INSTITUTO TECNOLÓGICO DE CHIHUAHUA

DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

RECONOCIMIENTO DE EMOCIONES Y ESTADOS AFECTIVOS EN TIEMPO REAL MEDIANTE EXPRESIONES FACIALES CON UN MODELO DE APRENDIZAJE DE MÁQUINA EMBEBIDO”

TESIS

PARA OBTENER EL GRADO DE

MAESTRO EN CIENCIAS EN INGENIERÍA ELECTRÓNICA

PRESENTA:

JESÚS JOSHUA MUÑOZ PACHECO

DIRECTOR DE LA TESIS:

DR. JUAN ALBERTO RAMÍREZ QUINTANA

CHIHUAHUA, CHIH., OCTUBRE 2025.

SEP
SECRETARÍA DE EDUCACIÓN PÚBLICA

TECNOLÓGICO NACIONAL DE MÉXICO

Instituto Tecnológico de Chihuahua
División de Estudios de Posgrado e Investigación

Chihuahua, Chih. 30 de septiembre de 2025

C. JESÚS JOSHUA MUÑOZ PACHECO
PRESENTE

En cumplimiento con los requerimientos para la obtención del grado de Maestro en Ciencias en Ingeniería Electrónica y a propuesta de su Comité Tutorial, la División de Estudios de Posgrado e Investigación le concede la autorización para imprimir la tesis titulada ***"Reconocimiento de emociones y estados afectivos en tiempo real mediante expresiones faciales con un modelo de aprendizaje de máquina embebido"*** dirigida por el Dr. Juan Alberto Ramírez Quintana con el siguiente contenido de capítulos:

I. Introducción
II. Marco Teórico
III. Antecedentes
IV. Materiales Y Métodos
V. Little Expression Net - Liexnet
VI. Implementación en tiempo real
VII. Resultados
VIII. Conclusiones

A T E N T A M E N T E
Excelencia en Educación Tecnológica®
"La Técnica por el Engrandecimiento de México"

EDUCACIÓN SECRETARÍA DE EDUCACIÓN PÚBLICA | TECNOLÓGICO NACIONAL DE MÉXICO
INSTITUTO TECNOLÓGICO DE CHIHUAHUA
DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

LEONCIO OCHOA CERVANTES
JEFE DE LA DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

LOC/vgo

2025 Año de La Mujer Indígena

Instituto Tecnológico de Chihuahua
División de Estudios de Posgrado e Investigación

Chihuahua, Chih. 17 de septiembre de 2025

LEONCIO OCHOA CERVANTES
JEFE DE LA DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN
PRESENTE

En cumplimiento con los requerimientos para la obtención del grado de Maestro en Ciencias en Ingeniería Electrónica, le notificamos que el documento de tesis del alumno **C. JESÚS JOSHUA MUÑOZ PACHECO,** titulado ***"Reconocimiento de emociones y estados afectivos en tiempo real mediante expresiones faciales con un modelo de aprendizaje de máquina embebido"*** dirigido por el Dr. Juan Alberto Ramírez Quintana, ha sido aprobado y aceptado para su impresión.

Por lo anterior, proponemos le sea concedida la autorización de impresión correspondiente.

Agradeciendo la atención a la presente, quedamos de usted:

A T E N T A M E N T E
Excelencia en Educación Tecnológica®
La técnica por el Engrandecimiento de México

Juan Ