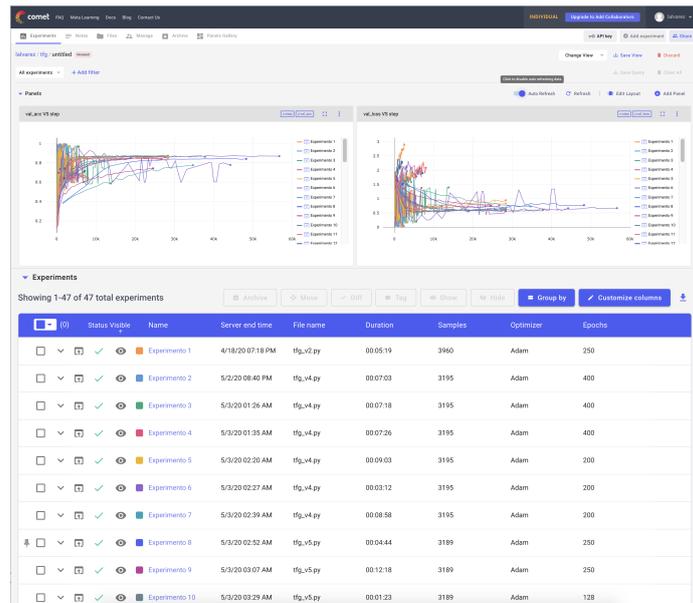


Fig. 3.2. Experimentos en Comet ML

Por cada experimento realizado, se almacena información relevante de dicho experimento. Esto es vital para poder analizar los resultados y determinar que nuevos experimentos deben ser realizados. En la Figura 3.3 se puede ver un ejemplo de un experimento donde se han almacenado los filtros convolucionales de cada una de las capas que proporcionó la red utilizada, es decir, las características extraídas para el entrenamiento.

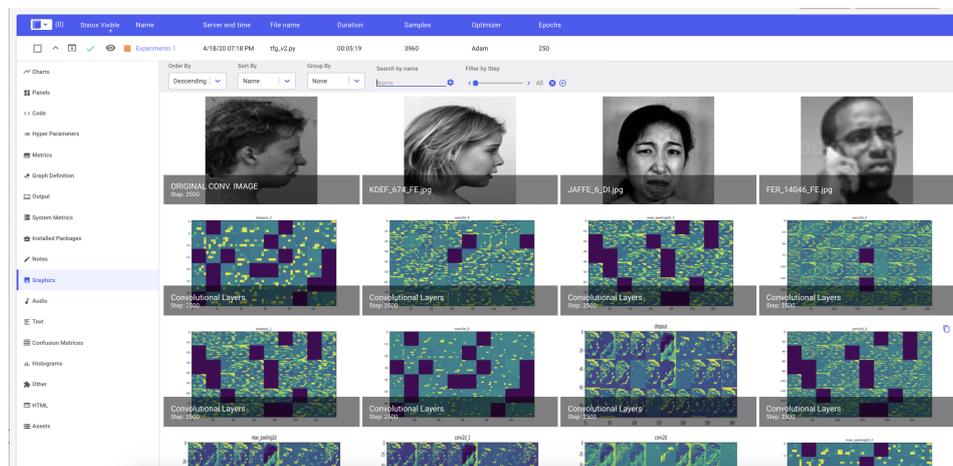


Fig. 3.3. Almacenamiento de filtros en Comet ML

De la misma manera, en la Figura 3.4 se puede observar otro experimento, donde se almacenaron los ejemplos que habían sido mal etiquetados por el modelo, esta técnica ha sido usada a lo largo de toda la experimentación. Cada imagen tiene como nombre la emoción que determinó el modelo seguido de la emoción con la que está etiquetada la imagen. De este modo se puede hacer un seguimiento de los ejemplos que no se han aprendido, detectar imágenes que han sido mal etiquetadas, y encontrar patrones que ayudarán a tomar decisiones que serán aplicadas en futuros experimentos.

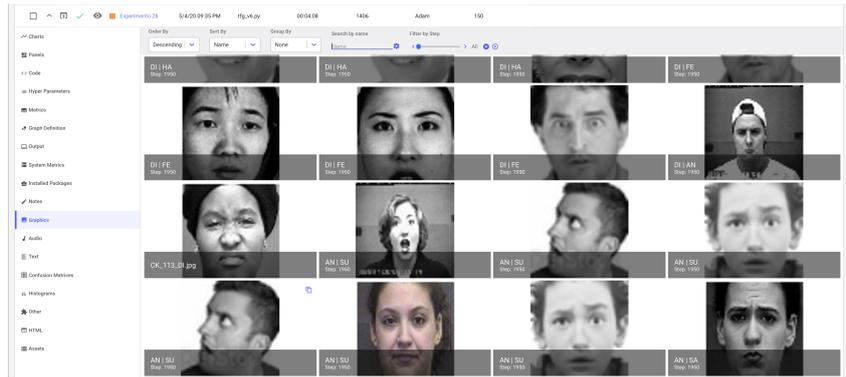


Fig. 3.4. Instancias mal clasificadas almacenadas en Comet ML

Finalmente, esta plataforma permite almacenar más información relevante, como la matriz de confusión resultado de cada análisis o el código que ha sido utilizado en cada experimento, un factor vital para poder repetir experimentos en el futuro.

En la Figura 3.5 se puede observar una matriz de confusión como la que se ha generado para cada experimento. Esta indica cuantas instancias de cada clase han sido etiquetadas correctamente para cada clase, situándose en las filas la clase real, respecto a la clase que ha sido predicha por el modelo situada en las columnas.

Mediante la observación de la matriz de confusión, junto con las imágenes almacenadas, y los diferentes parámetros utilizados, se toma una decisión sobre cómo de bueno es el modelo y que parámetros deberían modificarse para el siguiente experimento.

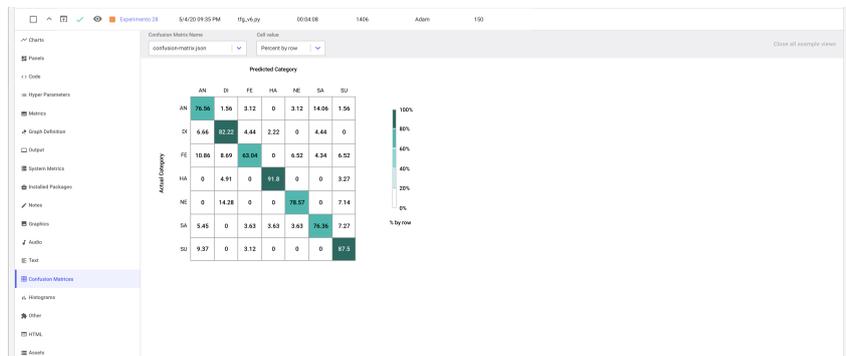


Fig. 3.5. Matriz de confusión en Comet ML

El uso de esta herramienta, ha facilitado la organización y estructuración del proyecto, pudiendo así haber realizado numerosos experimentos que serán descritos en los siguientes capítulos del proyecto. Todos los experimentos que serán comentados en este trabajo, están almacenados en dicha plataforma, conteniendo toda la información, recién comentada, relevante de los mismos.

3.4. Especificación de requisitos

El sistema final implementado a través de técnicas de aprendizaje profundo, deberá poseer unas características determinadas que acrediten su funcionalidad y exactitud. Dichos requisitos del sistema, serán determinados en este apartado del documento.

RZ-XX			
Descripción			
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias			

TABLA 3.1. PLANTILLA DE TABLA DE REQUISITOS

Los requisitos se presentarán en formato de tabla. La Tabla 3.1 muestra un ejemplo de dicha plantilla que será usada para determinar los requisitos tanto funcionales como no funcionales. La plantilla consta de cinco campos diferentes, que serán los siguientes:

- **Identificador:** El primer campo, situado en la parte superior de la tabla, corresponde al identificador del requisito. Este campo tendrá un valor único para cada requisito del sistema, y estará compuesto por las siglas RZ-XX donde:
 - La letra Z indicará el tipo de requisito. Podrá tomar los valores F (Funcional) o NF (No funcional)
 - Las letras XX determinarán el número de requisito.
- **Descripción.** Este campo contendrá un breve resumen de los objetivos y características del requisito.
- **Necesidad.** Se especifica la valoración que se le da a un requisito.
- **Prioridad.** Indica el nivel de importancia del requisito.
- **Complejidad.** Indica la dificultad que conlleva el desarrollo de dicho requisito.
- **Estabilidad.** Determina la probabilidad de cambio del requisito, cuanto mayor sea la estabilidad más improbable será que el requisito sea modificado.
- **Dependencias.** Este campo indica si el requisito depende de otro requisito para su realización.

Como se puede observar en la Figura 3.1, para determinar el valor que cierto requisito toma en cada uno de los campos, este valor se marcará con color en la tabla.

3.4.1. Requisitos Funcionales

Los requisitos que se incluyen en este apartado representan las funcionalidades que ofrecerá el sistema definido. Entre estas funcionalidades, se incluirán las relativas a la precisión y características propias del sistema.

RF-01			
Nombre	Conexión con cámara		
Descripción	El sistema deberá recoger imágenes a través de una cámara.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias			

TABLA 3.2. REQUISITO FUNCIONAL RF-01

RF-02			
Nombre	Procesado de imagen		
Descripción	El sistema procesará una imagen, para utilizarla como entrada a los modelos de aprendizaje automático.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias			

TABLA 3.3. REQUISITO FUNCIONAL RF-02

RF-03			
Nombre	Predicción sobre imagen		
Descripción	El sistema realizará una predicción, sobre la imagen introducida como entrada, basada en el modelo de aprendizaje automático entrenado.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias	RF-02		

TABLA 3.4. REQUISITO FUNCIONAL RF-03

RF-04			
Nombre	Detección de caras		
Descripción	El sistema detectará las caras de los sujetos presentes en las imágenes de entrada.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias			

TABLA 3.5. REQUISITO FUNCIONAL RF-04

RF-05			
Nombre	Detección conductor irritado		
Descripción	El sistema determinará si el usuario presente en la imagen muestra signos de irritabilidad o enfado, a través de sus rasgos faciales.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias	RF-02, RF-03, RF-04		

TABLA 3.6. REQUISITO FUNCIONAL RF-05

RF-06			
Nombre	Detección conductor contento		
Descripción	El sistema determinará si el usuario presente en la imagen muestra signos de felicidad, a través de sus rasgos faciales.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias	RF-02, RF-03, RF-04		

TABLA 3.7. REQUISITO FUNCIONAL RF-06

RF-07			
Nombre	Detección conductor sorprendido		
Descripción	El sistema determinará si el usuario presente en la imagen muestra signos de sorpresa, a través de sus rasgos faciales.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias	RF-02, RF-03, RF-04		

TABLA 3.8. REQUISITO FUNCIONAL RF-07

RF-08			
Nombre	Detección conductor asustado		
Descripción	El sistema determinará si el usuario presente en la imagen muestra signos de temor, a través de sus rasgos faciales.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias	RF-02, RF-03, RF-04		

TABLA 3.9. REQUISITO FUNCIONAL RF-08

RF-09			
Nombre	Detección conductor triste		
Descripción	El sistema determinará si el usuario presente en la imagen muestra signos de tristeza, a través de sus rasgos faciales.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias	RF-02, RF-03, RF-04		

TABLA 3.10. REQUISITO FUNCIONAL RF-09

RF-10			
Nombre	Detección conductor con signos de asco o aversión		
Descripción	El sistema determinará si el usuario presente en la imagen muestra signos de aversión ("disgust"), a través de sus rasgos faciales.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias	RF-02, RF-03, RF-04		

TABLA 3.11. REQUISITO FUNCIONAL RF-10

RF-11			
Nombre	Detección conductor neutral		
Descripción	El sistema determinará si el usuario presente en la imagen muestra un estado neutral, a través de sus rasgos faciales.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias	RF-02, RF-03, RF-04		

TABLA 3.12. REQUISITO FUNCIONAL RF-11

RF-12			
Nombre	Detección conducción segura		
Descripción	El sistema determinará si el usuario presente en la imagen está conduciendo de manera segura.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias	RF-02, RF-03		

TABLA 3.13. REQUISITO FUNCIONAL RF-12

RF-13			
Nombre	Detección uso del GPS		
Descripción	El sistema determinará si el usuario presente en la imagen está operando con el GPS.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias	RF-02, RF-03		

TABLA 3.14. REQUISITO FUNCIONAL RF-13

RF-14			
Nombre	Detección distracción mensaje de texto		
Descripción	El sistema determinará si el usuario presente en la imagen está escribiendo o leyendo un mensaje de texto en un teléfono móvil.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias	RF-02, RF-03		

TABLA 3.15. REQUISITO FUNCIONAL RF-14

RF-15			
Nombre	Detección llamada de teléfono móvil		
Descripción	El sistema determinará si el usuario presente en la imagen está hablando por un teléfono móvil.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias	RF-02, RF-03		

TABLA 3.16. REQUISITO FUNCIONAL RF-15

RF-16			
Nombre	Detección conductor cansado		
Descripción	El sistema determinará si el usuario presente en la imagen muestra síntomas de fatiga o somnolencia.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias	RF-02, RF-03		

TABLA 3.17. REQUISITO FUNCIONAL RF-16

RF-17			
Nombre	Detección conductor distraído bebiendo		
Descripción	El sistema determinará si el usuario presente en la imagen está bebiendo mientras conduce.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias	RF-02, RF-03		

TABLA 3.18. REQUISITO FUNCIONAL RF-17

RF-18			
Nombre	Detección conductor distraído mirando hacia atrás		
Descripción	El sistema determinará si el usuario presente en la imagen está intentando alcanzar algo de la parte trasera del vehículo.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias	RF-02, RF-03		

TABLA 3.19. REQUISITO FUNCIONAL RF-18

RF-19			
Nombre	Visualización de resultado para la emoción		
Descripción	El sistema mostrará en la imagen los resultados obtenidos sobre la predicciones realizadas acerca de la emoción presente en el conductor.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias	RF-02, RF-03, RF-04		

TABLA 3.20. REQUISITO FUNCIONAL RF-19

RF-20			
Nombre	Visualización de resultado para la acción		
Descripción	El sistema mostrará en la imagen los resultados obtenidos sobre las predicciones realizadas acerca de la acción que este siendo ejecutada por el conductor.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias	RF-02, RF-03, RF-04		

TABLA 3.21. REQUISITO FUNCIONAL RF-20

3.4.2. Requisitos No Funcionales

En este apartado se determinarán los requisitos relativos a las decisiones de diseño del sistema, así como las tecnologías utilizadas. También se especificarán aspectos no visibles para el usuario del sistema, pero que sí afectan a la utilidad y usabilidad del software.

RNF-01			
Nombre	Resultados en tiempo real		
Descripción	El sistema estará procesando imágenes y mostrando los resultados de manera ininterrumpida.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias			

TABLA 3.22. REQUISITO NO FUNCIONAL RNF-01

RNF-02			
Nombre	Disponibilidad del sistema		
Descripción	El sistema tendrá una disponibilidad del 99 %.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias			

TABLA 3.23. REQUISITO NO FUNCIONAL RNF-02

RNF-03			
Nombre	Desarrollo del sistema en Python		
Descripción	El sistema será desarrollado utilizando únicamente el lenguaje de programación Python.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias			

TABLA 3.24. REQUISITO NO FUNCIONAL RNF-03

RNF-04			
Nombre	Técnicas de Deep Learning		
Descripción	El sistema utilizará técnicas de aprendizaje profundo.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias			

TABLA 3.25. REQUISITO NO FUNCIONAL RNF-04

RNF-05			
Nombre	Librería Open CV para detección facial		
Descripción	El sistema utilizará la librería Open CV para la detección facial en imágenes.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias			

TABLA 3.26. REQUISITO NO FUNCIONAL RNF-05

RNF-06			
Nombre	Librería Open CV para redimensionado de imágenes		
Descripción	El sistema utilizará la librería Open CV, para llevar acabo el redimensionado de imágenes.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias			

TABLA 3.27. REQUISITO NO FUNCIONAL RNF-06

RNF-07			
Nombre	Uso de Keras y Tensowflow		
Descripción	Las redes de neuronas convolucionales serán desarrolladas mediante el uso de las librerías de Keras y Tensowflow.		
Necesidad	Esencial	Deseable	Opcional
Prioridad	Alta	Media	Baja
Complejidad	Alta	Media	Baja
Estabilidad	Alta	Media	Baja
Dependencias			

TABLA 3.28. REQUISITO NO FUNCIONAL RNF-07

3.5. Casos de uso

En este apartado, se definirán los casos de uso del sistema desarrollado en este proyecto. Un caso de uso indica las acciones que ha de seguir el usuario o sistema para lograr llevar a cabo una tarea, o poder utilizar una funcionalidad. En el caso de este sistema, el usuario no será el encargado de realizar ninguna acción, pero sí que será un punto clave para que este pueda funcionar y llevar a cabo su cometido.

3.6. Descripción tabular de los casos de uso

La descripción tabular es utilizada para describir individualmente cada caso de uso, y sus dependencias o relaciones con otros casos de uso del sistema. La Figura 3.29 muestra la plantilla de la tabla que será utilizada para cada caso de uso.

CU-XX	
Nombre	
Objetivo	
Actores	
Precondiciones	
Postcondiciones	

TABLA 3.29. PLANTILLA TABLA DE CASOS DE USO

Los campos que se encuentran en la Figura 3.29 son los siguientes:

- **Identificador.** En la parte superior, la primera fila contendrá un identificado único para cada uno de los casos de uso.
- **Objetivo.** El campo objetivo indica la finalidad con la que se lleva a cabo cada uno de los casos de usos definidos.
- **Actores.** Este campo indica los usuarios o sistemas presentes en el caso de uso y que interactúan directa o indirectamente con el sistema creado.
- **Precondiciones.** Estas condiciones serán necesarias para poder llevar a cabo el caso de uso.
- **Postcondiciones.** Estas condiciones determinan de manera explícita el estado del sistema una vez se haya llevado a cabo el caso de uso.

CU-01	
Nombre	Conexión con cámara y obtención de imagen.
Objetivo	Se realiza una conexión entre el software y la cámara que proporcionará las imágenes del conductor.
Actores	El conductor será el usuario que activará el sistema.
Precondiciones	La cámara deberá ser funcional y estar debidamente conectada enfocando al conductor del vehículo.
Postcondiciones	Las imágenes recogidas ya podrán ser utilizadas en futuras fases.

TABLA 3.30. CASO DE USO CU-01

CU-02	
Nombre	Preprocesado de imagen y detección facial.
Objetivo	El sistema realiza el preprocesado de imágenes necesario para futuras fases, incluyendo la detección de cara.
Actores	-
Precondiciones	La cámara deberá de haber tomado imágenes del conductor. El conductor ha de estar situado dentro del campo de visión de la cámara.
Postcondiciones	Las imágenes preprocesadas ya podrán ser utilizadas en futuras fases.

TABLA 3.31. CASO DE USO CU-02

CU-03	
Nombre	Predicción de emoción.
Objetivo	El sistema calcula a través de técnicas de aprendizaje profundo la emoción presente en la imagen.
Actores	-
Precondiciones	La cámara deberá de haber tomado imágenes del conductor. El conductor ha de estar situado dentro del campo de visión de la cámara.
Postcondiciones	El sistema presenta la emoción presente en el conductor.

TABLA 3.32. CASO DE USO CU-03

CU-04	
Nombre	Predicción de acción.
Objetivo	El sistema calcula a través de técnicas de aprendizaje profundo la acción presente en la imagen.
Actores	-
Precondiciones	La cámara deberá de haber tomado imágenes del conductor. El conductor ha de estar situado dentro del campo de visión de la cámara.
Postcondiciones	El sistema presenta la acción presente en el conductor.

TABLA 3.33. CASO DE USO CU-04

Finalmente la Tabla 3.34 muestra la relación que existe entre los casos de uso que se han descrito en esta sección y los requisitos definidos en la sección anterior. Esta tabla de trazabilidad sirve para comprender la necesidad de los requisitos, y entender cómo formarán parte del sistema. Cada una de las filas de la tabla corresponde a un requisito, mientras que las columnas corresponde a un caso de uso. Cada asociación entre caso de uso y requisito estará marcada con una X en la celda correspondiente,

Requisitos \ Casos de uso	CU-01	CU-02	CU-03	CU-04
RF-01	X			
RF-02		X		
RF-03			X	X
RF-04		X		
RF-05			X	
RF-06			X	
RF-07			X	
RF-09			X	
RF-10			X	
RF-11			X	
RF-12				X
RF-13				X
RF-14				X
RF-15				X
RF-16				X
RF-17				X
RF-18				X
RF-19				X
RF-20				X

TABLA 3.34. MATRIZ DE TRAZABILIDAD ENTRE REQUISITOS Y CASOS DE USO

3.6.1. Descripción gráfica de los casos de uso

Para un mayor entendimiento de la interacción entre los diferentes casos de uso propuestos en el sistema, se ha desarrollado una representación gráfica de estos. La representación puede ser observada en la Figura 3.6.

En la parte izquierda de la Figura 3.6 se puede observar al actor del diagrama, que en este caso concreto será el conductor. Por otro lado, en la parte de la derecha se muestran los casos de uso, cada uno en un círculo independiente, así como las relaciones existentes entre ellos.

El conductor interactúa de manera indirecta con el sistema, ya que es el protagonista de las continuas imágenes que tomará la cámara. Estas imágenes son vitales y serán lo que permitirán realizar la predicción sobre las emociones y los comportamientos que esté realizando el conductor, es por eso que, como se muestra en el diagrama de la Figura 3.6 mediante las relaciones “*include*”, todos los casos de uso dependen de este caso en concreto. A su vez, para poder tomar una decisión sobre las imágenes, es necesario que se realice un preprocesado, y que además se aplique el software de detección facial.

Se puede concluir, que el sistema basa sus funcionalidades en las imágenes recogidas mediante una cámara. Esta es la base que usará el sistema para, en fases posteriores, poder llevar a cabo un mayor entendimiento de las imágenes mediante las técnicas de preprocesado, que servirán como entrada para poder lograr la principal funcionalidad del sistema, determinar los comportamientos presentes en el conductor.

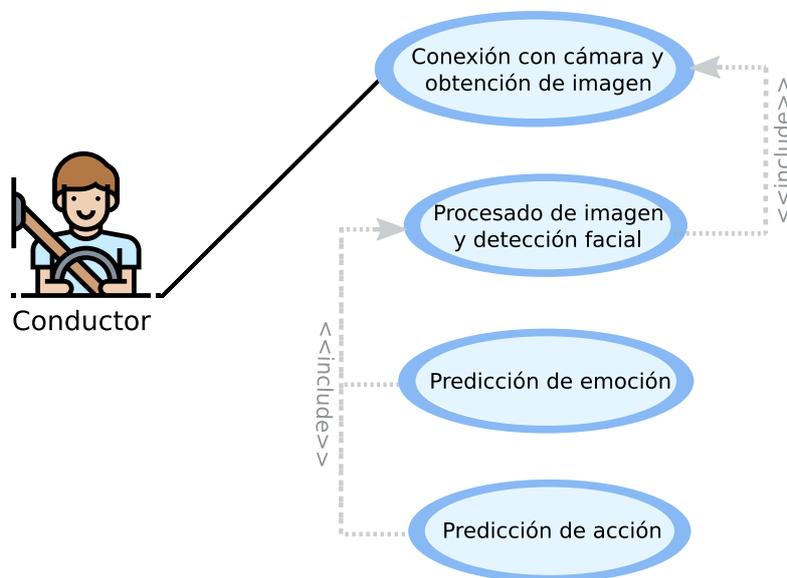


Fig. 3.6. Representación gráfica de los casos de uso

4. RECONOCIMIENTO DE EMOCIONES

En este capítulo se presentará el modelo de reconocimiento de emociones que será incluido en el asistente para conductores desarrollado para este proyecto.

Para poder desarrollar el sistema de reconocimiento de emociones, es necesario llevar a cabo un proceso de desarrollo de aprendizaje automático. Este proceso involucra las diferentes fases del ciclo de vida del aprendizaje automático.

En este capítulo se detallará el proceso de búsqueda y selección de los datos, para resolver el problema propuesto. Se presentarán los diferentes conjuntos de datos que han sido utilizados, tanto su procedencia como sus características principales. Además, se mostrarán y comentarán los resultados obtenidos mediante la experimentación.

4.1. Recolección de datos

El problema de reconocimiento de emociones a través de expresiones faciales, es un problema que se puede resolver tanto con aprendizaje supervisado, como no supervisado. Para este proyecto se utilizará aprendizaje supervisado, por lo tanto, todos los datos que se utilicen deberán de tener su etiqueta correspondiente que indicará la clase a la que pertenece.

Dado el interés que ha propiciado este tema durante la última década, ha crecido el número de conjuntos de datos que contienen las características adecuadas para la creación de un modelo mediante aprendizaje supervisado.

A continuación, se detallarán los conjuntos solicitados para llevar a cabo el entrenamiento del modelo.

4.1.1. The Extended Cohn-Kanade Dataset (CK+)

Cohn-Kanade Dataset [33] es set de datos que consta de imágenes de 210 adultos, de entre 18 y 50 años, donde un 69 % eran mujeres. Entre los candidatos que participaron en el estudio, el 81 % eran Euro-Americanos, el 13 % Afro-Americanos y el 6 % representaban otros grupos diferentes a los anteriores. Este set de datos es uno de los más interesantes y estudiados en el campo de reconocimiento de emociones, cuenta con diversidad en los participantes de las imágenes, además de calidad en las fotos pudiendo distinguirse perfectamente al sujeto (640x698) y la expresión que esta posando. Las imágenes contienen tanto expresiones posadas, como naturales.

Otros aspectos relevantes de este set, son la posición de las fotos, que se han tomado desde una vista frontal. Existe una gran diferencia entre una expresión desde una vista de perfil y una vista frontal, es por eso que para encontrar cohesión en el *dataset* y evitar

que el modelo no aprenda, se buscan conjuntos de datos que contengan imágenes tomadas desde vistas lo más similares.



Fig. 4.1. Muestra de imágenes del set de datos CK

La Figura 4.1 muestra algunos ejemplos de las imágenes que se pueden encontrar en este conjunto. Uno de los motivos por los cuales CK es uno de los conjuntos de imágenes más usados, es por la diversidad de los sujetos de las imágenes. Además, los participantes aparecen en las imágenes con accesorios en la parte superior de la cabeza, como gorras o sombreros. Esto que puede pasarse como un detalle insignificante, ayudará al modelo a poder clasificar correctamente las imágenes independientemente de los complementos que lleve el usuario, facilitando la extrapolación del modelo a imágenes del mundo real.

4.1.2. Facial Expression Recognition Challenge: FER (2013)

FER-2013 [36] es un *dataset* propuesto en la página web Kaggle. Este es el más extenso set de datos, para el reconocimiento de emociones, y consiste de un total de 28,000 imágenes etiquetadas para el entrenamiento, 3,500 imágenes para validación, y 3,500 imágenes para test. Este conjunto de datos se creó mediante la recolección de los resultados que aparecen en Google Imágenes, a través de la búsqueda de cada una de las siete emociones que se ven representadas. Debido a su naturaleza y el extenso número de datos que contiene, la calidad de las fotos se rebaja a un total de 64x64 píxeles por foto. Por otro lado, no se conoce la distribución de la población de los sujetos en las imágenes.

FER proporciona un número muy extenso de ejemplos, lo que permite que el modelo generalice las emociones y no se vea sesgado por ciertos tipos de características propias de las imágenes posadas, como puede ser la iluminación correcta o el enfocado perfecto del sujeto. Por otro lado, se han podido detectar ejemplos en el conjunto que no están correctamente clasificados, pero dado el extenso número de imágenes, no resulta viable la eliminación manual de dichos ejemplos. Independientemente de no conocer con certeza la distribución de la población, ni características relevantes de los sujetos como la edad o raza, sí que se puede observar que las imágenes contienen personas de todas las edades, géneros, y nacionalidades.

En la Figura 4.2 se puede ver una pequeña muestra de las imágenes que ofrece este conjunto de datos. Una de las características relevantes a comentar, es la diversidad de edades de los sujetos que está presente en este conjunto. Al contrario que en los otros

conjuntos, FER contiene ejemplos tanto de recién nacidos como de personas mayores. Esta variedad se consigue gracias a la procedencia de las imágenes, pero como consecuencia algunos de los ejemplos encontrados estarán mal clasificados, o directamente no serán fotos de personas como la segunda imagen de la Figura 4.2.



Fig. 4.2. Muestra de imágenes del set de datos FER2013

4.1.3. The japanese facial expression (JAFFE)

Uno de los objetivos que pretende cumplir este modelo, es la diversidad, ya que pretende detectar las expresiones de cualquier individuo, independientemente de su raza o género. El set de datos JAFFE contiene 213 imágenes, con siete expresiones faciales, posadas por diez mujeres modelos japonesas. La introducción de este *dataset* se basa en la posibilidad de que el modelo pudiera entender ejemplos de estas nacionalidades, y aprender a predecir las emociones independientemente de los rasgos faciales de los sujetos analizados.

En la Figura 4.3 se observa un conjunto de ejemplos de JAFFE, donde se muestran diferentes emociones expresadas por una de las modelos. Como se puede apreciar la diferencia entre las expresiones es más delicada que la observada en otras imágenes. Esto supondrá un reto para el modelo, ya que cuando mayor sea la expresividad del sujeto más fácil será encontrar patrones o diferencias.



Fig. 4.3. Muestra de imágenes del set de datos JAFFE

4.1.4. The Karolinska Directed Emotional Faces (KDEF)

Este *dataset* fue desarrollado por el Departamento de Neurociencia Clínica de *Karolinska Institute* en Suecia [48], las imágenes son posadas, tomadas desde cinco ángulos diferentes. Las imágenes tienen cuidada iluminación además de alta calidad en las fotos, siendo estas de 562x762 píxeles y representando siete emociones diferentes. En total se compone de 4900 imágenes, formadas por dos secuencias. Cada una de estas secuencias

contiene los mismos actores y las mismas poses para cada emoción, por lo que el número de fotos no duplicadas se reduce a 2450. La distribución de población, es más reducida que en los otros conjuntos de datos nombrados en ese apartado. Contiene un total de 70 sujetos de entre 20 y 30 años, de los cuales 35 son hombres y 35 son mujeres. A cada actor se le realizó una secuencia de imágenes por cada emoción, esta secuencia se compone de cinco ángulos diferentes. Entre estos ángulos se encuentra una imagen para cada de perfil, otra para cada medio perfil, y una de frente.

En la Figura 4.4 se muestra la secuencia obtenida para un actor. Uno de los motivos por los cuales se ha elegido este conjunto, es por las diferentes perspectivas que ofrece cada foto. Mientras que CK y JAFFE muestran todas sus imágenes desde una perspectiva frontal, gracias a este conjunto se le proporciona al modelo ejemplos de diferentes ángulos.



Fig. 4.4. Muestra de imágenes del set de datos KDEF

4.2. Unificación de los conjuntos

La naturaleza de los datos, aunque provienen de fuentes diferentes, es la misma. Todos contienen un conjunto de imágenes y una etiqueta que especifica la emoción de la imagen. Sin embargo, cada conjunto es definido por imágenes con diferentes dimensiones. De la misma manera, cada subconjunto de datos contiene una particular elección de emociones representadas en las fotos.

Para poder utilizar datos de diferentes fuentes, en el mismo problema de clasificación, es necesario poder unificarlas en cuanto a la distribución de los atributos y a las clases representadas. Dado que el modelado se realizará a partir de las imágenes, se necesitará que las imágenes representen las mismas emociones. Para ello se ha concretado un total de siete emociones, siendo estas un subconjunto de las más recurrentes en los datos.

<i>Dataset</i>	<i>Emociones</i>						
FER	<i>Anger</i>	<i>Disgust</i>	<i>Fear</i>	<i>Happiness</i>	<i>Sadness</i>	<i>Surprise</i>	<i>Neutral</i>
CK+	<i>Anger</i>	<i>Disgust</i>	<i>Fear</i>	<i>Happiness</i>	<i>Sadness</i>	<i>Surprise</i>	<i>Contempt</i>
JAFFE	<i>Anger</i>	<i>Disgust</i>	<i>Fear</i>	<i>Happiness</i>	<i>Sadness</i>	<i>Surprise</i>	<i>Neutral</i>
KDEF	<i>Anger</i>	<i>Disgust</i>	<i>Fear</i>	<i>Happiness</i>	<i>Sadness</i>	<i>Surprise</i>	<i>Neutral</i>

TABLA 4.1. EMOCIONES REPRESENTADAS EN LOS DATOS

La Tabla 4.1 muestra un resumen de las diferentes clases que representaba inicialmente cada conjunto de datos. Como se puede observar, la mayoría de las fuentes eligen definir las emociones básicas propuestas por Paul Ekman [8]. En este trabajo se ha tomado la decisión de mantener todas las clases representadas, con la excepción de la clase “*contempt*”, resaltada de color gris en la Tabla 4.1. Esta clase sólo estaba representada en un conjunto de datos, con menos de un centenar de ejemplos representativos, por lo que ha sido eliminada del problema.

4.3. Preprocesado de imágenes

Una de las ventajas que ofrece el aprendizaje profundo o Deep learning, y que ha sido comentada con anterioridad en este trabajo, es que no será necesario llevar a cabo un proceso de extracción de características. Sin embargo, sí que será necesario realizar un preprocesado para facilitar el aprendizaje a la red y evitar el aumento del coste computacional. En esta sección se detallarán los métodos utilizados para el preprocesado y ajuste de los datos.

4.3.1. Ajuste de dimensiones

Una de los principales requisitos al trabajar con redes convolucionales, es que todas las imágenes que se introduzcan deberán tener las mismas dimensiones. Este requisito es fundamental para el diseño de los filtros de las redes y el ajuste de los pesos en las diferentes capas de la red.

Originalmente, los conjuntos propuestos contenían cada uno diferentes dimensiones para las imágenes. Estas han sido modificadas a una misma dimensión de 64x64 píxeles. Este tamaño se ha obtenido de manera experimental, lanzando un conjunto de experimentos, se determinó que un cuadrado de 64 píxeles era el mínimo tamaño que conservaba suficiente información en la imagen para conseguir resultados competentes. Este balance es importante, ya que una imagen con demasiada información, tendrá como consecuente una red con tiempos de entrenamiento más elevados, así como de predicción, desfavoreciendo los resultados en tiempo real. La Figura 4.5 muestra un ejemplo del resultado obtenido al redimensionar una de las imágenes.

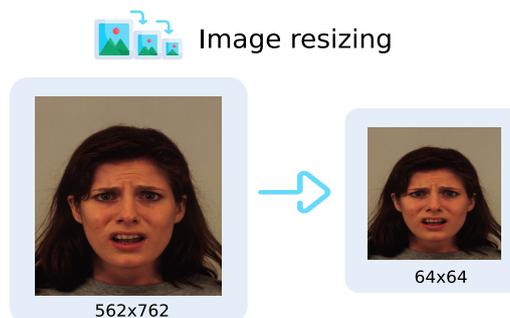


Fig. 4.5. Resultados de la reducción de dimensiones para cada conjunto de datos

Para lograr una reducción en el número de píxeles, sin afectar excesivamente a la calidad, se ha aplicado la técnica de interpolación lineal usando la librería OpenCV.

Como se puede observar en la Figura 4.5, la reducción del número en los píxeles, reduce notablemente la calidad de la imagen. Dado que en nuestro problema, la información que nos interesa es la representación de la emoción, esta reducción es irrelevante en ese aspecto, ya que se puede apreciar independientemente de la reducción la emoción observada. A cambio de la pérdida de calidad, esta técnica permite reducir la complejidad en tiempo y memoria del modelo considerablemente.

Con el fin de poder extraer la máxima información acerca de las características de las expresiones de los sujetos, y obviar aspectos redundantes para la solución del problema como el fondo de la imagen o incluso el cabello del fotografiado, se ha llevado a cabo un preprocesado avanzado en las imágenes. Este preprocesado, se compone de una serie de pasos que se pueden observar en la Figura 4.6.

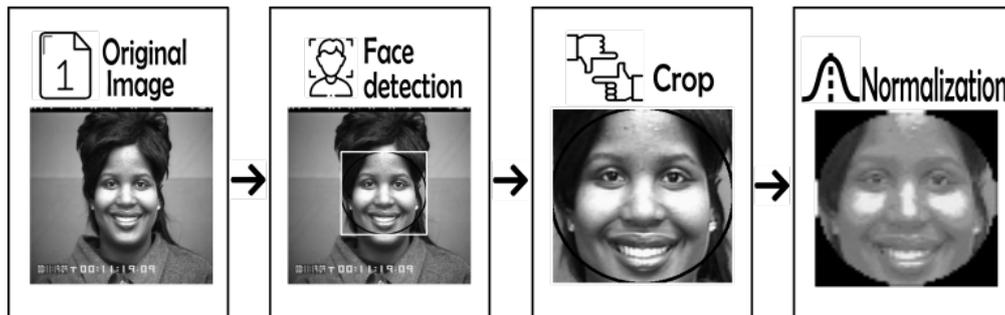


Fig. 4.6. Proceso de preprocesado de imágenes

Como se puede ver en la Figura 4.6, se realiza un proceso de detección facial sobre la imagen original. Este proceso se ha realizado utilizando el software de detección facial de la librería OpenCV. Una vez la cara en la imagen ha sido detectada, se dibuja un cuadrado al rededor de esta, y posteriormente un círculo la delimita.

Mediante el recorte del rostro de la imagen, y la eliminación de los píxeles extra convirtiéndolos en color negro, se consiguen eliminar los píxeles que no aportan información. Esta técnica no sólo facilita al modelo de aprendizaje automático la tarea de inferir la emoción de la imagen, ya que disminuye el número de posibles patrones que pueden determinar la clase, sino que también permite disminuir el tamaño que ocupa cada imagen, sin repercutir en la exactitud alcanzada por el modelo.

La posibilidad de reducir de tamaño las imágenes facilita increíblemente el desarrollo de este proyecto, esto es debido a la complejidad de las redes de aprendizaje profundo, así como de la cantidad de ejemplos que estos modelos necesitan para poder proporcionar buenos resultados. Si se puede reducir la dimensión, también se consigue reducir el número de pesos que tiene que almacenar el modelo, y esto permite entrenar un modelo en un menor tiempo y utilizando arquitecturas de red más complejas, sin llegar a sobrepasar

los límites de computación de los ordenadores convencionales.

4.3.2. Creación de subconjuntos de datos

Uno de los aspectos a tener en cuenta en cuanto al modelado de un problema de aprendizaje automático, es que no siempre más datos implica mejores resultados. Por este motivo, uno de los parámetros que variará en la experimentación, será el conjunto de imágenes utilizado en cada entrenamiento.

La experimentación se llevará a cabo con distintos subconjuntos de imágenes, que incluirán o excluirán algunos de los set de datos iniciales, o simplemente utilizan muestras de estos.

En la Tabla 4.3 se presentan las diferentes variantes de conjuntos de datos creadas. Esta tabla servirá de referencia para entender los resultados obtenidos de la experimentación, por ello, se ha incluido una nomenclatura que explica la información que tiene cada uno de los subconjuntos de imágenes. La Tabla 4.2 ha sido diseñada para ayudar al lector a entender dicha nomenclatura, sin necesidad de retornar a la tabla donde se describe de manera explícita dicha información.

Código	Descripción
J	Las imágenes incluyen el conjunto de datos JAFFE.
F	Las imágenes incluyen el conjunto de datos FER2013.
K	Las imágenes incluyen el conjunto de datos KDEF.
C	Las imágenes incluyen el conjunto de datos CK+.
*	Utiliza una muestra de imágenes del conjunto de datos que sigue al símbolo.
FS	Se ha aplicado la técnica de recorte de cara en las imágenes.
PREP	Se ha aplicado el proceso de preprocesado descrito en la Figura 4.6.

TABLA 4.2. GUÍA PARA ENTENDER LOS CÓDIGOS ASOCIADOS A LOS SUBCONJUTNOS DE IMÁGENES

Code	Datasets	Description
JFKC	KDEF, FER, CK+, JAFFE	Primera versión incluyendo todos los conjuntos de datos.
JFKC-FS	KDEF, FER, CK+, JAFFE	Se incluyen todos los conjuntos y se aplica la técnica de selección de cara.
JF*KC-FS	KDEF, FER, CK+, JAFFE	Eliminación manual de ejemplos ambiguos en el conjunto de datos FER2013.
JF*K*C-FS	KDEF, FER, CK+, JAFFE	Eliminación manual de las fotos de perfil completo en JAFFE.
FKC-FS	KDEF, FER, CK+, JAFFE	No se incluye el conjunto de datos JAFFE.
JFKC-FS-PREP	KDEF, FER, CK+, JAFFE	Se ha aplicado el proceso de preprocesado descrito en la Figura 4.6.

TABLA 4.3. DESCRIPCIÓN DE SUBCONJUTNOS DE IMÁGENES CREADOS

Finalmente, cabe destacar que los conjuntos desarrollados son parte de una evolución en el desarrollo del modelo. Es por esto, que, aunque no se explique el por qué de cada conjunto, estos son producto de decisiones informadas basadas en el análisis de los resultados obtenidos durante el entrenamiento del modelo.

4.4. Desarrollo del modelo

En este apartado se detallará el proceso de experimentación. Esta fase incluye el entrenamiento y ajuste del modelo. A continuación, se explicarán los hiper-parámetros elegidos. De la misma manera, se presentarán la batería de experimentos realizados, y se explicarán los resultados y entendimientos que dichas pruebas nos aportan.

4.4.1. Elección de hiper-parámetros y arquitectura

En este proyecto, se ha elegido la arquitectura CNN 1, descrita en la Tabla 4.4, como base. Esta fue la primera organización que consiguió dar forma al problema, y proponer resultados que, aunque presentaran un bajo nivel de exactitud, demostraban que existía un entendimiento básico sobre la solución.

A partir de la arquitectura base, se fueron modificando sus componentes, en base a los resultados obtenidos con cada modificación, creando así un conjunto de veinte arquitecturas, cuyas definiciones se pueden observar en la Tabla 4.4, Tabla 4.5, Tabla 4.6 y Tabla 4.7.

Como se puede observar en cada una de las arquitecturas, se ha utilizado un código de colores para facilitar la visualización de las mismas. Las capas menos densas que contienen menos filtros están coloreadas de colores más claros, mientras que las más densas van aumentando el tono del color a uno más oscuro. La misma idea se ha utilizado con la red de neuronas al final de cada arquitectura, correspondiéndose los colores más oscuros a un mayor número de neuronas en cada capa.

CNN 1	CNN 2	CNN 3	CNN 4	CNN 5
32 CONV 2D				
32 CONV 2D				
MaxPooling2D (2,2)				
DROPOUT (0.25)				
64 CONV 2D				
64 CONV 2D				
MaxPooling2D (2,2)				
DROPOUT (0.25)				
128 CONV 2D	128 CONV 2D	64 CONV 2D	128 CONV 2D	64 CONV 2D
128 CONV 2D	128 CONV 2D	64 CONV 2D	128 CONV 2D	64 CONV 2D
MaxPooling2D (2,2)				
DROPOUT (0.25)				
FLATTEN	FLATTEN	FLATTEN	FLATTEN	FLATTEN
DENSE 1000 UNITS	DENSE 500 UNITS	DENSE 500 UNITS	DENSE 1000 UNITS	DENSE 1000 UNITS
DROPOUT (0.25)				
OUTPUT SOFTMAX				

TABLA 4.4. ARQUITECTURA REDES CONVOLUCIONALES 1-5

CNN 6	CNN 7	CNN 8	CNN 9	CNN 10
128 CONV 2D	32 CONV 2D	64 CONV 2D	32 CONV 2D	32 CONV 2D
128 CONV 2D	MaxPooling2D (2,2)	MaxPooling2D (2,2)	MaxPooling2D (2,2)	MaxPooling2D (2,2)
MaxPooling2D (2,2)	DROPOUT (0.25)	DROPOUT (0.25)	DROPOUT (0.25)	DROPOUT (0.25)
DROPOUT (0.25)	64 CONV 2D	128 CONV 2D	64 CONV 2D	32 CONV 2D
64 CONV 2D	MaxPooling2D (2,2)	MaxPooling2D (2,2)	MaxPooling2D (2,2)	MaxPooling2D (2,2)
64 CONV 2D	DROPOUT (0.25)	DROPOUT (0.25)	DROPOUT (0.25)	DROPOUT (0.25)
MaxPooling2D (2,2)	128 CONV 2D	256 CONV 2D	MaxPooling2D (2,2)	MaxPooling2D (2,2)
DROPOUT (0.25)	MaxPooling2D (2,2)	MaxPooling2D (2,2)	DROPOUT (0.25)	DROPOUT (0.25)
128 CONV 2D	DROPOUT (0.25)	DROPOUT (0.25)	FLATTEN	FLATTEN
128 CONV 2D	FLATTEN	FLATTEN	DENSE 1000 UNITS	DENSE 500 UNITS
MaxPooling2D (2,2)	DENSE 1000 UNITS	DENSE 1000 UNITS	DROPOUT (0.25)	DROPOUT (0.25)
DROPOUT (0.25)	DROPOUT (0.25)	DROPOUT (0.25)	OUTPUT SOFTMAX	OUTPUT SOFTMAX
FLATTEN	OUTPUT SOFTMAX	OUTPUT SOFTMAX		
DENSE 500 UNITS				
DROPOUT (0.25)				
OUTPUT SOFTMAX				

TABLA 4.5. ARQUITECTURA REDES CONVOLUCIONALES 6-10

CNN 11	CNN 12	CNN 13	CNN 14	CNN 15
32 CONV 2D				
32 CONV 2D				
MaxPooling2D (2,2)				
DROPOUT (0.25)				
64 CONV 2D				
64 CONV 2D				
MaxPooling2D (2,2)				
DROPOUT (0.25)				
FLATTEN	FLATTEN	64 CONV 2D	64 CONV 2D	128 CONV 2D
DENSE 500 UNITS~	DENSE 128 UNITS	64 CONV 2D	64 CONV 2D	128 CONV 2D
DROPOUT (0.25)	DROPOUT (0.25)	MaxPooling2D (2,2)	MaxPooling2D (2,2)	MaxPooling2D (2,2)
OUTPUT SOFTMAX	OUTPUT SOFTMAX	DROPOUT (0.25)	DROPOUT (0.25)	DROPOUT (0.25)
		FLATTEN	FLATTEN	FLATTEN
		DENSE 1500 UNITS	DENSE 500 UNITS	DENSE 1500 UNITS
		DROPOUT (0.25)	DENSE 1500 UNITS	DROPOUT (0.25)
		OUTPUT SOFTMAX~~	DROPOUT (0.25)	OUTPUT SOFTMAX~~
			OUTPUT SOFTMAX~~	

TABLA 4.6. ARQUITECTURA REDES CONVOLUCIONALES 11-15

CNN 16	CNN 17	CNN 18	CNN 19	CNN 20
32 CONV 2D				
32 CONV 2D	32 CONV 2D		32 CONV 2D	MaxPooling2D (2,2)
MaxPooling2D (2,2)	MaxPooling2D (2,2)		MaxPooling2D (2,2)	DROPOUT (0.25)
DROPOUT (0.25)	DROPOUT (0.25)	32 CONV 2D	DROPOUT (0.25)	64 CONV 2D
32 CONV 2D	64 CONV 2D		32 CONV 2D	MaxPooling2D (2,2)
32 CONV 2D	64 CONV 2D		32 CONV 2D	DROPOUT (0.25)
MaxPooling2D (2,2)	MaxPooling2D (2,2)	MaxPooling2D (2,2)	MaxPooling2D (2,2)	128 CONV 2D
DROPOUT (0.25)			DROPOUT (0.25)	256 CONV 2D
64 CONV 2D	DROPOUT (0.25)	DROPOUT (0.25)	64 CONV 2D	
64 CONV 2D			64 CONV 2D	FLATTEN
MaxPooling2D (2,2)	FLATTEN	FLATTEN	MaxPooling2D (2,2)	
DROPOUT (0.25)			DROPOUT (0.25)	DENSE 1500 UNITS
FLATTEN				
DENSE 1500 UNITS	DENSE 1500 UNITS	DENSE 1500 UNITS	64 CONV 2D	
			64 CONV 2D	
DROPOUT (0.25)	DROPOUT (0.25)	DROPOUT (0.25)	MaxPooling2D (2,2)	DROPOUT (0.25)
			DROPOUT (0.25)	
			FLATTEN	
OUTPUT SOFTMAX	OUTPUT SOFTMAX	OUTPUT SOFTMAX	DENSE 1500 UNITS	OUTPUT SOFTMAX
			OUTPUT SOFTMAX	

TABLA 4.7. ARQUITECTURA REDES CONVOLUCIONALES 16-20

La Tabla 4.8 resume las decisiones tomadas para cada uno de los parámetros. Algunos parámetros como la razón de aprendizaje, o la función de activación se mantendrán fijos, y sus respectivos valores esta definidos en dicha tabla. Para la función de activación, se realizaron una serie de experimentos con diferentes funciones, todas no lineales, y finalmente se determinó la que mejores resultados propuso. De la misma manera se determinó Adam como el optimizador que se mantendrá para todos los experimentos.

El número de épocas indica el número de veces que la red itera sobre los datos, en términos generales, cuantas más iteraciones, mejores resultados serán obtenidos. Cuando un modelo converge a un valor, este no aumentará su exactitud, o cualquiera de las métricas que estén siendo evaluadas. Para este trabajo, no se varía el número de épocas, sino que se permite converger al modelo, decidiendo el mismo cuándo ha de parar.

El criterio de parada utilizado es el siguiente: si el modelo no ha mejorado o incrementado en más de un 0.05 la *accuracy* global en los resultados en la validación de las últimas siete iteraciones, entonces el modelo parará automáticamente porque ha encontrado el mejor resultado posible en base a sus parámetros. Esta medida es muy útil para evitar que el modelo no se sobre-entrene y por lo tanto no aprenda a generalizar el problema. Realizando una selección automática del número de épocas se evita tener que entrenar el mismo modelo con las mismas características más de una vez.

Si se lanza un modelo con un número determinado de épocas, y este finaliza cuando aún no había completado el aprendizaje, es necesario relanzar el mismo modelo con un número superior de épocas. De la misma manera, si se lanza un modelo con un número demasiado elevado de épocas, se incrementaría innecesariamente el tiempo de entrenamiento y además se sobre-entrenaría el modelo pudiendo derivar en peores resultados cuando se le presenten datos nuevos.

Finalmente, el *batch size*, esta profundamente ligado al número de datos con el que se esté trabajando, dado que el número de imágenes variará consecuencia de los diferentes conjuntos de datos utilizados, también será modificado este valor durante la experimentación.

Hiperparámetro	Valor	Descripción
Función de activación	Relu	Se mantiene fijo ya que ha mostrado resultados superiores frente a otras funciones no lineales.
Optimizador	Adam	Se mantiene fijo ya que ha mostrado resultados superiores frente a otros optimizadores en los datos. Se mantiene fija en un valor centesimal.
Razón de aprendizaje	0.001	Decisión tomada en base al comportamiento del modelo frente a razones de aprendizaje más altas. Mantener este valor facilita la búsqueda en el espacio de arquitecturas de la red.
Batch size	Variable	Dependiendo del experimento se varía este valor.
Épocas	Variable	Se utiliza un software que permite parar el modelo cuando este ha convergido a un valor

TABLA 4.8. DESCRIPCIÓN Y VALORES DE HIPERPARÁMETROS

4.5. Experimentación y ajuste

En este apartado, se hará una revisión conjunta de la batería completa de experimentos realizados.

En la Tabla 4.9 se puede observar que se han realizado un total de 45 experimentos válidos. Para cada uno, se muestra el conjunto de datos utilizado, cuyo nombre define las características del mismo, así como los hiper-parámetros que varían en cada experimento, es decir la arquitectura de redes de neuronas convolucionales con la que se ha entrenado dicho conjunto de datos y el *batch size* determinado.

En la última columna de la Tabla 4.9 se especifica si se ha realizado la técnica de *cross-validation* en los datos, y en caso afirmativo muestra el número de *folds* definido para la prueba. Esto consiste en ir alternando los datos que son utilizados para *test* y para *train* en cada entrenamiento, siendo el número de *folds* el total de entrenamientos diferentes realizados, finalmente el valor final es el que proporciona mejor *accuracy* dentro de las combinaciones que han sido probadas. Cómo se puede observar dado que esta técnica no mostraba cambios significativos en los resultados, se ha limitado a pruebas puntuales. Finalmente, se muestra una columna con la *accuracy* obtenida para los datos de test, así como el tiempo que ha llevado cada entrenamiento.

El mejor resultado se encuentra en el experimento 35, marcado en negrita, alcanzando un 85.5 de *accuracy* global en las instancias de test. Además, en la Tabla 4.9 se puede observar como a medida que avanzan los experimentos, ordenados por orden cronológico, se va a la vez incrementando la media de *accuracy* general obtenida. Esto se debe a las modificaciones en los parámetros de la red, que se han determinado en base a la experiencia obtenida de resultados anteriores.

Experiment	Dataset	Model	Batch Size	Global Acc.	Time (h:m)	CV Folds
1	JFKC	CNN 1	500	57.7	0:05	-
2	JFKC	CNN 2	500	73.1	0:07	-
3	JFKC	CNN 3	500	61.8	0:07	-
4	JFKC	CNN 4	500	68.9	0:07	-
5	JFKC	CNN 5	128	72.2	0:07	-
6	JFKC	CNN 5	128	68.9	0:09	3
7	JFKC	CNN 5	128	70.1	0:03	3
8	JFKC-FS	CNN 2	500	71.4	0:08	-
9	JFKC-FS	CNN 6	500	70	0:04	-
10	JFKC-FS	CNN 7	128	65.6	0:12	-
11	JFKC-FS	CNN 7	128	65.6	0:01	-
12	JFKC-FS	CNN 8	128	63.6	0:02	-
13	JFKC-FS	CNN 9	128	64.4	0:02	-
14	JFKC-FS	CNN 2	500	69.1	0:04	-
15	JFKC-FS	CNN 2	250	71.1	0:04	-
16	JFKC	CNN 2	128	66	0:04	-
17	JFKC-FS	CNN 3	250	71.9	0:04	-
18	JFKC-FS	CNN 10	250	68.3	0:04	-
19	JFKC-FS	CNN 11	500	70	0:04	-
20	JF*KC-FS	CNN 11	250	67.8	0:02	-
21	JF*KC-FS	CNN 11	250	63.6	0:02	-
22	JF*KC-FS	CNN 11	128	72.4	0:04	-
23	JF*KC-FS	CNN 11	128	66.7	0:04	-
24	JF*KC-FS	CNN 11	128	80	0:04	4
25	JF*KC-FS	CNN 12	128	78	0:04	4
26	JF*KC-FS	CNN 12	128	56.7	0:04	-
27	JF*K*C-FS	CNN 12	128	57.8	0:04	-
28	FKC-FS	CNN 12	500	57.9	0:32	-
29	FKC-FS	CNN 12	1024	57.9	0:39	-
30	FKC-FS	CNN 12	128	78.5	1:01	-
31	JFKC-FS-PREP	CNN 12	128	81.2	0:59	-
32	JFKC-FS-PREP	CNN 12	128	84.1	0:04	-
33	JFKC-FS-PREP	CNN 12	128	83.7	0:09	-
34	JFKC-FS-PREP	CNN 5	128	84.4	0:13	-
35	JFKC-FS-PREP	CNN 13	128	85.5	0:18	-
36	JFKC-FS-PREP	CNN 14	128	80.5	0:14	-
37	JFKC-FS-PREP	CNN 15	128	85	0:14	-
38	JFKC-FS-PREP	CNN 16	128	84.6	0:14	-
39	JFKC-FS-PREP	CNN 16	128	83.9	0:15	-
40	JFKC-FS-PREP	CNN 17	128	83.8	0:12	-
41	JFKC-FS-PREP	CNN 16	64	84.6	0:11	-
42	JFKC-FS-PREP	CNN 16	16	71.7	0:15	-
43	JFKC-FS-PREP	CNN 18	16	80.1	0:13	-
44	JFKC-FS-PREP	CNN 19	64	71	0:04	-
45	JFKC-FS-PREP	CNN 20	64	78.8	0:10	-

TABLA 4.9. RESULTADOS EXPERIMENTACIÓN

Una de las alternativas, cuando se estudia la elección de una arquitectura de redes convolucionales que resuelva un problema, es la búsqueda en el espacio de arquitecturas. Las dimensiones, y por ende posibilidades, que este espacio ofrece, dificulta la detección de una solución óptima. Sin embargo, una búsqueda puede resultar útil para encontrar una arquitectura base, que después podrá ser modificada.

Aunque técnicas más avanzadas de algoritmos evolutivos pueden ser usadas para optimizar la búsqueda, para este trabajo se ha optado por una optimización bayesiana, presente en la librería Keras. Esta permite una búsqueda más informada que la búsqueda aleatoria, basándose en modelos probabilísticos sobre los resultados obtenidos.

Más detalles sobre los parámetros elegidos para ajustar esta búsqueda se encuentran en la Tabla 4.10. En cada una de las iteraciones no solo se varía la arquitectura, sino que también otros parámetros relevantes como la función de activación o el optimizador utilizado.

Hiper-parámetro	Valor	Descripción
Max. trials	50	Total de arquitecturas evaluadas.
Épocas	25	Número de épocas de entrenamiento por cada arquitectura.

TABLA 4.10. PARÁMETROS DE LA BÚSQUEDA BAYESIANA

Finalmente, en la Tabla 4.11 se puede observar los resultados fruto de la búsqueda realizada. El primer experimento, de más de 24 horas de duración, llevo a cabo la búsqueda proporcionando una potencial red de neuronas convolucional. Esta red obtuvo menos de un cincuenta por ciento de *accuracy* global en test. Pudiendo ser esto causa de las pocas iteraciones llevadas a cabo en la búsqueda, necesarias para poder evaluar diferentes modelos, se realizaron cinco entrenamientos extra con el mismo modelo incrementando el número de épocas. Como se puede observar, esto no proporcionó mejores resultados, siendo estos mucho menores que los obtenidos mediante la experimentación informada no automática.

Experiment	Dataset	Batch Size	Epochs	Global Acc.	Time (h:m)
Bayesian-Optimization 1	JFKC-FS	128	25	49.1	26:15
Bayesian-Optimization 2	JFKC-FS	250	35	42.9	0:14
Bayesian-Optimization 3	JFKC-FS	250	35	49.1	0:13
Bayesian-Optimization 4	JFKC-FS	250	55	49.5	0:20
Bayesian-Optimization 5	JFKC-FS	250	55	42.5	0:12
Bayesian-Optimization 6	JFKC-FS	128	150	46.9	0:35

TABLA 4.11. RESULTADOS OBTENIDOS DE LA BUSQUEDA BAYESIANA

La Tabla 4.12 proporciona un resumen de aspectos relevantes obtenidos de la experimentación. Se han realizado un total de 79 experimentos, de los cuales 45 han sido válidos. Un experimento válido, se diferencia de uno que no lo es en el hecho de que el primero muestra unos resultados donde se observa que el modelo ha partido de una inicialización de pesos válida, que le ha permitido obtener unos resultados, independientemente de lo buenos que estos sean. Un modelo no válido, es aquel que proporcionará unos resultados que carecen de sentido, y normalmente se detectan porque todas las instancias se etiquetan con la misma clase.

Número de experimentos		Tiempo de experimentación		Accuracy	
Totales	Válidos	Total	Válida	Media	Mejor valor
79	45	60 horas y 38 minutos	36 horas y 10 minutos	72.45	85.5

TABLA 4.12. RESUMEN DE LA EXPERIMENTACIÓN

La Figura 4.7 presenta un ejemplo de un experimento no válido. Este hecho ocurre debido a la inicialización aleatoria inicial de los pesos. Si se lanza dos veces el mismo modelo, este factor de inicialización puede afectar a los resultados increíblemente, pudiendo proporcionar el primer experimento un valor global de *accuracy* del 90% y el segundo del 20%, siendo ambos producto el mismo código sin ninguna modificación. Estos experimentos no proporcionan resultados, ni información válida de cómo de buena es la arquitectura elegida, por eso no se han comentado, pero si que requieren un tiempo de ejecución y son inevitables, por esto se han incluido en el resumen general de la experimentación.

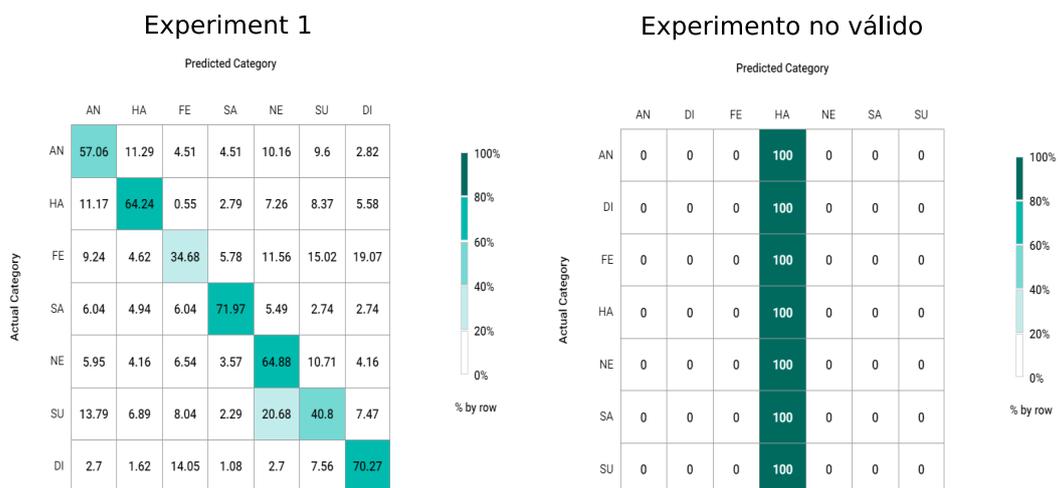


Fig. 4.7. Comparación de un experimento válido (Experiment 1) y uno no válido

4.6. Modelo final

Finalmente, antes de concluir con la experimentación, en este apartado se estudiarán algunas de las características que presenta el que se ha definido como el mejor modelo y por lo tanto el que se usará para futuras pruebas.

En la Figura 4.8 se muestra una representación en 3D de la arquitectura que ha sido utilizada para entrenar el modelo final. En esta figura se puede observar como la imagen se modifica acorde al preprocesado explicado en secciones anteriores, y finalmente se introduce a la red convolucional. Esta representación en 3D permite observar las diferentes capas de la red de una manera más intuitiva, para mayor legibilidad no se han incluido las capas de *subsampling*. Tras obtenerse las características de la imagen a través de las capas de convolución, la red de neuronas situada al final determinará a cual de las clases pertenece la imagen.

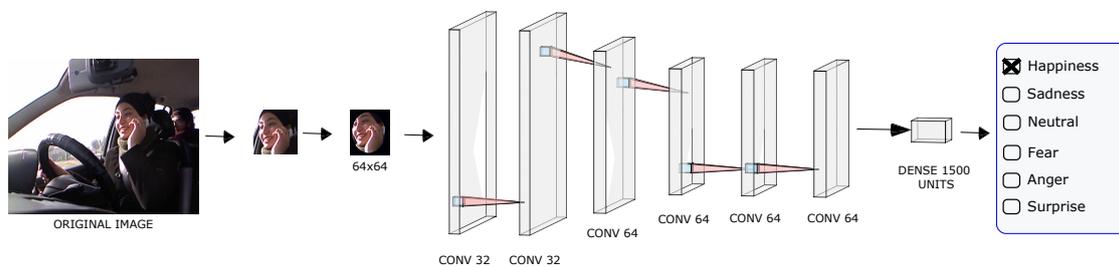


Fig. 4.8. Representación 3D de la mejor arquitectura de CNN

La Figura 4.9 muestra como la exactitud o *accuracy* aumenta a medida que el modelo ajusta sus pesos para adaptarlos al problema propuesto. Una vez se alcanza un pico, el modelo comienza a converger hacia el resultado final, variando levemente su porcentaje de acierto. En el eje Y se encuentra la *accuracy* obtenida, en cada una de las iteraciones en las que se han modificado los pesos. Por otro lado, en el eje X se observa los *steps* que se han realizado, es decir, las veces que se han evaluado los datos, tras modificar los pesos, en un mismo entrenamiento.

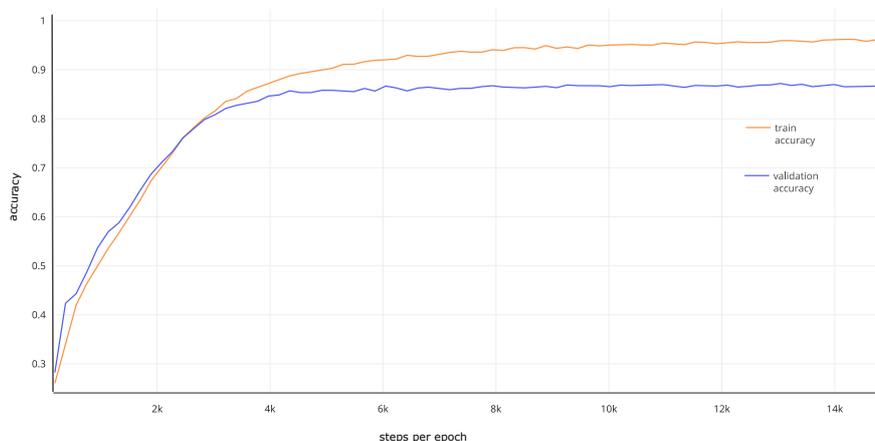


Fig. 4.9. Evolución de *accuracy* en mejor modelo

En la Tabla 4.12 se observan los resultados globales de exactitud o *accuracy* para el mejor modelo. Sin embargo, en la Figura 4.10, se muestra la matriz de confusión obtenida para las instancias de test analizadas con el modelo final.



Fig. 4.10. Matriz de confusión del mejor modelo

De la matriz en la Figura 4.12, se puede inferir como se comporta el modelo para cada una de las clases estudiadas. Cada columna y cada fila está titulada con las dos primeras letras, de los nombres en inglés, de cada una de las emociones estudiadas. La diagonal nos muestra, el porcentaje de instancias correctamente clasificadas para cada una de las clases. En las filas se puede observar la categoría real, y en la columna la categoría predicha, por lo que en una celda de la matriz, el número que aparece es el porcentaje de instancias etiquetadas con la emoción mostrada en la fila, que han sido predichas, por el modelo, con la emoción mostrada en la columna.

Se puede observar, al igual que se observó en la mayoría de experimentos realizados, que las clases más difíciles de aprender, y que por lo tanto muestran porcentajes más bajos en la matriz de confusión, son la emoción “*fear*” y la emoción “*sadness*”. Por otro lado, la clase que mayor valor de exactitud ha presentado, durante los numerosos experimentos, ha sido la clase “*happiness*”.

De esta manera, se puede concluir, que las expresiones faciales mostradas cuando se quiere expresar alegría o felicidad son más fáciles de aprender, para el modelo, que otras expresiones características de emociones más complejas como la tristeza o el miedo.

4.7. Ambigüedad en los resultados

Uno de los problemas que se puede presentar al trabajar con datos que representan conceptos complejos, como lo son las emociones, es que puede surgir cierta ambigüedad en los resultados obtenidos. Para imágenes que vienen de fuentes fiables, y han sido desarrolladas por investigadores y categorizadas por psicólogos, se ha prevalecido siempre la etiqueta otorgada a cada imagen.

Por otro lado, algunos de los conjuntos de datos como FER2013 [36], no han tenido ninguna supervisión o revisión en el etiquetado de las imágenes. Es por eso, que aunque los resultados del modelo son bastante favorables alcanzando un 85 % de *accuracy* global, cabe destacar que algunos de las imágenes que se han categorizado como mal etiquetadas por el modelo, presentan una cierta ambigüedad y podrían redefinirse como imágenes bien etiquetadas, ya que presentan las cualidades para serlo según las expresiones faciales típicas en cada emoción.

Image				
Predicted Label	Surprise Neutral	Surprise Happiness	Sadness Neutral	Sadness Fear
Source	FER 2013	FER 2013	FER 2013	FER 2013

Fig. 4.11. Ejemplos de posibles errores en el etiquetado

Aunque resulta inviable la revisión manual de todas las imágenes que se han utilizado para entrenar y ajustar el modelo, tras realizar un barrido general sobre las imágenes se han encontrado casos como el expuesto en la Figura 4.11. Como se puede observar en la figura, basándonos en los estándares que definen las características propias de una emoción, véase Figura 2.2, se puede determinar que las imágenes no estaban correctamente etiquetadas. Es por esto que aunque para el modelo la predicción realizada haya sido categorizada como errónea, en realidad el resultado no lo es.

5. RECONOCIMIENTO DE COMPORTAMIENTO EN CONDUCTORES

En este capítulo se presentará el modelo de reconocimiento de comportamientos en conductores que será incluido en el Asistente para Conductores desarrollado para este proyecto.

Será presentado el proceso de búsqueda y selección de los datos, para resolver el problema propuesto. Se presentarán los diferentes conjuntos de datos que han sido utilizados, tanto su procedencia como sus características principales. Además, se mostraran y comentarán los resultados obtenidos mediante la experimentación.

5.1. Descripción de datos

Hoy en día, existe un gran número de conjuntos de datos, los cuales contienen imágenes etiquetadas de conductores mostrando diferentes comportamientos. Para este trabajo se ha realizado un estudio de diferentes bases de datos, examinando las características de las imágenes proporcionadas por cada una de ellas.

Las imágenes proporcionadas por las diferentes fuentes varían no solo en las acciones que esté ejecutando el sujeto sino también en las perspectivas de la imagen según la posición de la cámara en el coche. Para ilustrar esta información se ha desarrollado un diagrama donde se incluyen las posibles vistas del conductor que se pueden encontrar, a día de hoy, dependiendo de la base de datos que se precise utilizar.



Fig. 5.1. Vistas de las imágenes en 3MDAD

El proyecto busca un sistema que no sólo informe del comportamiento, sino también de la emoción del conductor. Teniendo esto presente, se ha elegido trabajar con un conjunto de imágenes donde se pueda apreciar la cara del conductor desde una vista frontal, para así poder utilizar ambos modelos al mismo tiempo sin necesidad de añadir cámaras extra al vehículo.

La Figura 5.1 muestra dos ejemplos de imágenes que se pueden encontrar en [49], la vista que se ha elegido para el entrenamiento del modelo corresponde a la vista frontal de la figura.

Tras realizar un estudio sobre las diferentes conjuntos de datos disponibles, que cumplan con la características que se buscan en este proyecto, la única fuente de datos que proporciona un vista frontal del conductor es *Multimodal Multiview and Multispectral Driver Action Dataset (3MDAD)* [49]. Este conjunto de datos desarrollado por Jegham et al. contiene imágenes de cincuenta sujetos diferentes realizando un total de dieciséis acciones. Dichas acciones, que serán identificadas de ahora en adelante en este documento como clases, se encuentran desglosadas en la Tabla 5.1. Para este proyecto se han mantenido como etiquetas los nombres originales en inglés.

Código	Descripción	Código	Descripción
AC1	<i>Safe Driving</i>	AC9	<i>Having picture</i>
AC2	<i>Doing hair and makeup</i>	AC10	<i>Talking to passenger</i>
AC3	<i>Adjusting radio</i>	AC11	<i>Singing or dancing</i>
AC4	<i>GPS operating</i>	AC12	<i>Fatigue and somnolence</i>
AC5	<i>Writing message using right hand</i>	AC13	<i>Drinking using right hand</i>
AC6	<i>Writing message using left hand</i>	AC14	<i>Drinking using left hand</i>
AC7	<i>Talking phone using right hand</i>	AC15	<i>Reaching behind</i>
AC8	<i>Talking phone using left hand</i>	AC16	<i>Smoking</i>

TABLA 5.1. DESCRIPCIÓN DE LAS ETIQUETAS DEL CONJUTNO DE DATOS

Finalmente, en la Figura 5.2 se puede observar una muestra de diferentes imágenes tomadas para el mismo sujeto, donde se encuentra realizando las acciones descritas en la Tabla 5.1. Como se puede observar en la tabla, cada imagen muestra en la parte superior el código asociado a cada una de las etiquetas descritas en la Tabla 5.1.



Fig. 5.2. Muestra de las clases representadas en el conjunto de imágenes

De la misma manera, la Figura 5.2 presenta ejemplos de las imágenes que serán utilizadas para el entrenamiento del modelo. Inicialmente, se partirá del uso de las dieciséis clases, pero como se argumentará en el apartado de experimentación, con vistas a mejorar los resultados para las clases clave de gran interés para el problema este número de clases será reducido posteriormente. Durante este trabajo se utilizarán ‘AC’ y acción indistintamente en las tablas y figuras.

Originalmente, este conjunto cuenta con un total de 111,006 imágenes. Estas se dividen en secuencias de aproximadamente 130 fotos para cada uno de los sujetos imitando cada uno de los 16 comportamientos desglosados en la Tabla 5.1.

5.2. Preparación de los datos

Antes de crear el modelo, es necesario redimensionar las imágenes para obtener el menor tamaño posible, sin perder el significado de la imagen. Como se ha explicado en previos apartados, esto nos permitirá un entrenamiento del modelo con menor tiempo de computación y menos gasto de recursos computacionales. En la Figura 5.3 se puede observar el tamaño final de la imagen tras aplicarle la técnica de redimensión, la cual ha sido realizada de nuevo mediante la técnica de interpolación lineal.

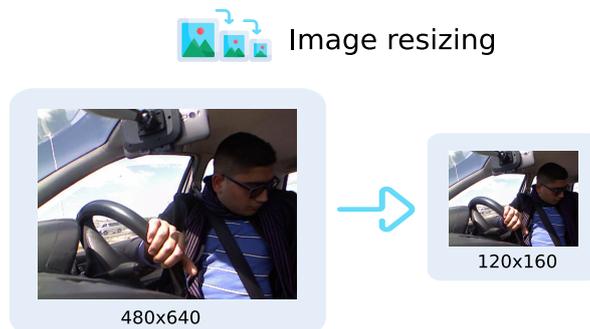


Fig. 5.3. Redimensionado de las imágenes

Para este conjunto de datos, ha sido necesario realizar además una limpieza en de las secuencias de imágenes. Estas muestran al sujeto realizando cada una de las acciones, pero también muestra el proceso previo y posterior a la acción, lo que podría confundir al modelo. En la Figura 5.4 se ha ilustrado este problema.

En la Figura 5.4 se ha incluido una secuencia completa de imágenes, donde se puede observar que las fotografías iniciales no corresponden con la acción propuesta. Una vez esto se detectó en la parte experimental del trabajo, las imágenes conflictivas fueron eliminadas del conjunto de datos.



Fig. 5.4. Ejemplo de secuencia de imágenes para eliminación

Finalmente, los números en la Figura 5.4 representan el número de imagen en la secuencia, no debe confundirse con la acción que se representa, todas las imágenes de la figura contienen la misma etiqueta asociada a la acción ‘*message right*’.

Para la eliminación de dichas imágenes, se llevaron a cabo dos estrategias diferentes.

Inicialmente, se eliminaron las diez primeras imágenes de cada secuencia, este proceso será referido en el trabajo como PRE-1.

Tras observar que los resultados no eran del todo favorables, se llevo a cabo un proceso exhaustivo de eliminación manual de imágenes mal etiquetadas. Manualmente se reviso cada una de las secuencias y se eliminaron las imágenes que no representaban la acción indicada, este proceso será referido en el trabajo como PRE-2.

La Tabla 5.2 resume la información dada a cerca del preprocesado de la base de datos.

Código	Descripción
PRE-1	Eliminado automático de las diez primeras imágenes de la secuencia.
PRE-2	Revisión intensiva de las secuencias con eliminación manual de ejemplos mal etiquetados.

TABLA 5.2. PREPROCESADO APLICADO A LAS IMÁGENES DE 3MDAD

En los siguientes apartados, se detallará de nuevo que tipo de preprocesado ha sido usado en cada experimento utilizando el código descrito en la Tabla 5.2.

Finalmente, en la Tabla 5.3 se puede observar las distintas versiones de conjuntos de datos que han sido creadas para este proyecto. En las columnas se presenta el conjunto de acciones, y en las filas el número de versión del conjunto. Cada casilla marcada con una X implica que el conjunto de datos creado incluye la clase a la que corresponde cada una de las columnas.

Para referirnos a cada una de las versiones de subconjuntos creadas en el apartado de experimentación, se utilizarán los códigos asociados en la Tabla 5.4.

Versión	Acción																Preprocesado	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	PRE-1	PRE-2
V1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X		
V2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	
V3	X	X		X	X	X	X	X	X	X	X	X	X	X	X	X	X	
V4	X	X		X	X	X	X	X		X	X	X	X	X	X	X	X	
V5	X	X		X	X	X	X	X		X	X	X	X	X	X		X	
V6	X			X	X	X	X	X		X	X	X	X	X	X		X	
V7	X			X	X	X	X	X		X		X	X	X	X		X	
V8	X			X	X	X	X	X				X	X	X	X		X	
V9	X			X	X	X	X	X				X	X	X	X		X	X
V10	X			X	X		X					X	X		X		X	X
V11	X			X	X		X						X		X		X	X
V12				X	X		X						X		X		X	X

TABLA 5.3. DESCRIPCIÓN DE LOS SUBCONJUNTOS CREADOS A PARTIR DE 3MDAD

Versión	Código
V1	A
V2	A*
V3	A-(3)*
V4	A-(3,9)*
V5	A-(3,9,16)*
V6	A-(3,9,16,2)*
V7	A-(3,9,16,2,11)*
V8	A-(3,9,16,2,10,11)*
V9	A-(3,9,16,2,10,11)**
V10	A-(3,9,16,2,10,11,6,8,14)**
V11	A-(3,9,16,2,10,11,6,8,14,12)**
V12	A-(3,9,16,2,10,11,6,8,14,12,1)**

TABLA 5.4. CÓDIGO ASIGNADO A CADA SUBCONJUNTO

5.3. Construcción del modelo: Experimentación y ajuste

La estrategia utilizada para la experimentación, es la misma que en el capítulo anterior. Los hiper-parámetros utilizados y sus valores pueden verse en la Tabla 4.8.

La elección de los hiper-parámetros ha sido la misma que la abordada en el desarrollo del modelo de emociones. No obstante, las arquitecturas desarrolladas durante esta segunda experimentación varían respecto a las utilizadas en el modelo de emociones.

En la Tabla 5.5 y Tabla 5.6 se puede observar las arquitecturas desarrolladas que serán producto de la experimentación y el ajuste del modelo.

CNN 1	CNN 2	CNN 3	CNN 4
32 CONV 2D	32 CONV 2D	32 CONV 2D	32 CONV 2D
32 CONV 2D	32 CONV 2D	32 CONV 2D	32 CONV 2D
MaxPooling2D (2,2)	MaxPooling2D (2,2)	MaxPooling2D (2,2)	MaxPooling2D (2,2)
DROPOUT (0.25)	DROPOUT (0.25)	DROPOUT (0.25)	DROPOUT (0.25)
32 CONV 2D	32 CONV 2D	32 CONV 2D	64 CONV 2D
32 CONV 2D	32 CONV 2D	32 CONV 2D	64 CONV 2D
MaxPooling2D (2,2)	MaxPooling2D (2,2)	MaxPooling2D (2,2)	MaxPooling2D (2,2)
DROPOUT (0.25)	DROPOUT (0.25)	DROPOUT (0.25)	DROPOUT (0.25)
64 CONV 2D	64 CONV 2D	64 CONV 2D	128 CONV 2D
64 CONV 2D	64 CONV 2D	64 CONV 2D	128 CONV 2D
MaxPooling2D (2,2)	MaxPooling2D (2,2)	MaxPooling2D (2,2)	MaxPooling2D (2,2)
DROPOUT (0.25)	DROPOUT (0.25)	DROPOUT (0.25)	DROPOUT (0.25)
128 CONV 2D	128 CONV 2D	128 CONV 2D	256 CONV 2D
128 CONV 2D	128 CONV 2D	128 CONV 2D	256 CONV 2D
MaxPooling2D (2,2)	MaxPooling2D (2,2)	MaxPooling2D (2,2)	MaxPooling2D (2,2)
DROPOUT (0.25)	DROPOUT (0.25)	DROPOUT (0.25)	DROPOUT (0.25)
FLATTEN	FLATTEN	256 CONV 2D	FLATTEN
DENSE 128 UNITS	DENSE 1500 UNITS	256 CONV 2D	DENSE 128 UNITS
DROPOUT (0.25)	DROPOUT (0.25)	MaxPooling2D (2,2)	DROPOUT (0.25)
OUTPUT SOFTMAX	OUTPUT SOFTMAX	DROPOUT (0.25)	OUTPUT SOFTMAX
		FLATTEN	
		DENSE 128 UNITS	
		DROPOUT (0.25)	
		OUTPUT SOFTMAX	

TABLA 5.5. ARQUITECTURA REDES CONVOLUCIONALES 1-4

CNN 5	CNN 6	CNN 7	CNN 8	CNN 9
32 CONV 2D				
32 CONV 2D				
MaxPooling2D (2,2)				
DROPOUT (0.25)				
64 CONV 2D				
64 CONV 2D				
MaxPooling2D (2,2)				
DROPOUT (0.25)				
64 CONV 2D	128 CONV 2D	64 CONV 2D	64 CONV 2D	64 CONV 2D
64 CONV 2D	128 CONV 2D	64 CONV 2D	64 CONV 2D	64 CONV 2D
MaxPooling2D (2,2)				
DROPOUT (0.25)				
128 CONV 2D	256 CONV 2D	128 CONV 2D	128 CONV 2D	128 CONV 2D
128 CONV 2D	256 CONV 2D	128 CONV 2D	128 CONV 2D	128 CONV 2D
MaxPooling2D (2,2)				
DROPOUT (0.25)				
256 CONV 2D			256 CONV 2D	256 CONV 2D
256 CONV 2D	FLATTEN	FLATTEN	256 CONV 2D	256 CONV 2D
MaxPooling2D (2,2)	DENSE 500 UNITS	DENSE 500 UNITS	MaxPooling2D (2,2)	MaxPooling2D (2,2)
DROPOUT (0.25)				
FLATTEN			FLATTEN	FLATTEN
DENSE 500 UNITS			DENSE 500 UNITS	DENSE 500 UNITS
DROPOUT (0.25)	OUTPUT SOFTMAX	OUTPUT SOFTMAX	DROPOUT (0.25)	DENSE 1500 UNITS
OUTPUT SOFTMAX			DROPOUT (0.25)	DROPOUT (0.25)
			OUTPUT SOFTMAX	OUTPUT SOFTMAX

TABLA 5.6. ARQUITECTURA REDES CONVOLUCIONALES 5-9

Para este conjunto de datos, se han realizado un total de 22 experimentos descritos en la Tabla 5.7. En estos experimentos, las únicas variaciones respecto a los realizados en previos apartados, es que contienen como parámetro el tamaño del subconjunto de imágenes que serán utilizadas para el test.

Se puede observar como la *accuracy* global obtenida es generalmente creciente, aumentando conforme se van realizando más experimentos. El mejor valor de *accuracy* obtenido para todas las clases, se sitúa en 97.7.

Existe un notable escalón entre los experimentos que se realizaron utilizando un tamaño del conjunto de test de 0.25, y los que por el contrario se redujo esa cifra. Al aumentar las imágenes con las que se entrena el modelo, y por ende disminuir las imágenes analizadas, el modelo presenta resultados mucho más remarcables.

Como se puede observar en la Tabla 5.7, algunos de los experimentos presentan los mismos valores, esta técnica se ha utilizado para evaluar las consecuencias del índice de aleatoriedad en el resultado obtenido.

Experiment	Dataset	Test size	Model	Batch Size	Global Acc.	Time (h:m)
1	A	0.25	CNN 1	256	22.8	2:56
2	A	0.25	CNN 2	256	17.1	0:40
3	A	0.25	CNN 3	256	33.2	0:53
4	A	0.25	CNN 4	256	32.3	1:15
5	A	0.25	CNN 5	256	35.2	0:58
6	A*	0.25	CNN 5	256	34.2	0:56
7	A-(3)*	0.25	CNN 5	256	30.1	0:58
8	A-(3,9)*	0.25	CNN 5	256	32.4	1:02
9	A-(3,9)*	0.25	CNN 6	125	25.4	0:40
10	A-(3,9,16)*	0.25	CNN 5	125	35.3	0:48
11	A-(3,9,16,2)*	0.25	CNN 5	125	30.1	0:42
12	A-(3,9,16,2,10,11)*	0.25	CNN 5	125	48.3	0:24
13	A-(3,9,16,2,10,11)*	0.25	CNN 7	125	32.2	0:16
14	A-(3,9,16,2,10,11)**	0.25	CNN 8	125	40.3	0:23
15	A-(3,9,16,2,10,11,6,8,14)**	0.25	CNN 8	125	42.4	0:18
16	A-(3,9,16,2,10,11,6,8,14)**	0.25	CNN 8	125	47.1	0:12
17	A-(3,9,16,2,10,11,6,8,14,12)**	0.25	CNN 8	125	55.3	0:13
18	A-(3,9,16,2,10,11,6,8,14,12,1)**	0.25	CNN 8	125	60.4	0:10
19	A-(3,9,16,2,10,11,6,8,14,12,1)**	0.25	CNN 9	125	50.8	0:11
20	A-(3,9,16,2,10,11,6,8,14,12,1)**	0.05	CNN 9	125	99.1	0:14
9	A-(3,9,16,2,10,11)**	0.05	CNN 9	125	97.5	0:32
22	A-(3,9,16,2,10,11)**	0.14	CNN 9	125	97.7	0:31

TABLA 5.7. RESULTADOS DE LOS EXPERIMENTOS

La Tabla 5.8 muestra las estadísticas obtenidas fruto de la experimentación realizada en el conjunto de datos propuesto. Como se puede observar, la media de *accuracy* global obtenida es significativamente menor que el mejor valor. Esto es debido a la diferencia obtenida en los resultados al realizar el incremento del número de instancias de entrenamiento. Provocando este cambio resultados notablemente mejores en los últimos experimentos en comparación con los identificados a lo largo del proyecto.

Algunas de las características relevantes que nos muestra la Tabla 5.8 sobre la experimentación, es que el *batch size* con valor 125 ofreció mejores resultados que el uso de *batch size* con valores más altos. De la misma manera, una reducción de las clases utilizadas, proporcionó mejores resultados de *accuracy*. El modelo elegido, será el que proporciona mejores resultados de *accuracy* reduciendo lo mínimo posible el número de clases utilizadas.

Como la Tabla 5.8 muestra, se han realizado casi treinta horas de experimentación, de las cuales sólo se han registrado en la Tabla 5.7 los más relevantes para comprender la evolución. Los experimentos restantes, se han denominado como experimentos no válidos, siendo el significado de este término el mismo que se comentó en capítulo anterior.

Número de experimentos		Tiempo de experimentación		Accuracy	
Totales	Válidos	Total	Válida	Media	Mejor valor
53	22	29 horas y 28 minutos	15 horas y 12 minutos	45.4	97.7

TABLA 5.8. RESUMEN DE LA EXPERIMENTACIÓN

5.4. Modelo final

El modelo elegido, y por lo tanto el que se usará en futuras pruebas, será el modelo con mejor *accuracy* global, marcado en negrita en la Tabla 5.7.

En la Figura 5.5, se puede observar como se ha comportado dicho modelo durante el entrenamiento. La gráfica presente muestra como ha evolucionado el modelo en cada uno de los steps. Una de las características más relevantes que se pueden observar en ambas gráficas de la Figura 5.5, es que tanto en validación como en test las dos curvas acaban convergiendo al mismo resultado. Esto es muy interesante, ya que en términos generales, los resultados en el entrenamiento tienden a ser mejores que los usados en validación. Esto nos indica que el modelo ha conseguido generalizar el problema gratamente para ambos conjuntos de datos, lo que explica los buenos resultados que se pueden observar en la matriz de confusión de la derecha en la Figura 5.6.

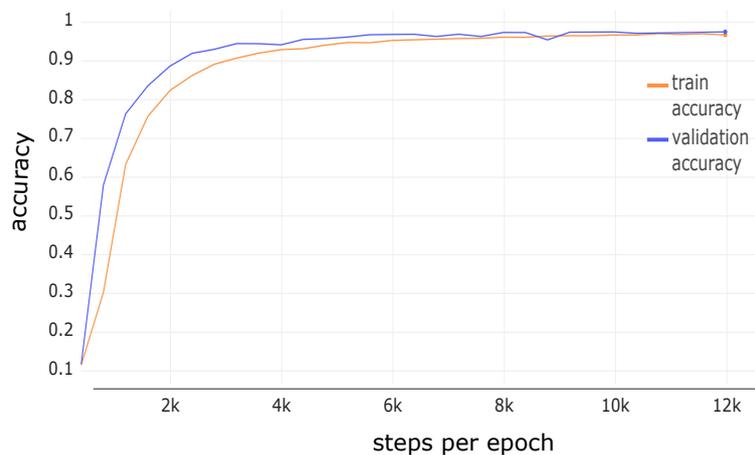


Fig. 5.5. Evolución en el entrenamiento del modelo

En la Figura 5.6 se puede observar dos matrices de confusión, la izquierda pertenece al primer experimento, mientras que la derecha tiene los datos del mejor modelo. De esta manera, se puede analizar la evolución conseguida gracias al entrenamiento realizado.

Mientras que en la imagen de la izquierda de la Figura 5.6 no se percibe una diagonal, y por lo tanto no se ha aprendido a resolver el problema para ninguna de las clases, en la de la derecha se observa una diagonal marcada con colores oscuros. Esta diagonal nos presenta el porcentaje de exactitud o *accuracy* obtenido por clase, cuando un porcentaje es más alto, tendrá un color verde más oscuro, y si en cambio los resultados son bajos, el color tiende a blanco.

La Figura 5.6 muestra en su izquierda la matriz de confusión del primer experimento, y en su derecha la matriz de confusión del modelo final. En el primer experimento, el modelo no aprendió ninguna clase, por eso no se ve ningún color más marcado, y toda la matriz contiene valores dispersos en la imagen izquierda de la Figura 5.6. Tras el proceso

de experimentación y ajuste realizado, mediante los cambios que han sido presentados en la Tabla 5.7, se ha conseguido evolucionar desde unas predicciones en las que se demostraba que el modelo no había conseguido aprender el problema, a una matriz definida donde se ha aprendido a reconocer cada una de las clases propuestas.

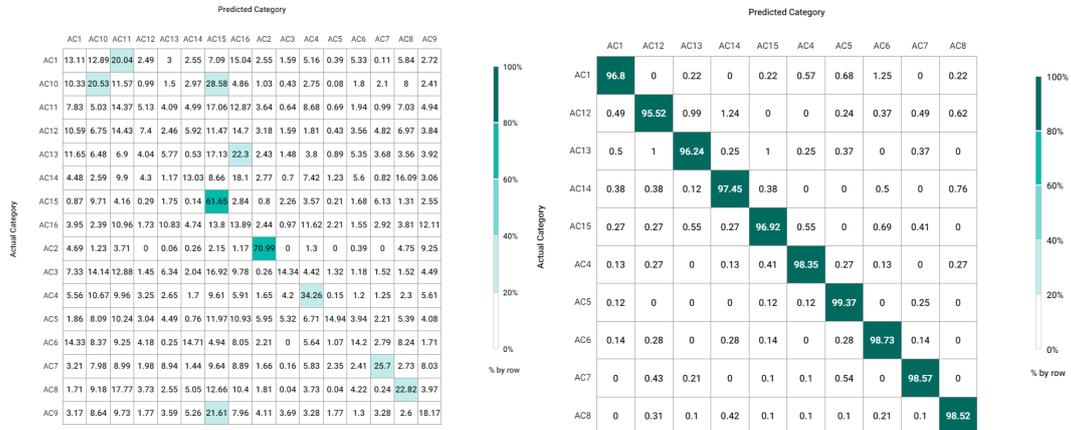


Fig. 5.6. Evolución en la matriz de confusión tras varios experimentos

La Figura 5.6 muestra de manera clara la evolución obtenida a través de los experimentos. Además, en la imagen de la derecha de la figura, también se muestra de manera individual los resultados obtenidos para cada clase. Como se puede observar, el modelo ha generalizado el problema, y ha conseguido clasificar correctamente más de un 97.7% de las imágenes propuestas en el conjunto de test.

Finalmente, con el fin de ilustrar el funcionamiento interno del modelo final, la Figura 5.7 muestra una representación en 3D de la arquitectura CNN 9. Se puede observar tanto las capas de convolución como sus dimensiones, es decir, el número de filtros que contienen. Al final de la red se observan las 10 clases que finalmente están contempladas en el modelo.

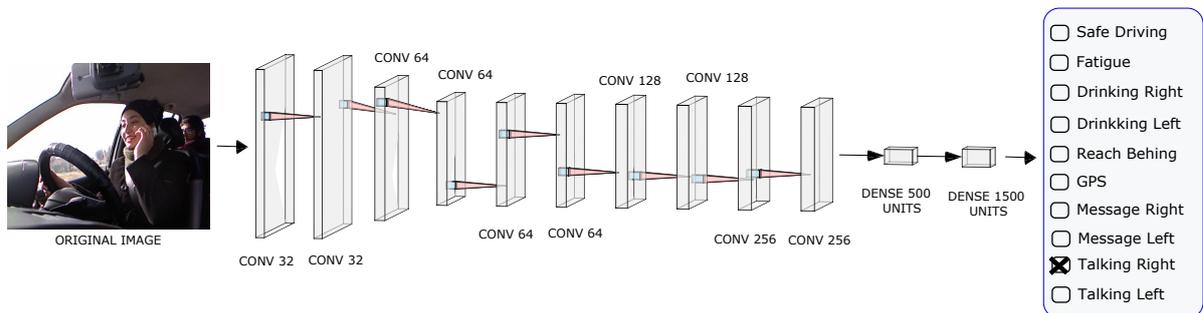


Fig. 5.7. Representación en 3D de la CNN

Se puede concluir que, la reducción en el número de clases, y del *batch size*, han resultado clave para que el modelo consiga generalizar el problema. Consiguiendo poder predecir correctamente imágenes dentro de 10 clases, frente a las 16 que inicialmente contenía el conjunto de datos.

6. INTEGRACIÓN Y PRUEBAS DEL ADAS

6.1. Integración de modelos de aprendizaje automático

Para crear un sistema con los modelos de redes convolucionales que han sido desarrollados para este proyecto, se ha realizado una integración de los modelos en un programa para así poder evaluar el sistema y su comportamiento.

En la Figura 6.1 se puede observar el diagrama de flujo del programa que ha sido implementado. El sistema recoge la imagen obtenida a través de una cámara, y a continuación realiza la detección facial en la imagen. Si se detecta una o más caras en la imagen, se procede a preprocesar la imagen para que sirva de entrada tanto al modelo de detección emociones, como al de comportamientos. Finalmente, la imagen es editada con la información obtenida a través de los modelos, y proporcionada como salida al sistema.

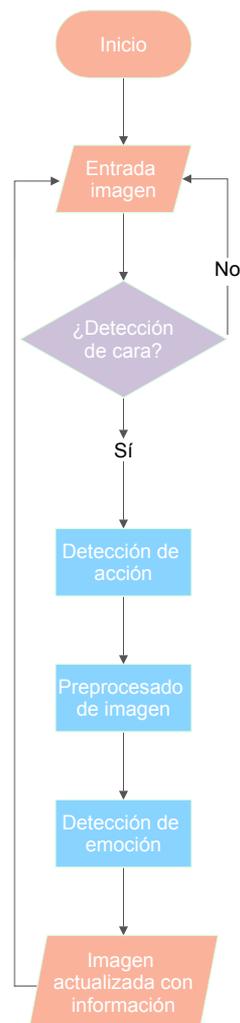


Fig. 6.1. Diagrama de flujo del sistema

6.2. Pruebas unitarias del sistema

En este apartado se especificarán las pruebas realizadas para evaluar el sistema desarrollado en este proyecto. El objetivo de estas pruebas será evaluar el sistema diseñado en base a los requisitos que fueron especificados al inicio del proyecto. Para describir las pruebas se utilizará la plantilla mostrada en la Tabla 6.1.

Para analizar el sistema, se ha hecho uso de instancias de test proporcionadas por el conjunto de datos *Multimodal Multiview and Multispectral Driver Action Dataset (3MDAD)*.

PU-XX	
Descripción	
Objetivo	
Resultado esperado	
Resultado obtenido	
Requisitos asociados	

TABLA 6.1. PLANTILLA DE TABLA DE PRUEBAS UNITARIAS

El la Tabla 6.1 se puede observar los siguientes campos:

- Identificador: El primer campo, situado en la parte superior de la tabla, corresponde al identificador de la prueba unitaria. Este campo tendrá un valor único para cada una de las pruebas realizadas, y estará compuesto por las siglas PU-XX donde:
 - Las siglas PU corresponden a Prueba Unitaria.
 - Las letras XX determinarán el número de requisito.
- Descripción. Este campo contendrá un breve resumen de las características de la prueba realizada.
- Objetivo. Se especifica la finalidad de realizar dicha prueba.
- Resultado esperado. Se indicará el comportamiento esperado por el sistema en base a los requisitos definidos.
- Resultado obtenido. Indica el resultado real obtenido por el sistema desarrollado.
- Requisitos asociados. Este campo indica que requisitos funcionales abarca la prueba realizada.

La matriz de trazabilidad mostrada en la Tabla 6.22, asocia a cada prueba un requisito que será el que se estará validando en cada caso.

PU-01	
Descripción	Se inicia el sistema y se comprueba la conexión con la cámara
Objetivo	Comprobar que el sistema se conecta con el dispositivo hardware asociado, en este caso la cámara del PC.
Resultado esperado	Se muestran por pantalla las imágenes recogidas por la cámara.
Resultado obtenido	Se muestran por pantalla las imágenes recogidas por la cámara.
Requisitos asociados	RF-01

TABLA 6.2. PRUEBA UNITARIA PU-01

PU-02	
Descripción	El sistema procesa la imagen y la proporciona como entrada a los modelos de aprendizaje automático.
Objetivo	Comprobar que el sistema preprocesa correctamente las imágenes recibidas a través de la cámara.
Resultado esperado	Los modelos de Aprendizaje Automático reciben y procesan correctamente las imágenes proporcionadas.
Resultado obtenido	Los modelos de Aprendizaje Automático reciben y procesan correctamente las imágenes proporcionadas.
Requisitos asociados	RF-02

TABLA 6.3. PRUEBA UNITARIA PU-02

PU-03	
Descripción	Se proporciona una imagen previamente preprocesada a los modelos de aprendizaje automático y se analiza el resultado proporcionado por estos.
Objetivo	Comprobar que los modelos proporcionan una predicción en base a la imagen introducida como entrada.
Resultado esperado	Los modelos de Aprendizaje Automático proporcionan una salida cuando se les introduce como entrada una imagen.
Resultado obtenido	Los modelos de Aprendizaje Automático proporcionan una salida cuando se les introduce como entrada una imagen.
Requisitos asociados	RF-03

TABLA 6.4. PRUEBA UNITARIA PU-03

PU-04	
Descripción	Se introduce un conjunto de imágenes y se analiza si el sistema detecta la cara del conductor del vehículo.
Objetivo	Comprobar que el sistema detecta correctamente al conductor del vehículo.
Resultado esperado	El sistema detecta la cara del conductor del vehículo.
Resultado obtenido	El sistema detecta la cara del conductor del vehículo.
Requisitos asociados	RF-04

TABLA 6.5. PRUEBA UNITARIA PU-04

PU-05	
Descripción	Se introduce un ejemplo de un conductor irritado y se analiza si el sistema detecta dicha emoción.
Objetivo	Comprobar que el sistema detecta correctamente el estado de ánimo del conductor del vehículo.
Resultado esperado	El sistema define irritado o como la emoción presente en el conductor.
Resultado obtenido	El sistema define irritado o como la emoción presente en el conductor.
Requisitos asociados	RF-05

TABLA 6.6. PRUEBA UNITARIA PU-05

PU-06	
Descripción	Se introduce un ejemplo de un conductor contento y se analiza si el sistema detecta dicha emoción.
Objetivo	Comprobar que el sistema detecta correctamente el estado de ánimo del conductor del vehículo.
Resultado esperado	El sistema define contento como la emoción presente en el conductor.
Resultado obtenido	El sistema define contento como la emoción presente en el conductor.
Requisitos asociados	RF-06

TABLA 6.7. PRUEBA UNITARIA PU-06

PU-07	
Descripción	Se introduce un ejemplo de un conductor sorprendido y se analiza si el sistema detecta dicha emoción.
Objetivo	Comprobar que el sistema detecta correctamente el estado de ánimo del conductor del vehículo.
Resultado esperado	El sistema define sorprendido como la emoción presente en el conductor.
Resultado obtenido	El sistema define sorprendido como la emoción presente en el conductor.
Requisitos asociados	RF-08

TABLA 6.8. PRUEBA UNITARIA PU-07

PU-08	
Descripción	Se introduce un ejemplo de un conductor asustado y se analiza si el sistema detecta dicha emoción.
Objetivo	Comprobar que el sistema detecta correctamente el estado de ánimo del conductor del vehículo.
Resultado esperado	El sistema define miedo como la emoción presente en el conductor.
Resultado obtenido	El sistema define miedo como la emoción presente en el conductor.
Requisitos asociados	RF-08

TABLA 6.9. PRUEBA UNITARIA PU-08

PU-09	
Descripción	Se introduce un ejemplo de un conductor triste y se analiza si el sistema detecta dicha emoción.
Objetivo	Comprobar que el sistema detecta correctamente el estado de ánimo del conductor del vehículo.
Resultado esperado	El sistema define triste como la emoción presente en el conductor.
Resultado obtenido	El sistema define asustado como la emoción presente en el conductor.
Requisitos asociados	RF-09

TABLA 6.10. PRUEBA UNITARIA PU-09

PU-10	
Descripción	Se introduce un ejemplo de un conductor mostrando síntomas de asco y aversión y se analiza si el sistema detecta dicha emoción.
Objetivo	Comprobar que el sistema detecta correctamente el estado de ánimo del conductor del vehículo.
Resultado esperado	El sistema define asco como la emoción presente en el conductor.
Resultado obtenido	El sistema define asco como la emoción presente en el conductor.
Requisitos asociados	RF-10

TABLA 6.11. PRUEBA UNITARIA PU-10

PU-11	
Descripción	Se introduce un ejemplo de un conductor mostrando síntomas de asco y aversión y se analiza si el sistema detecta dicha emoción.
Objetivo	Comprobar que el sistema detecta correctamente el estado de ánimo del conductor del vehículo.
Resultado esperado	El sistema define neutral como la emoción presente en el conductor.
Resultado obtenido	El sistema define neutral como la emoción presente en el conductor.
Requisitos asociados	RF-11

TABLA 6.12. PRUEBA UNITARIA PU-11

PU-12	
Descripción	Se introduce un ejemplo de un conductor mostrando una conducción segura y se analiza si el sistema detecta dicho comportamiento.
Objetivo	Comprobar que el sistema detecta correctamente el comportamiento que está siendo llevado a cabo por el conductor del vehículo.
Resultado esperado	El sistema define conducción segura como el comportamiento presente en el conductor.
Resultado obtenido	El sistema define conducción segura como el comportamiento presente en el conductor.
Requisitos asociados	RF-12

TABLA 6.13. PRUEBA UNITARIA PU-12

PU-13	
Descripción	Se introduce un ejemplo de un conductor utilizando el GPS y se analiza si el sistema detecta dicho comportamiento.
Objetivo	Comprobar que el sistema detecta correctamente el comportamiento que está siendo llevado a cabo por el conductor del vehículo.
Resultado esperado	El sistema define uso del GPS como el comportamiento presente en el conductor.
Resultado obtenido	El sistema define uso del GPS como el comportamiento presente en el conductor.
Requisitos asociados	RF-13

TABLA 6.14. PRUEBA UNITARIA PU-13

PU-14	
Descripción	Se introduce un ejemplo de un conductor distraído mandando un mensaje de texto y se analiza si el sistema detecta dicho comportamiento.
Objetivo	Comprobar que el sistema detecta correctamente el comportamiento que está siendo llevado a cabo por el conductor del vehículo.
Resultado esperado	El sistema define envío de mensaje de texto como el comportamiento presente en el conductor.
Resultado obtenido	El sistema define envío de mensaje de texto como el comportamiento presente en el conductor.
Requisitos asociados	RF-14

TABLA 6.15. PRUEBA UNITARIA PU-14

PU-15	
Descripción	Se introduce un ejemplo de un conductor distraído realizando una llamada telefónica y se analiza si el sistema detecta dicho comportamiento.
Objetivo	Comprobar que el sistema detecta correctamente el comportamiento que está siendo llevado a cabo por el conductor del vehículo.
Resultado esperado	El sistema define llamada telefónica como el comportamiento presente en el conductor.
Resultado obtenido	El sistema define llamada telefónica como el comportamiento presente en el conductor.
Requisitos asociados	RF-15

TABLA 6.16. PRUEBA UNITARIA PU-15

PU-16	
Descripción	Se introduce un ejemplo de un conductor cansado que muestra síntomas de somnolencia realizando una llamada telefónica y se analiza si el sistema detecta dicho comportamiento.
Objetivo	Comprobar que el sistema detecta correctamente el comportamiento que está siendo llevado a cabo por el conductor del vehículo.
Resultado esperado	El sistema define conductor cansado como el estado presente en el conductor.
Resultado obtenido	El sistema define conductor cansado como el estado presente en el conductor.
Requisitos asociados	RF-16

TABLA 6.17. PRUEBA UNITARIA PU-16

PU-17	
Descripción	Se introduce un ejemplo de un conductor distraído bebiendo y se analiza si el sistema detecta dicho comportamiento.
Objetivo	Comprobar que el sistema detecta correctamente el comportamiento que está siendo llevado a cabo por el conductor del vehículo.
Resultado esperado	El sistema define conductor bebiendo como el estado presente en el conductor.
Resultado obtenido	El sistema define conductor bebiendo como el comportamiento presente en el conductor.
Requisitos asociados	RF-17

TABLA 6.18. PRUEBA UNITARIA PU-17

PU-18	
Descripción	Se introduce un ejemplo de un conductor distraído mirando atrás y se analiza si el sistema detecta dicho comportamiento.
Objetivo	Comprobar que el sistema detecta correctamente el comportamiento que está siendo llevado a cabo por el conductor del vehículo.
Resultado esperado	El sistema define conductor distraído mirando atrás como el estado presente en el conductor.
Resultado obtenido	El sistema define conductor mirando atrás como el comportamiento presente en el conductor.
Requisitos asociados	RF-18

TABLA 6.19. PRUEBA UNITARIA PU-18

PU-19	
Descripción	Se introduce una imagen y se analiza el resultado de la emoción mostrado por pantalla.
Objetivo	Comprobar que el sistema proporciona la información pertinente a las emociones por pantalla.
Resultado esperado	El sistema muestra la información relevante a las emociones en la propia imagen del conductor.
Resultado obtenido	El sistema muestra la información relevante a las emociones en la propia imagen del conductor.
Requisitos asociados	RF-19

TABLA 6.20. PRUEBA UNITARIA PU-19

PU-20	
Descripción	Se introduce una imagen y se analiza el resultado del comportamiento mostrado por pantalla.
Objetivo	Comprobar que el sistema proporciona la información pertinente al comportamiento del conductor por pantalla.
Resultado esperado	El sistema muestra la información relevante al comportamiento del conductor en la propia imagen del conductor.
Resultado obtenido	El sistema muestra la información relevante al comportamiento del conductor en la propia imagen del conductor.
Requisitos asociados	RF-20

TABLA 6.21. PRUEBA UNITARIA PU-20

	PU-01	PU-02	PU-03	PU-04	PU-05	PU-06	PU-07	PU-08	PU-09	PU-10	PU-11	PU-12	PU-13	PU-14	PU-15	PU-16	PU-17	PU-18	PU-19	PU-20
RF-01	X																			
RF-02		X																		
RF-03			X																	
RF-04				X																
RF-05					X															
RF-06						X														
RF-07							X													
RF-08								X												
RF-09									X											
RF-10										X										
RF-11											X									
RF-12												X								
RF-13													X							
RF-14														X						
RF-15															X					
RF-16																X				
RF-17																	X			
RF-18																		X		
RF-19																			X	
RF-20																				X

TABLA 6.22. MATRIZ DE TRAZABILIDAD ENTRE REQUISITOS Y PRUEBAS UNITARIAS

6.3. Resultados de las pruebas

En este apartado se mostrarán algunos de los resultados obtenidos producto de las pruebas realizadas.

Las imágenes que se proporcionan en esta sección son producto de aplicar el sistema desarrollado a un conjunto de imágenes obtenidas del conjunto de datos 3MDAD.

Como se puede observar en la Figura 6.2, el sistema rodea con un cuadrado la cara del conductor cuando la detecta en la imagen. De la misma manera se presentan los resultados de los modelos desarrollados alrededor del cuadrado. En la parte superior se especifica el comportamiento del conductor, mientras que en la parte derecha se puede ver el porcentaje que le otorga a cada una de las emociones que presenta este. En el caso de las emociones, se muestra la probabilidad obtenida por el modelo para cada una de las clases.

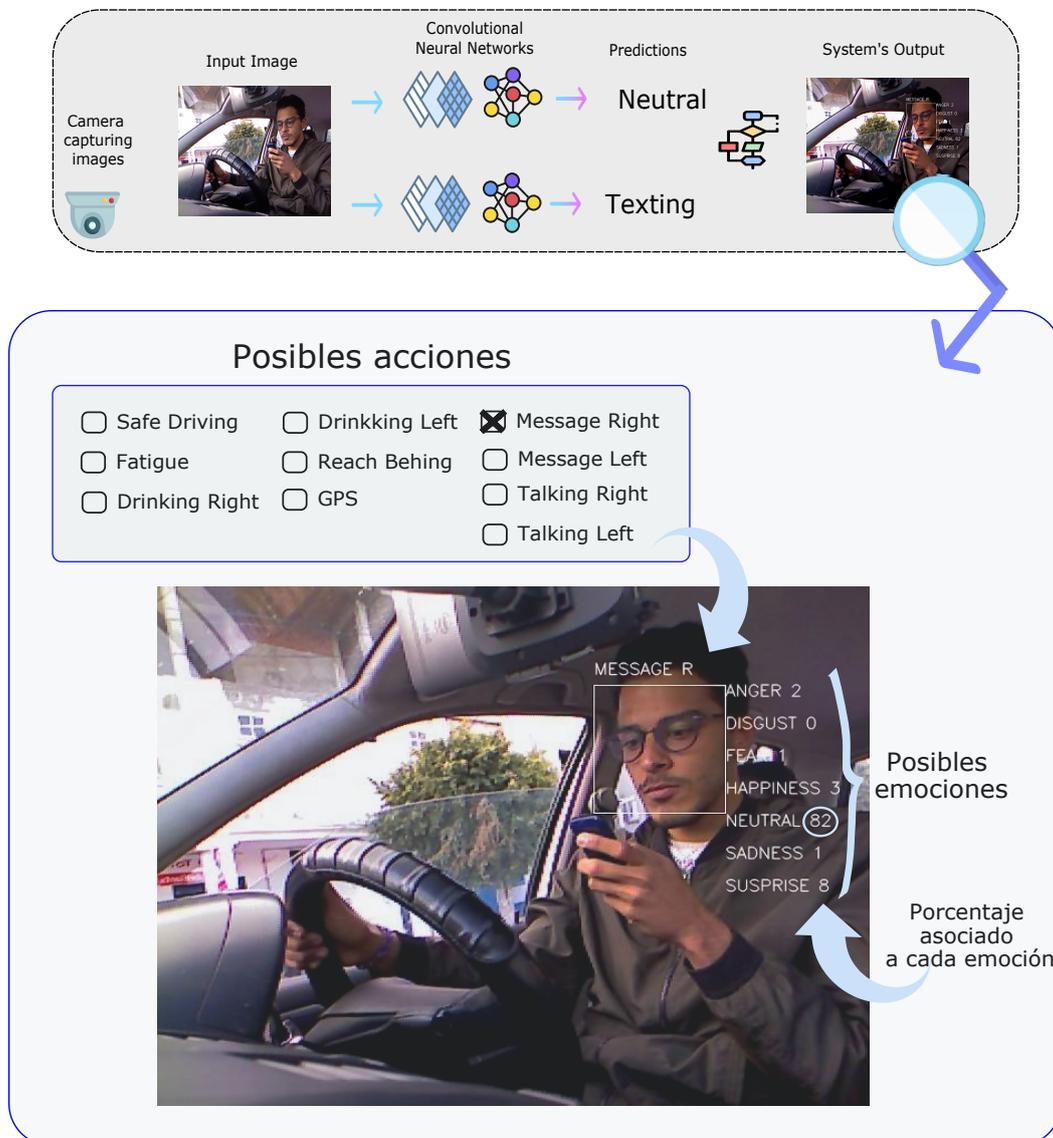


Fig. 6.2. Explicación del resultado del sistema

En la Figura 6.2 se puede observar como el modelo predice la categoría “neutral” con un 90 %, siendo esta la clase con porcentaje más alto, y por lo tanto, la predicción final del modelo.

Se ha decidido presentar de manera visual todas las emociones, y sus resultados, para así poder tener más información cuando se presenten casos de ambigüedad en las expresiones faciales. Por otro lado, para el comportamiento del conductor, se informa solo de la acción que esta tomando en cada momento. En la Figura 6.2 se puede observar como el modelo muestra “message r”, indicando que el conductor esta mandando un mensaje de texto con la mano derecha.

Finalmente, la siguiente batería de imágenes que se puede observar en la Figura 6.3 proporciona varios posibles escenarios, combinando diferentes acciones, con emociones en el conductor.

Las instancias presentes en la Figura 6.3 han sido proporcionadas al realizar las pruebas unitarias del sistema. En estas se muestra a los sujetos realizando diferentes acciones, como manejar el GPS o utilizar el teléfono, además también se pueden observar diferentes reacciones en los conductores como tristeza o estados como fatiga.

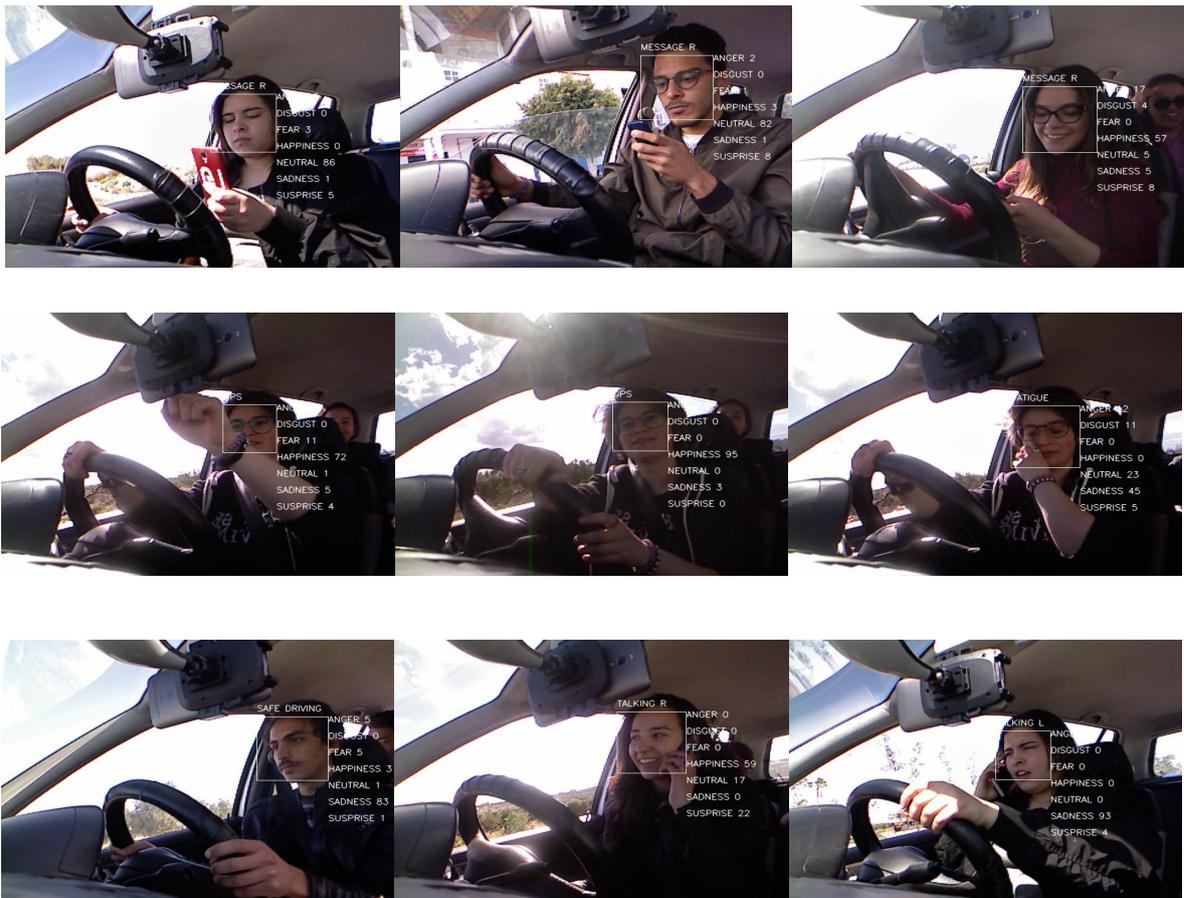


Fig. 6.3. Selección de resultados de las pruebas del sistema

7. GESTIÓN DEL PROYECTO

7.1. Planificación

En esta sección, se desglosarán las tareas realizadas durante el desarrollo del proyecto. Para cada tarea, se han detallado las fechas durante las que se ha prolongado la actividad, así como las horas que ha supuesto.

En la Tabla 7.1, se pueden observar cada una de las tareas que ha requerido el proyecto, así como su duración en horas. Además, se puede observar que algunas de las tareas, como puede ser la solicitud de conjuntos de datos, se prolongan durante prácticamente todo el desarrollo del proyecto. Esto se debe a que los conjuntos se han solicitado a otros investigadores vía e-mail, lo que ha implicado que hasta el final del proyecto se recibieran respuestas a las solicitudes con nuevos datos.

Actividad	Fecha de inicio	Fecha de finalización	Duración(Horas)
Planteamiento del proyecto	24/02/2020	18/03/2020	60
Estudio sobre las emociones en conductores	24/02/2020	26/02/2020	10
Estudio sobre comportamientos en conductores	27/02/2020	28/02/2020	5
Estudio sobre los asistentes de conducción	29/02/2020	02/03/2020	10
Estudio sobre los datos disponibles	03/03/2020	09/03/2020	20
Estudio sobre las redes de neuronas convolucionales	10/03/2020	12/03/2020	5
Estudio sobre las redes convolucionales aplicadas a emociones	13/03/2020	15/03/2020	5
Estudio sobre las redes convolucionales aplicadas a conductores	16/03/2020	18/03/2020	5
Análisis del problema	19/03/2020	22/03/2020	10
Diseño del problema	23/03/2020	27/03/2020	10
Implementación	10/03/2020	29/06/2020	205
Solicitud de conjuntos de datos	10/03/2020	20/06/2020	15
Preprocesado de datos	15/03/2020	20/06/2020	40
Desarrollo del modelo de reconocimiento de emociones	20/03/2020	28/03/2020	30
Experimentación	29/03/2020	02/05/2020	60
Desarrollo del modelo de reconocimiento de comportamientos	03/05/2020	10/05/2020	15
Experimentación	11/05/2020	31/05/2020	30
Integración de modelos	08/06/2020	29/06/2020	15
Validación del sistema	30/06/2020	17/07/2020	40
Pruebas del modelo de reconocimiento de emociones	30/06/2020	03/07/2020	15
Pruebas del modelo de reconocimiento de comportamientos	04/07/2020	06/07/2020	10
Pruebas de la integración de los modelos	07/07/2020	17/07/2020	15
Redacción de la memoria del proyecto	01/06/2020	27/08/2020	80

TABLA 7.1. PLANIFICACIÓN DEL PROYECTO

En el diagrama de Gantt, por meses, que se muestra en la Figura 7.1, se puede observar la distribución de las diferentes tareas ejecutadas a lo largo del proyecto. Este diagrama se ha dividido por meses para una mayor legibilidad, sin embargo, en el Anexo A de este trabajo se puede observar un diagrama más detallado dividido por días.

En términos generales, el proyecto se ha prolongado durante 6 meses, dónde la implementación ha ocupado aproximadamente un 50 % de las horas destinadas al proyecto. Este valor es ligeramente más elevado del porcentaje de horas que se espera de desarrollo debido a que en estas horas se incluye el tiempo dedicado a la experimentación. Como se ha comentado en capítulos anteriores, algunos modelos pueden entrenarse en minutos, mientras que otros se prolongan durante horas, es por eso que el tiempo de experimentación, provoca un aumento considerable en las horas invertidas en la implementación del proyecto.

Diagrama de Gantt - Mensual

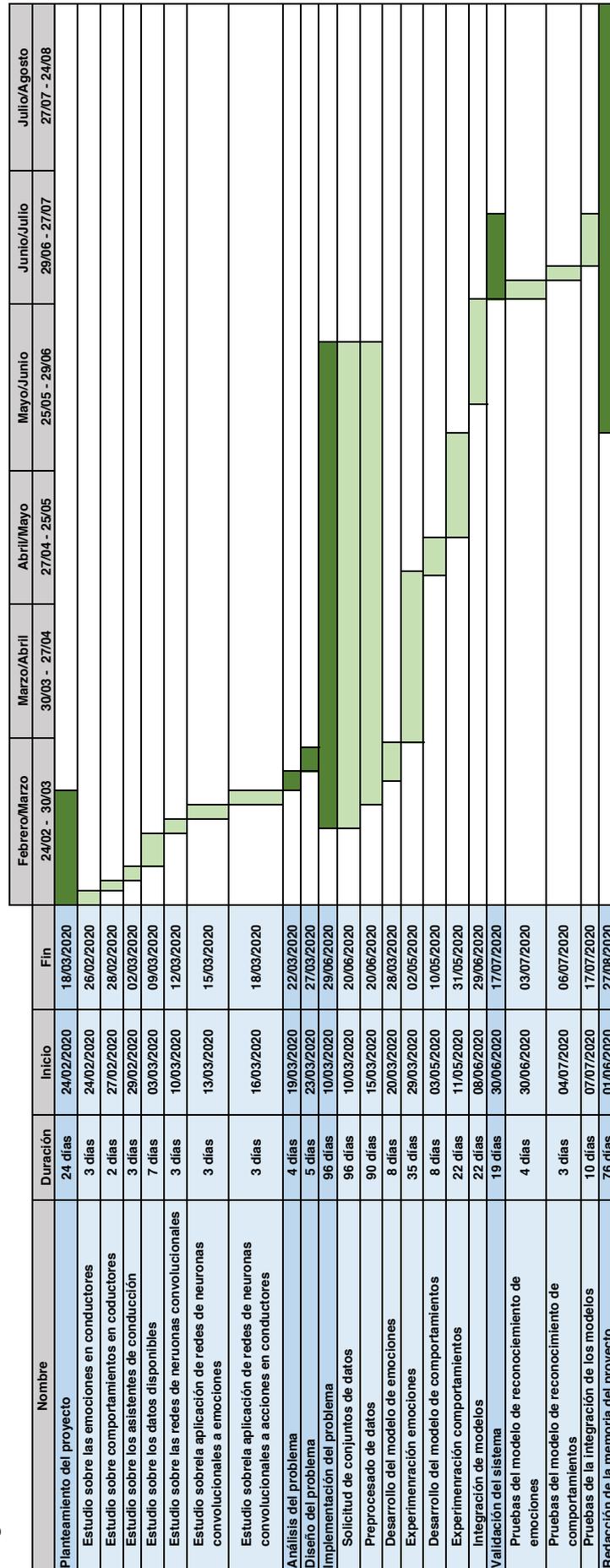


Fig. 7.1. Diagrama de Gantt del proyecto

7.2. Presupuesto

En este apartado del proyecto, se detallarán los recursos necesarios para el desarrollo del proyecto. Estos recursos englobarán tanto los costes de materiales y herramientas, como los costes de personal.

Los recursos necesarios para el desarrollo de este trabajo engloban tanto los recursos de software como los de hardware.

Para la elaboración de un proyecto como el que se ha llevado a cabo, es necesario formar un equipo que contenga diferentes perfiles. Los perfiles necesarios incluyen a un jefe de proyecto que llevará a cabo la planificación y gestión de recursos, así como el estudio inicial del sistema y su contexto. Un perfil experto en Data Scientist encargado de determinar las técnicas y algoritmos de Machine Learning que han de ser utilizadas y de llevar a cabo la experimentación pertinente para la creación de los modelos. Finalmente, se precisará de un desarrollador con conocimientos de Python que llevará a cabo la programación del sistema y las pruebas acorde a los requisitos del sistema propuesto.

En la Tabla 7.2 se puede observar el salario medio en España, estimado en 2020, para cada uno de los perfiles según el buscador de empleo Indeed. Para el cálculo del coste total por perfil, se ha asumido que cada sueldo corresponde a un total de 38 horas trabajadas a la semana, y se ha aplicado la siguiente fórmula:

$$\text{Coste total (€)} = \frac{\text{Salario (€/mes)}}{38 \text{ (horas/semana)} * 4 \text{ (semanas/mes)}} * \text{Duración (horas)}$$

Rol	Nº de empleados	Duración (horas)	Salario (€/mes)	Coste total (€)
Jefe de proyecto	1	80	3.041,5 €[50]	1.600,0 €
Data Scientist	1	190	2.704,5 €[51]	3.380,6 €
Programador Junior	1	55	1.562,0 €[52]	565,0 €
Coste total de personal				5545,6 €

TABLA 7.2. TABLA DE COSTES DE PERSONAL

Por otro lado, además del coste de personal, se han calculado los costes producto de los recursos utilizados para el desarrollo del proyecto. La Tabla 7.3 contiene un desglose de los costes obtenidos tanto para hardware como software. También se han incluido los costes indirectos del proyecto.

Para determinar el coste de cada servicio en el proyecto, se ha utilizado la siguiente fórmula:

$$\text{Coste para el proyecto (€)} = \frac{\text{Coste total (€)}}{\text{Vida útil (meses)}} * \text{Uso (meses)}$$

Producto	Coste total (€)	Uso (meses)	Vida útil (meses)	Coste para el proyecto (€)
Hardware				
PC desarrollo	900,0 €	6	48	112,5 €
Servidor experimentación (Deep Learning)	4.500,0	3	120	112,5 €
Coste total en Hardware				225,5 €
Software				
Python	Gratis	6	-	-
Pycharm	Gratis	6	-	-
Tensorflow	Gratis	6	-	-
Keras	Gratis	6	-	-
Open CV	Gratis	6	-	-
Comet ML	Gratis	6	-	-
Trello	Gratis	6	-	-
GitKraken	Gratis	6	-	-
Overleaf	Gratis	6	-	-
Windows 10	Gratis	6	-	-
Coste total en Software				0 €
Costes indirectos				
Agua y electricidad	35,0 €	6	-	210,0 €
Conexión internet	28,9 €	6	-	173,7 €
Coste total indirecto				383,7 €

TABLA 7.3. COSTES DEL PROYECTO EN HARDWARE, SOFTWARE E INDIRECTOS

Finalmente, el coste total del proyecto incluirá tanto los costes directos en personal y recursos, como los costes indirectos producto del uso de recursos básicos como la electricidad o el servicio de internet. A estos gastos también se le añadirá un margen de riesgo que financiará posibles contratiempos en el transcurso del proyecto, al igual que un porcentaje de beneficios. En la Tabla 7.4 se resume esta información y se proporciona un presupuesto final del proyecto.

Costes del proyecto	
Total costes directos	5.770,7 €
Total costes indirectos	383,7 €
Margen de riesgo (15 %)	865,6 €
Beneficios (15 %)	865,6 €
Base imponible	7.885,6
IVA (21 %)	1.655,9 €
Coste total del proyecto	9.541,5 €

TABLA 7.4. PRESUPUESTO TOTAL DEL PROYECTO

7.3. Impacto socio-económico

El desarrollo de un sistema de ayuda a la conducción, al igual que la integración de muchas otras nuevas tecnologías, tiene un impacto en la sociedad y en la industria. En este apartado, se comentará un posible escenario futuro, consecuencia de la introducción del sistema de ayuda desarrollado para este trabajo en el mercado.

En el primer capítulo de este trabajo, se determinó la relación existente entre los accidentes de tráfico y las distracciones por parte del conductor. La motivación que reside bajo el desarrollo de este sistema es la de acabar, o disminuir, en la medida de lo posible, con los accidentes en la carretera. Al disminuir el número de accidentes, también lo harán por ende los gastos producidos por dichos accidentes. Según un estudio realizado en 2008, la siniestralidad en carretera había producido gastos equivalentes al 2 % del PIB anual en España. Esta cifra equivale a aproximadamente cerca de 16.000 millones de euros. El desarrollo y la implantación del sistema planteado, podría ser causa de un decremento en el gasto por parte del gobierno y de los ciudadanos, mejorando así la economía del país.

Es necesario estudiar como un sistema como el propuesto, afecta tanto a los vehículos, como a las industrias y los servicios relacionados con ellos. En el punto anterior, se comentaban uno de los aspectos positivos de la implantación del sistema propuesto. Si bien, la reducción de accidentes también provocará una posible reducción de carga de trabajo en hospitales y en servicios dedicados a la reparación de automóviles. Esta reducción, no debería ser causa de excesivas pérdidas de ingresos ni causa de cierre en ninguno de estos servicios, ya que ambos poseen ingresos que provienen de otras fuentes independientes a los accidentes de tráfico.

La integración de un sistema de ayuda a la conducción, conlleva la necesidad de personal cualificado capaz de integrar el sistema tanto en vehículos nuevos como los ya existentes. De esta manera, nuevos puestos de trabajo y especializaciones habrán de ser creadas para satisfacer la demanda, mejorando así la economía del país y el bienestar de sus habitantes.

Por otro lado, la introducción en la sociedad de una novedad tecnológica como esta, provocará grandes cambios en la relación existente entre la tecnología, concretamente los sistemas autónomos, y el ser humano. El uso de tecnologías autónomas o inteligentes para mejorar un aspecto de la sociedad, en este caso la reducción de accidentes, podría mejorar la percepción del ser humano a la introducción en la sociedad de estas nuevas tecnologías. Facilitando así, la incorporación de futuras innovaciones, que involucren sistemas inteligentes, en el mercado.

Finalmente, la consecuencia que afectará de manera más directa en la sociedad y sus ciudadanos, será la reducción del número de fatalidades en la carretera. Una reducción en el número de víctimas anuales en la carretera, producirá un aumento en la población mundial y en la esperanza de vida. De la misma manera, se reducirían el número de lesiones permanentes e incapacidades presentes en las víctimas de un accidente.

8. CONCLUSIONES

8.1. Conclusiones generales

Al inicio de este trabajo, se determinaron unos objetivos, incluyendo unos requisitos y unas funcionalidades. Los objetivos que se presentaron al inicio de esta memoria, han sido completados.

Los primeros capítulos de esta memoria plasman el estudio que ha sido realizado, tanto de las emociones y las distracciones en conductores, como su influencia en la conducción. De la misma manera, durante el desarrollo de este trabajo se ha realizado un proceso de recolección de datos, donde el alumno ha buscado y solicitado a varios investigadores el uso de sus conjuntos de imágenes.

Tras cientos de experimentos, se ha conseguido el desarrollo de dos modelos de Deep Learning que conforman un sistema de ayuda a la conducción, y que han sido integrados para recibir imágenes y presentar los resultados. El desarrollo de este sistema contribuye a la emergente línea de investigación en la que los sistemas de ayuda a la conducción están basados en el conductor.

Además de las aportaciones al estado del arte que ha proporcionado esta tesis, también ha contribuido al crecimiento personal y profesional del alumno. A través de este trabajo, el alumno ha aplicado los conocimientos adquiridos durante el grado en bases de datos, aprendizaje automático, aprendizaje profundo y desarrollo en Python. Habiendo así ganado experiencia, tanto en la planificación y desarrollo de un proyecto de ingeniería informática, como en el campo del aprendizaje automático. Por otro lado, el alumno también se ha interesado y ha investigado sobre otras líneas de investigación de la rama de computación, que no habían sido estudiadas en el grado, como es la computación afectiva.

Para concluir, los modelos desarrollados durante este proyecto conforman una parte de un sistema de ayuda a la conducción, y además, podrán ser incorporados en otros sistemas que busquen mejorar la experiencia en la carretera. De la misma manera, este proyecto contribuye a la investigación en la computación afectiva, acercando algunos de los procesos cognitivos más complejos como son las emociones al lenguaje de las máquinas. De la misma manera, contribuye al desarrollo de sistemas inteligentes, y a la investigación en técnicas de Deep Learning.

8.2. Limitaciones y dificultades encontradas

El ámbito de las emociones humanas es un campo complejo y en el que todavía queda pendiente mucha investigación. En este trabajo se ha demostrado la capacidad de detectar algunas de las emociones más básicas a través de una imagen, sin embargo, algunas de

las imágenes resultan ambiguas y difíciles de categorizar, incluso para un ser humano. Si bien merece la pena seguir investigando en el estudio de las emociones a través de expresiones faciales, resulta un reto poder determinar con seguridad la emoción de un individuo utilizando sólo una imagen.

La complejidad de la detección de emociones ha sido una de las dificultades que se ha encontrado en este proyecto, ya que, independientemente de que los resultados obtenidos son muy positivos, no se puede asegurar un cien por cien de aciertos en modelos subjetivos como este. Esto se debe a las múltiples y diversas expresiones faciales que existen, y que son exclusivas en cada individuo.

8.3. Trabajos futuros

En este proyecto se ha abordado desde el planteamiento de un sistema ADAS, hasta el desarrollo de los modelos que lo conformarían y su evaluación en conjunto. Esto deja la puerta abierta a poder utilizar este trabajo como base para futuros proyectos en los que se quiera dar un paso más lejos.

Uno de los posibles proyectos futuros podría ser la implantación del sistema desarrollado en un vehículo, así como el estudio y evaluación del mismo en entornos reales de conducción. Esta implantación deberá tener en cuenta los requisitos de seguridad necesarios para proteger las imágenes que son tomadas del usuario del vehículo. Si bien el prototipo desarrollado no almacena las imágenes, y se conecta directamente a la aplicación nativa de la cámara, el potencial sistema llevado al coche deberá mantener dichas condiciones de seguridad para asegurar la protección de privacidad del usuario.

La base que ha sido desarrollada hasta ahora es funcional y proporciona un alto porcentaje de aciertos, sin embargo, estos resultados están condicionados a que exista luz ambiental suficiente para poder tomar las imágenes. El entrenamiento de un modelo capaz de determinar emociones y comportamientos, a través de imágenes tomadas de noche con baja calidad lumínica, le aportaría versatilidad al sistema ya creado y aumentaría su eficacia en entornos no controlados.

Finalmente, al inicio de este documento se comentaron las razones por las que algunas emociones negativas en la carretera, como la irritación o enfado, eran frecuentes y negativas para la conducción. De la misma manera, se determinó que estas estaban causadas por la situación del tráfico en sí misma. En este proyecto, se ha desarrollado un detector de emociones, incluyendo emociones negativas. La investigación de estímulos que puedan calmar al conductor cuando se detecte, con el modelo desarrollado, un estado de enfado puede ser un trabajo a futuro con aplicaciones muy interesantes en la seguridad vial.

BIBLIOGRAFÍA

- [1] *Automotive Industry - Statistics & Facts* | Statista. dirección: <https://www.statista.com/topics/1487/automotive-industry/>.
- [2] *The number of cars worldwide is set to double by 2040* | World Economic Forum. dirección: <https://www.weforum.org/agenda/2016/04/the-number-of-cars-worldwide-is-set-to-double-by-2040>.
- [3] *Causes of Death - Our World in Data*. dirección: <https://ourworldindata.org/causes-of-death>.
- [4] *Road traffic injuries*. dirección: <https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries>.
- [5] “Unit 2. RISK FACTORS FOR ROAD TRAFFIC INJURIES • 2 21 1”, inf. téc.
- [6] E. Petridou y M. Moustaki, “Human factors in the causation of road traffic crashes”, *European Journal of Epidemiology*, vol. 16, n.º 9, págs. 819-826, 2000. doi: 10.1023/A:1007649804201. dirección: <http://link.springer.com/10.1023/A:1007649804201>.
- [7] *Universal Emotions | What are Emotions?* | Paul Ekman Group. dirección: <https://www.paulekman.com/universal-emotions/> (visitado 01-08-2020).
- [8] P. Ekman, “Basic emotions”, *Handbook of cognition and emotion*, vol. 98, n.º 45-60, pág. 16, 1999.
- [9] J. Russell, “A Circumplex Model of Affect”, *Journal of Personality and Social Psychology*, vol. 39, págs. 1161-1178, dic. de 1980. doi: 10.1037/h0077714.
- [10] C. Darwin, *The expression of the emotions in man and animals*. New York ;D. Appleton y Co., 1916, pág. 406, <https://www.biodiversitylibrary.org/bibliography/4820> — Includes index. dirección: <https://www.biodiversitylibrary.org/item/24064>.
- [11] P. Ekman y col., “Universals and cultural differences in the judgments of facial expressions of emotion.”, *Journal of personality and social psychology*, vol. 53, n.º 4, pág. 712, 1987.
- [12] A. Dzedzickis, A. Kaklauskas y V. Bucinskas, *Human emotion recognition: Review of sensors and methods*, feb. de 2020. doi: 10.3390/s20030592. dirección: www.mdpi.com/journal/sensors.
- [13] *Understanding body language: 7 basic facial emotions*. dirección: https://www.cbc.ca/natureofthings/m%7B%5C_%7Dfeatures/the-seven-universal-emotions-we-wear-on-our-face (visitado 19-07-2020).

- [14] C. Pêcher, C. Lemercier y J. M. Cellier, “Emotions drive attention: Effects on driver’s behaviour”, *Safety Science*, vol. 47, n.º 9, págs. 1254-1259, nov. de 2009. doi: 10.1016/j.ssci.2009.03.011.
- [15] “Effects of emotions on driving behavior”, *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 59, págs. 150-163, nov. de 2018. doi: 10.1016/j.trf.2018.08.012.
- [16] *Tailgating | Is it illegal? | RAC Drive*. dirección: <https://www.rac.co.uk/drive/advice/driving-advice/tailgating-what-you-need-to-know/> (visitado 26-08-2020).
- [17] S. Box, *Transportation Institute releases findings on driver behavior and crash factors | Virginia Tech Daily | Virginia Tech*, 2006. dirección: <https://vtnews.vt.edu/articles/2006/04/2006-237.html> (visitado 01-08-2020).
- [18] *CAA National*. dirección: https://www.caa.ca/17-cao%7B%5C_%7Davi-infographic%7B%5C_%7Deng/ (visitado 19-07-2020).
- [19] *Timeline: Milestones in ADAS Before 2010 | 2020-01-27 | ADAPT Automotive*. dirección: <https://www.adaptautomotive.com/articles/51-timeline-milestones-in-adas-before-2010> (visitado 02-07-2020).
- [20] A. Shaout, D. Colella y S. Awad, “Advanced Driver Assistance Systems - Past, present and future”, *Proc. 7th Int. Computer Engineering Conf*, dic. de 2011. doi: 10.1109/ICENCO.2011.6153935.
- [21] R. (M. R. Johnson, *Advanced Driver Assistance Systems (ADAS) Market 2020-2025*, 2020. dirección: <https://www.openpr.com/news/1953342/advanced-driver-assistance-systems-adas-market-2020-2025%7B%5C#%7Dprid-1953342> (visitado 02-07-2020).
- [22] A. M. Turing, “Computing machinery and intelligence”, en *Parsing the turing test*, Springer, 2009, págs. 23-65.
- [23] *24 Affective Computing Applications / Use Cases in 2020*, 2020. dirección: <https://research.aimultiple.com/affective-computing-applications/> (visitado 19-07-2020).
- [24] *Cousins of Artificial Intelligence | by Seema Singh | Towards Data Science*. dirección: <https://towardsdatascience.com/cousins-of-artificial-intelligence-dda4edc27b55> (visitado 26-08-2020).
- [25] K. Keshari, *Latest Machine Learning Projects to Try in 2019 - Edureka*, 2019. dirección: <https://www.edureka.co/blog/machine-learning-projects/> (visitado 01-08-2020).
- [26] *Artificial Intelligence Demystified*. dirección: <https://www.analyticsvidhya.com/blog/2016/12/artificial-intelligence-demystified/> (visitado 26-08-2020).

- [27] *Perceptrón 5 unidades - Wikimedia Commons*. dirección: https://commons.wikimedia.org/wiki/File:Perceptr%7B%5C'%7Bo%7D%7Dn%7B%5C_%7D5%7B%5C_%7Dunidades.svg (visitado 01-09-2020).
- [28] *Comparación entre un clasificador lineal (A) y uno no-lineal (B)*. dirección: https://www.researchgate.net/figure/Figura-53-Comparacin-entre-un-clasificador-lineal-A-y-uno-no-lineal-B-La-nea%7B%5C_%7Dfig4%7B%5C_%7D310374887/actions%7B%5C#%7Dreference (visitado 11-08-2020).
- [29] V. K. Ojha, A. Abraham y V. Snášel, “Metaheuristic design of feedforward neural networks: A review of two decades of research”, *Engineering Applications of Artificial Intelligence*, vol. 60, págs. 97-116, 2017.
- [30] E. Fallon, L. Murphy, J. Murphy y C. Ma, “TRAWL – A Traffic Route Adapted Weighted Learning Algorithm”, vol. 6649, jun. de 2011, págs. 1-14. DOI: 10.1007/978-3-642-21560-5_1.
- [31] Y. Wang, Y. Li, Y. Song y X. Rong, “Facial Expression Recognition Based on Random Forest and Convolutional Neural Network”, *Information*, vol. 10, pág. 375, nov. de 2019. DOI: 10.3390/info10120375.
- [32] K. Shan, J. Guo, W. You, D. Lu y R. Bie, “Automatic facial expression recognition based on a deep convolutional-neural-network structure”, *Proceedings - 2017 15th IEEE/ACIS International Conference on Software Engineering Research, Management and Applications, SERA 2017*, págs. 123-128, 2017. DOI: 10.1109/SERA.2017.7965717.
- [33] P. Lucey y col., “The extended Cohn-Kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression”, *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops, CVPRW 2010*, n.º July, págs. 94-101, 2010. DOI: 10.1109/CVPRW.2010.5543262.
- [34] J. G. Michael J. Lyons, Miyuki Kamachi, “Japanese Female Facial Expressions (JAFFE), Database of digital images”, n.º January, 1997.
- [35] M. Sundaram, M. Prakash, I. Surenter, N. V. Balaji y S. Kannimuthu, “Facial expression recognition using convolutional neural networks”, *Test Engineering and Management*, vol. 83, págs. 15 075-15 085, 2020.
- [36] *Challenges in Representation Learning: Multi-modal Learning | Kaggle*. dirección: <https://www.kaggle.com/c/challenges-in-representation-learning-multi-modal-learning> (visitado 01-08-2020).
- [37] A. Krizhevsky, I. Sutskever y G. E. Hinton, “Imagenet classification with deep convolutional neural networks”, en *Advances in neural information processing systems*, 2012, págs. 1097-1105.

- [38] B. Ramdhani, E. C. Djamal y R. Ilyas, “2018 International Symposium on Advanced Intelligent Informatics (SAIN).”, *2018 International Symposium on Advanced Intelligent Informatics (SAIN)*, págs. 96-101, 2018.
- [39] E. Pranav, S. Kamal, C. Satheesh Chandran y M. H. Supriya, “Facial Emotion Recognition Using Deep Convolutional Neural Network”, *2020 6th International Conference on Advanced Computing and Communication Systems, ICACCS 2020*, págs. 317-320, 2020. doi: 10.1109/ICACCS48705.2020.9074302. arXiv: arXiv:1910.05602v1.
- [40] L. Nwosu y col., “Deep Convolutional Neural Network for Facial Expression Recognition Using Facial Parts”, *Proceedings - 2017 IEEE 15th International Conference on Dependable, Autonomic and Secure Computing, 2017 IEEE 15th International Conference on Pervasive Intelligence and Computing, 2017 IEEE 3rd International Conference on Big Data Intelligence and Compu*, vol. 2018-Janua, págs. 1318-1321, 2018. doi: 10.1109/DASC-PICom-DataCom-CyberSciTec.2017.213.
- [41] R. Torres, O. Ohashi, E. Carvalho y G. Pessin, “A deep learning approach to detect distracted drivers using a mobile phone”, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10614 LNCS, págs. 72-79, 2017. doi: 10.1007/978-3-319-68612-7{_}9.
- [42] *State Farm Distracted Driver Detection | Kaggle*. dirección: <https://www.kaggle.com/c/state-farm-distracted-driver-detection> (visitado 01-08-2020).
- [43] “Detection of distracted driver using convolutional neural network”, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, vol. 2018-June, págs. 1145-1151, 2018. doi: 10.1109/CVPRW.2018.00150.
- [44] K. Simonyan y A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition”, *CoRR*, vol. abs/1409.1556, 2015.
- [45] V. Sathe, N. Prabhune y A. Humane, “Distracted Driver Detection using CNN and Data Augmentation Techniques”, *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 7, n.º 4, págs. 130-135, 2018. doi: 10.17148/IJARCCCE.2018.7428. dirección: <https://ijarcce.com/upload/2018/april-18/IJARCCCE%2028.pdf>.
- [46] *Driver Monitoring Systems’ Comprehensive Solution to Driver Safety Through Deep Learning*. dirección: <https://blog.affectiva.com/driver-monitoring-systems-comprehensive-solution-to-driver-safety-through-deep-learning> (visitado 01-08-2020).
- [47] *Affectiva*. dirección: <https://www.affectiva.com/> (visitado 01-08-2020).
- [48] *KDEF & AKDEF*. dirección: <https://www.kdef.se/index.html> (visitado 28-08-2020).

- [49] I. Jegham, A. Ben Khalifa, I. Alouani y M. A. Mahjoub, “A novel public dataset for multimodal multiview and multispectral driver distraction analysis: 3MDAD”, *Signal Processing: Image Communication*, vol. 88, pág. 115 960, ago. de 2020. doi: 10.1016/j.image.2020.115960.
- [50] *Salarios para empleos de Jefe de proyecto en España | Indeed.com*. dirección: <https://es.indeed.com/salaries/jefe-de-proyecto-Salaries> (visitado 06-09-2020).
- [51] *Salarios para empleos de Data scientist en España | Indeed.com*. dirección: <https://es.indeed.com/salaries/data-scientist-Salaries> (visitado 06-09-2020).
- [52] *Salarios para empleos de Programador/a junior en España | Indeed.com*. dirección: <https://es.indeed.com/salaries/programador-junior-Salaries> (visitado 06-09-2020).
- [53] *7 Emotional Symptoms of Stress: What They Are and What You Can Do*. dirección: <https://www.healthline.com/health/emotional-symptoms-of-stress#anxiety>.

9. ANEXO A

Diagrama de Gantt

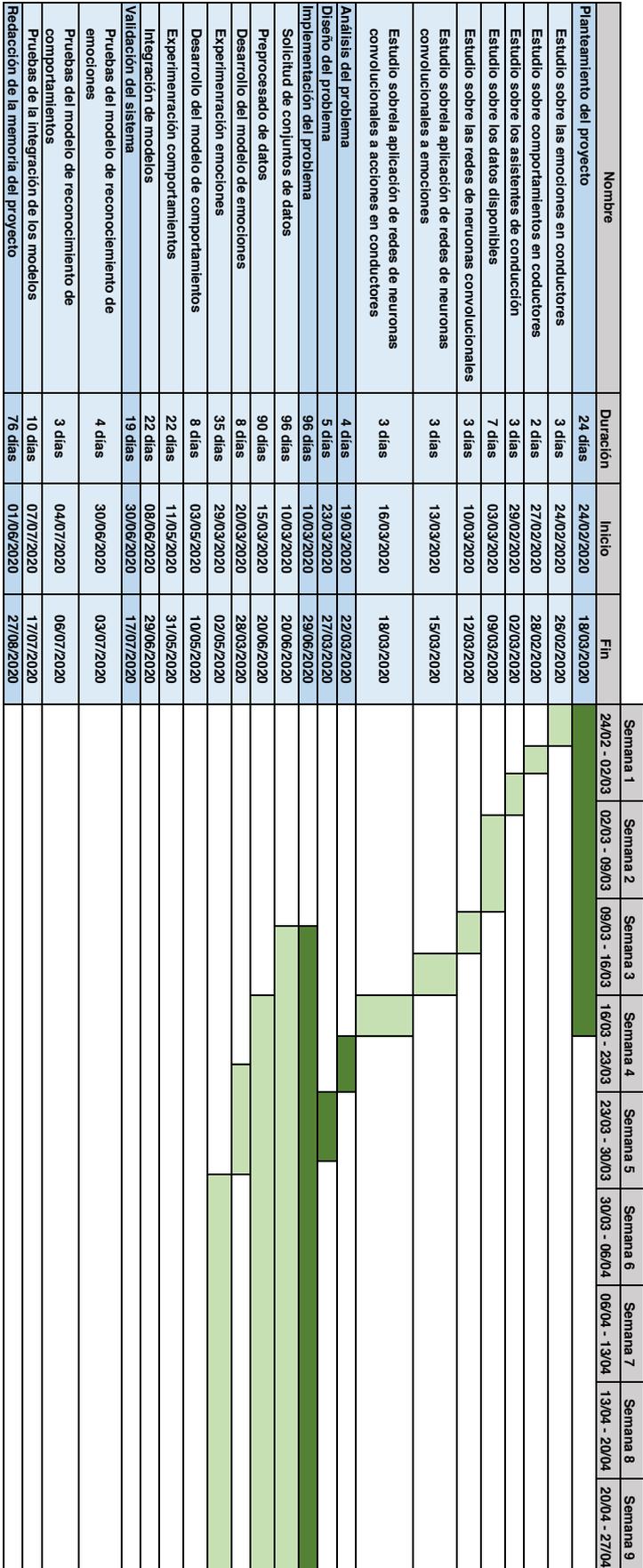


Fig. 9.1. Diagrama de Gantt del proyecto por días 1

Diagrama de Gantt

Nombre	Duración	Inicio	Fin	Semana 10	Semana 11	Semana 12	Semana 13	Semana 14	Semana 15	Semana 16	Semana 17	Semana 18	Semana 19
				27/04 - 04/05	04/05 - 11/05	11/05 - 18/05	18/05 - 25/05	25/05 - 01/06	01/06 - 08/06	08/06 - 15/06	15/06 - 22/06	22/06 - 29/06	29/06 - 06/07
Planteamiento del proyecto	24 días	24/02/2020	18/03/2020										
Estudio sobre las emociones en conductores	3 días	24/02/2020	28/02/2020										
Estudio sobre comportamientos en conductores	2 días	27/02/2020	28/02/2020										
Estudio sobre los asistentes de conducción	3 días	29/02/2020	02/03/2020										
Estudio sobre los datos disponibles	7 días	03/03/2020	09/03/2020										
Estudio sobre las redes de neuronas convolucionales	3 días	10/03/2020	12/03/2020										
Estudio sobre la aplicación de redes de neuronas convolucionales a emociones	3 días	13/03/2020	15/03/2020										
Estudio sobre la aplicación de redes de neuronas convolucionales a acciones en conductores	3 días	16/03/2020	18/03/2020										
Análisis del problema	4 días	19/03/2020	22/03/2020										
Diseño del problema	5 días	23/03/2020	27/03/2020										
Implementación del problema	96 días	10/03/2020	29/06/2020										
Solicitud de conjuntos de datos	96 días	10/03/2020	20/06/2020										
Preprocesado de datos	90 días	15/03/2020	20/06/2020										
Desarrollo del modelo de emociones	8 días	20/03/2020	28/03/2020										
Experimentación emociones	35 días	29/03/2020	02/05/2020										
Desarrollo del modelo de comportamientos	8 días	03/05/2020	10/05/2020										
Experimentación comportamientos	22 días	11/05/2020	31/05/2020										
Integración de modelos	22 días	08/06/2020	29/06/2020										
Validación del sistema	19 días	30/06/2020	17/07/2020										
Pruebas del modelo de reconocimiento de emociones	4 días	30/06/2020	03/07/2020										
Pruebas del modelo de reconocimiento de comportamientos	3 días	04/07/2020	06/07/2020										
Pruebas de la integración de los modelos	10 días	07/07/2020	17/07/2020										
Redacción de la memoria del proyecto	76 días	01/06/2020	27/08/2020										

Fig. 9.2. Diagrama de Gantt del proyecto por días 2

Diagrama de Gantt

Nombre	Duración	Inicio	Fin	Semana 20	Semana 21	Semana 22	Semana 23	Semana 24	Semana 25	Semana 26	Semana 26
Plantamiento del proyecto	24 días	24/02/2020	18/03/2020	06/07 - 13/07	13/07 - 20/07	20/07 - 27/07	27/07 - 03/08	03/08 - 10/08	10/08 - 17/08	17/08 - 24/08	17/08 - 24/08
Estudio sobre las emociones en conductores	3 días	24/02/2020	26/02/2020								
Estudio sobre comportamientos en conductores	2 días	27/02/2020	28/02/2020								
Estudio sobre los asistentes de conducción	3 días	29/02/2020	02/03/2020								
Estudio sobre los datos disponibles	7 días	03/03/2020	09/03/2020								
Estudio sobre las redes de neuronas convolucionales	3 días	10/03/2020	12/03/2020								
Estudio sobre la aplicación de redes de neuronas convolucionales a emociones	3 días	13/03/2020	15/03/2020								
Estudio sobre la aplicación de redes de neuronas convolucionales a acciones en conductores	3 días	16/03/2020	18/03/2020								
Análisis del problema	4 días	19/03/2020	22/03/2020								
Diseño del problema	5 días	23/03/2020	27/03/2020								
Implementación del problema	96 días	10/03/2020	29/06/2020								
Solicitud de conjuntos de datos	96 días	10/03/2020	20/06/2020								
Preprocesado de datos	90 días	15/03/2020	20/06/2020								
Desarrollo del modelo de emociones	8 días	20/03/2020	28/03/2020								
Experimentación emociones	35 días	29/03/2020	02/05/2020								
Desarrollo del modelo de comportamientos	8 días	03/05/2020	10/05/2020								
Experimentación comportamientos	22 días	11/05/2020	31/05/2020								
Integración de modelos	22 días	08/06/2020	29/06/2020								
Validación del sistema	19 días	30/06/2020	17/07/2020								
Pruebas del modelo de reconocimiento de emociones	4 días	30/06/2020	03/07/2020								
Pruebas del modelo de reconocimiento de comportamientos	3 días	04/07/2020	06/07/2020								
Pruebas de la integración de los modelos	10 días	07/07/2020	17/07/2020								
Redacción de la memoria del proyecto	76 días	01/06/2020	27/08/2020								

Fig. 9.3. Diagrama de Gantt del proyecto por días 3

10. ANEXO B

Abstract

Driving is one of the most common causes of death present in today's society. With the aim of decreasing the number of fatalities that take place on the road, this project presents an Advanced Assistance Driving System (ADAS) based on the driver's emotions and behaviour. This system is made of two Deep Learning models, one specialized in the emotions and the second one in the driver's actions. In this work convolutional neural network (CNN) techniques are applied in a variety of emotion recognition datasets including FER2013, CK+, JAFFE and KDEF. The Deep learning model is trained using images from all the mentioned datasets, and advanced preprocessing techniques such as facial detection are applied to the images. The best test accuracy achieved in this model is of 85.5 %. Similarly, a CNN model is trained using the 3MDMA distracted driver's database performing a 97.7 % of accuracy in test. Both models are combined to provide a system that can provide relevant information from a driver's image.

Introduction

For the time being, driving is the most widely used way to get around, especially when commuting between close distances. The automobile industry is one of the largest sectors of this century, according to the most recent statistics, the automobile sales are estimated to be around 59.5 million units this 2020 [1].

What used to be an exclusive and expensive conveyance, it is now spread all over the world, and available for 19 % of the population. Figure 10.1 shows an interesting approach to how this situation is predicted to evolve within the next decades. Although there are already over one billion cars circulating worldwide, this number is expected to double before 2050. As long as the number of vehicles continues to grow, so will accidents do. This perspective of the future, make the study and prevention of road traffic crashes (RTCs) essential.

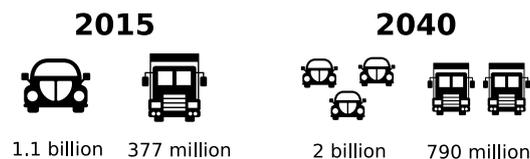


Fig. 10.1. Global transport growth over the next 25 years [2]

According to the World Health Organization (WHO) statistics from 2020, traffic accidents are the death cause of approximately 1.35 million people each year [4]. An study

performed by WHO [3] compared road injuries to the most common causes of death, being road injuries the eleventh most common cause of death in 2017.

Nonetheless, the number of deaths are just the tip of the iceberg. Road traffic crashes cause non-fatal injuries, that may lead to a disability, to between 29 and 50 million of people [4].

The consequences of traffic accidents are not only about the lives lost in the road but also the costs that its damages cause to the families involved, as well as the impact they have on the economy. The WHO states that traffic crashes cost most countries 3 % of their gross domestic product [4].

This thesis is focused on the factors that occur at the pre-crash phase, and are only related to human factors. This particular stage, includes different aspects related to the driver, such as its attitudes and aptitudes, as well as his social context.

More about the factors that affect RTCs can be found in [6], where Petridou et al. developed an extended list of these factors divided into two main categories. Among the factors discovered, this project focuses specifically on the factors related to psychological stress and temporary distractions. Irritability and anger can become common traits in people who are stressed [53], along with those, a more extended set of human emotions will be investigated, since they might have potential influence in driver's behavior.

Based on other authors' research that will be discussed later, it cannot be denied that there is a connection between the current state of a driver and whether it leaves or not to an accident. Willing to reduce the increasing number of deaths caused by traffic accidents.

We aim to apply the most pioneering computer science techniques, in order to propose a solution, based on the existing relationship between human behavior and RTCs, that will aim to reduce the number of car crashes and to provide safer journeys to the drivers.

Objectives

This project aims to develop an Advanced Driver Assistance System (ADAS) based on Artificial Intelligence (AI) that will detect dangerous situations in an automobile based on the driver's actions and emotions.

When dealing with hazardous conditions on the road, it is crucial to alert the driver. The sooner a system can perceive and inform the passengers, the earlier they can react to the present dangers. This project will be focused on a system that can detect the driver's emotions and behaviour in real-time and make an informed decision concerning the car's safety.

State of the Art

Influence of emotions in driving

Drivers have control over their actions and decisions while driving. A health driver focused on the car, the traffic, and the environment can foresee and react to an unexpected danger. A problem arises when the driver is concerned about an external event that has affected his mental state. This typical behaviour makes the task of driving more difficult and redirects the driver's attention that should focus on having a safe journey.

Many experts in the field have studied the effects of different emotions while driving. It has been demonstrated that negative feelings such as Anger, Sadness, or Anxiety can have a negative impact on the overall driver's performance [14]. On the other hand, the effects of positive emotions have been less studied. However, some experiments described in [14] reflect that those positive feelings can also increase the number of errors made by humans.

We expect further research in this field of interest, to better understand the consequences of the driver's behaviour. Of all different emotions, "anger" is the most repeated feeling among drivers. According to [14], "anger" is prevalent on roads because of the large number of influencing factors such as traffic, stress, horn-honking, etc.

In his work, Pêcher et al. [14] stated the leading cause of anger in the traffic situation itself. This is an interesting statement that might lead to questioning how we can avoid the consequences of angry drivers if we cannot change what triggers the driver's frustration. Providing the car with relevant information regarding the driver's emotional state could be a fascinating solution to this problem.

By simulating a driving environment, and presenting the subjects with a stimulus to observe their behaviours, Steinhauser et al. [15] found that the emotions altered the subjects directly (e.g., promoting aggressive driving) and indirectly (e.g., by attenuating dual-task costs). Their results provide evidence that emotions affect driving, and this effect depends both on the task the driver is performing and the emotion introduced.

Distracted behaviours and driving

Besides emotions, there are other distracting behaviours we would like to discuss that are even more lethal. These behaviours involve communicating with others using technological devices, singing, using the GPS, etc.

More on these common behaviours will be discussed in this section. However, before going on with that subject, it is essential to highlight the fact that these behaviours are usually triggered by strong emotions. For instance, a passenger experiencing a sad moment while driving a car could be tempted to use his phone to find psychological help.

There are countless examples we could think of that show the relationship between

distracted behaviours and emotions. Moreover, by analyzing the relationship between behaviour and emotions, we can understand why positive emotions are also crucial to better understand the driver's conduct. An example that illustrates this is how we behave when we are in a good mood. A driver that is feeling happy will be more likely to perform dangerous moves, without being aware of it, such as turning to talk to the passengers or turning up the radio volume. Hence, the driver will be losing both vision and concentration on the road.

An study performed by the Virginia Tech Transportation Institute (VTTI) revealed that almost 80 percent of crashes and 65 percent of near-crashes involved some form of driver inattention within three seconds before the event [17]. This article, also provides interesting information regarding how each of the typical distracting behaviours can affect the passenger's safety. The article states that a driver texting is 23.2 times more likely to have a car crash, and 2.8 times more likely if the driver is dialing a cell phone. In addition, the study performed in 2016 by the university of Iowa states that 60 % of teen chases involve distractions.

Emotion recognition via Deep Learning

Many authors in the field have applied the use of Deep Learning in order to provide models capable of recognizing human emotions from images. This section will cover some of the most inspiring works in emotion recognition until today.

In 2017 Shan et al. [32] published an interesting work in emotion recognition. In their project , the datasets CK+[33] and JAFFE [34] were used independently to train a convolutional neural network. Face detection along with histogram equalization techniques were used to preprocessing the images before feeding them into the neural network. This paper achieved an accuracy of 76.7 % (JAFFE) and 80.3 %(CK+).

Another interesting project in Emotion Recognition was performed in [35]. This paper includes a comparison of some of the most used convolutional neural networks architectures. The dataset FER2013 [36] was used as input for these convolutional neural networks models. The best result was achieved using the architecture AlexNet [37] with a 64.2 % of accuracy in test.

Ramdhani et al. [38] presented a model trained using the FER2013[36] dataset. In this paper, they not only use the well known FER2013 dataset but also compare it with their own made dataset. The study focuses in analyzing how changes in the batch size values affect the accuracy results for different CNN architectures. The best results were achieved by using their own dataset for training, with a batch size with value 8. The accuracy obtained was 73.8 %.

One of the most recent related works was published in 2019 by Pranav et al. [39]. FER2013 was used to compare different machine learning models including decision trees, neural networks and convolutional neural networks. The model that provided better

results was the proposed convolutional neural network. The architecture of this network was made of two layers of 32 and 64 filters. The accuracy obtained was 55.6 %.

In 2017, Nwosu et al. [40] published a paper that achieved impressive results in terms of accuracy. Their model was a convolutional neural network made of two different channels. The first channel included a cropped image of the eyes while the second channel was made of an image of the mouth's expression. The accuracy was a 97.7 % in JAFFE and a 95.7 % in CK+.

The main difference between the above mentioned works and this project relies on the way the models are being trained. For this work, the objective is to train a convolutional neural network using four different datasets. This will provide a large amount of examples that will make the model less dependant of the environment and the context of the picture.

Distracted driver's detection using Deep Learning

In this section, the most recent works in distracted drivers detection using deep learning will be commented. In contrast with emotion recognition, less works have been published in distracted drivers.

In 2017, Torres et al. [41] developed a convolutional neural network model that used tree layers including 32, 32 and 64 filters. The accuracy achieved in this project was a 99 % using a reduced version of the dataset State Farm [42]. The reduction of the dataset was performed by removing 4 out of the 9 initial classes represented in the dataset.

Another interesting work was performed by Baheti et al. [43]. They present a modification in the convolutional network architecture VGG-16 [44]. This modification reduces to 15 millions the total number of parameters in the CNN. This is a significant reduction, the original architecture was made of 140 millions of parameters. This improvements allows to have considerably reduced times in training and predicting using the model. The accuracy obtained was a 95.5 % using the State Farm dataset of distracted drivers.

To conclude, Sathe et al. [45] proposed in 2018 a convolutional neural network model using the data augmentation technique. By using this technique they could increase the size and variety of the State Farm dataset [42]. The accuracy obtained in this project is 96.7 %.

ADAS based on the driver

For this project, both models will be combined to create an ADAS that will aim to improve safety and the overall driving experience. This section will provide an overview of other ADAS that use Artificial Intelligence in order to provide information regarding the driver's behaviour and emotions.

The system that relates the most to the one that is being proposed in this project is the one developed by Affectiva[46]. Affectiva created a system capable of knowing the

driver's distraction level. This software is also capable of detecting the level of drowsiness in the driver as well as other states such as joy.

This system by Affectiva is similar to the one that is proposed in this project. Nevertheless, the machine learning models developed for this work will not only detect the level of drowsiness but also the current action of the driver along with the driver's emotion.

System's architecture

The design of the system that it is proposed in this project combines two Convolutional Neural Networks models. One model will be specialized in detecting emotions while the other will be in the actions of the driver. Those CNN models will be included in a bigger system that will process an image from a camera and produce an output including the relevant information of the driver. This information will be made of the emotional state of the driver, and the distracting activity that the driver is performing.

In this section, the two convolutional networks models that make up the system will be presented and analyzed. It is important to mention that both models have been trained and developed to be part of one main system, hence some of the design decisions have been taken into account.

Emotion recognition

This section presents the different stages that involve the creation of the Emotion Recognition model. First, the datasets used to train the model will be presented, and then the experimental stage, including the best model's accuracy and its architecture.

Datasets

At the time this paper has been written, the range of labelled images available for research purposes in emotional artificial intelligence is wide and extensive. In pursuance of the creation of a data base that could perform well in every situation presented, different data sets have been used for the model's training and test.

This section will provide an overview of the datasets that have been used for training the convolutional neural networks.

The extended Cohn-Kanade database (CK+) [33] contains 327 images from 118 different subjects. Participants were 18 to 50 years of age, 69 % female, 81 %, Euro-American, 13 % Afro-American, and 6 % other groups.

The Karolinska Directed Emotional Faces (KDEF)[48] is a set of totally 4900 pictures of human facial expressions. The set of pictures contains 70 individuals displaying 7 different emotional expressions, 70 amateur actors, 35 females and 35 males, between 20

and 30 years of age.

The FER-2013 dataset [36] consists of 28,000 labelled images in the training set, 3,500 labelled images in the development set, and 3,500 images in the test set. The FER-2013 dataset was created by gathering the results of a Google image search of each emotion, no much more is known about the population distribution.

The Japanese Female Facial Expression (JAFFE) Database. The database contains 213 images posed by 10 Japanese female models. [34]

These data sets were carefully selected, in order to create a big uniform dataset. This is meant to be diverse in gender and race including people of all ages from all over the world, therefore the CNN must be trained using images from all the datasets. In order to achieve this, the images have been resized to the same dimensions (64x64). All images contains one of the six basic emotions proposed by Paul Ekman [8], plus the neutral emotion, being this seven emotions the classes that will be predicted by the proposed CNN model.

Preprocessing

Different strategies have been applied in order to decide the image's preprocessing that achieves the best results in terms of accuracy. The first approach was using the resized images as input for feeding the CNN without any other additional technique. Later on, the same images but with a face selection crop were used as input. This provided higher accuracy, without increasing the CNN's architecture complexity, due to the meaningless information that was deleted when cropping the face using a square shape. This lead to the idea of creating an elaborated preprocessing technique (see Figure 10.2) that ended up providing the best accuracy results using simpler CNNs.

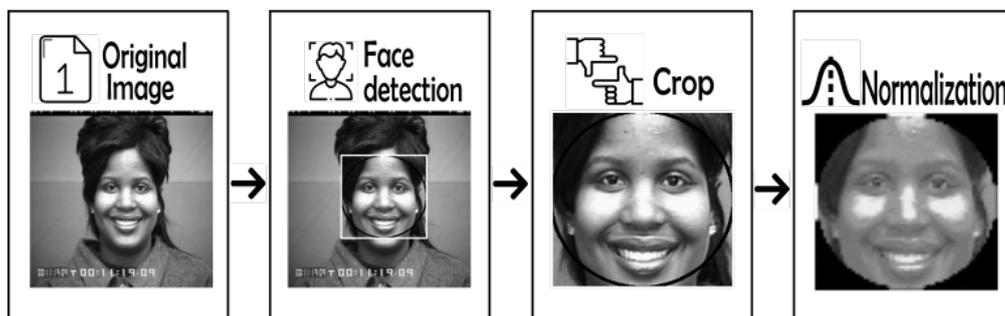


Fig. 10.2. Preprocessing steps

Experiments and final results

The experimental stage of this project involved training different convolutional neural networks using a variety of combinations between the presented datasets. A total of 45

experiments have been performed by changing the batch size and the CNN architecture's definition.

Table 10.1 summarizes the experiments realized to create the emotional artificial intelligence model. From the total 75 experiments, just 45 of them were valid. A not valid experiment is defined in this project as one that, due to the incorrect random initialization of the weights, it doesn't lead to any relevant conclusions regarding the model.

Number of experiments		Time		Accuracy	
Total	Valid	Total	Valid	Mean	Best value
79	45	60 hours y 38 minutes	36 hours y 10 minutes	72.45	85.5

TABLA 10.1. SUMMARY OF THE EXPERIMENTS

Figure 10.3 presents the architecture of the model that achieves the best accuracy. The model was trained using images from all the datasets that were presented in this project. Those images were preprocessed using the mentioned preprocessing algorithm explained in Figure 10.2. They served as an input for the convolutional neural network that is defined in Figure 10.3. This neural network's architecture is made of 6 convolutional layers followed by a dense layer containing 1500 neurons in its hidden layer. The image that serves as input in Figure 10.3 is from the 3MDAD Dataset of distracted drivers. It gives an example of how this model will be used in drivers. The network's output provides information based on the driver's emotional state. In addition, Table 10.2 presents the remaining hyperparameters used to train the convolutional neural network.

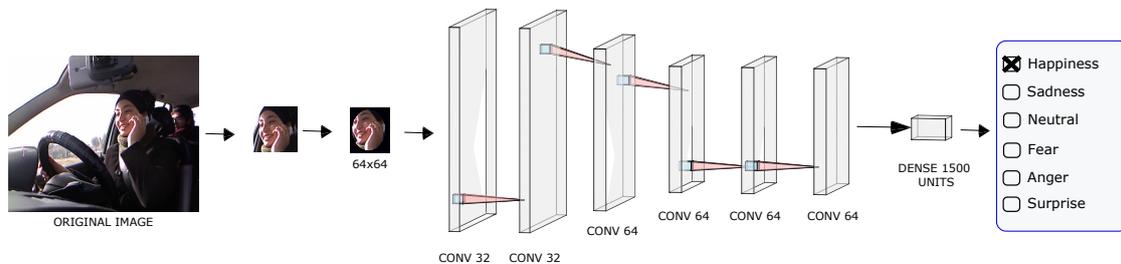


Fig. 10.3. 3D representation of CNN architecture

Hyperparameter	Value
Activation function	Relu
Optimizer	Adam
Learning rate	0.001
Batch size	128

TABLA 10.2. BEST MODEL HYPERPARAMETERS

Driver's behaviour recognition

This section covers the distracted drivers final model and its development. A review of the images used, the training parameters and the final results will be presented throughout this section.

Dataset

Among the different sources of driver's images, the Multimodal Multiview and Multispectral Driver Action Dataset (3MDAD)[49] has been selected for this project. This dataset created by Jegham et al. has images of over 50 different subjects performing 16 different actions (see Table 10.3) while driving. One of the key considerations that differentiates this dataset among others is the different views the pictures are taken from. There is one view in this dataset where all photos are taken from a frontal side of the car where you can see the driver's face and the action that it is taking place. Therefore, this dataset has been used not only for training the model but also for testing the entire ADAS system.

Code	Description	Code	Description
AC1	Safe Driving	AC9	Having picture
AC2	Doing hair and makeup	AC10	Talking to passenger
AC3	Adjusting radio	AC11	Singing or dancing
AC4	GPS operating	AC12	Fatigue and somnolence
AC5	Writing message using right hand	AC13	Drinking using right hand
AC6	Writing message using left hand	AC14	Drinking using left hand
AC7	Talking phone using right hand	AC15	Reaching behind
AC8	Talking phone using left hand	AC16	Smoking

TABLE 10.3. DISTRACTED DRIVERS PROBLEM'S CLASSES

Preprocessing

The image preprocessing that was initially applied to this model consisted in a simpler resizing of the images from 480x640 pixels to 120x160 pixels. Eventually, some of the images had to be deleted. The dataset is made of the frames involved in each of the actions, at the very beginning there is nothing relevant in the images that implies which action is being performed, but it is still labelled as if the image belongs to the action. As a consequence, a sampling of the dataset were performed to reduce it to the frames where you can clearly see which action is being performed. This sampling was done first by automatically deleting the first and last ten images of a sequence, and then by manually deleting the extra frames that may still be miss-labelled.

Experiments and final results

The relevant information about the experiments have been summarized in Table 10.4. The total number of experiments performed before achieving the best result was 53, being only 22 valid experiments. The best value, in terms of accuracy, was a 97.7 % in test. In this scenario, there is a notable difference between the mean value of accuracy and the best value. During the experimental stage, the number of classes to predict was reduced in order to improve the accuracy levels in test. As a consequence, there is a huge step between the low initial levels of accuracy and the ones obtained after removing some of the classes, and therefore, simplifying the problem.

Number of experiments		Time		Accuracy	
Total	Valid	Total	Valid	Mean	Best value
53	22	29 hours y 28 minutes	15 hours y 12 minutes	45.4	97.7

TABLA 10.4. SUMMARY OF THE EXPERIMENTS

The final model combined the hyperparameters from Table 10.2, and the convolutional neural network architecture shown in Figure 10.4. In the figure it can be appreciated the final classes that have been determined as the most relevant, the remaining classes have been removed from the problem in order to make it less complex.

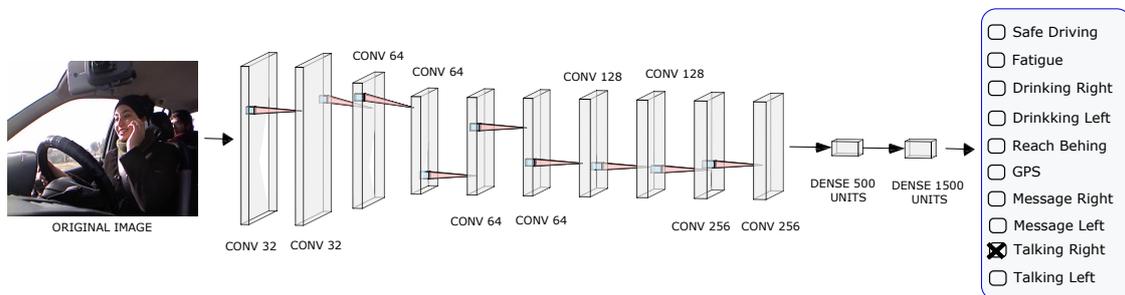


Fig. 10.4. 3D representation of CNN architecture

The Advanced Driving Assistance System

The ADAS that is being proposed in this project has been developed by combining both CNN models into one system. This algorithm written in Python code first establishes a connection with a an external camera, preprocess the image, and applied the models to the image. Later on, this information is presented in the original image that was first introduced from the camera. Figure 10.5 shows an example of the output that the models provide from an image.

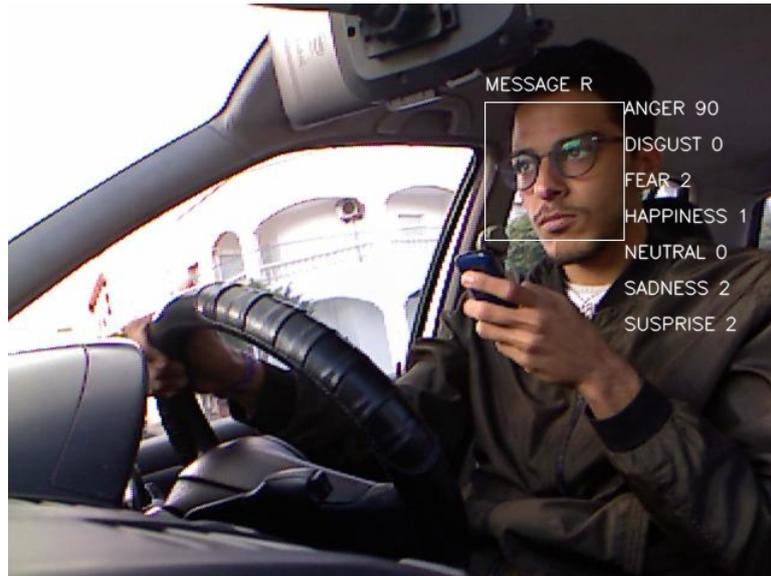


Fig. 10.5. ADAS preview

Conclusions

The principal goal of this project was creating an ADAS that could perform in real-time. This expectation has been achieved due to the numerous experiments performed. Due to the time investment in finding the simpler architecture that provides the best accuracy values a system that not only gives remarkable results but also performs in real time have been developed.

An interesting future work for this project could involve the integration of the developed ADAS software in a vehicle, and therefore, testing its performance in a real-life environment.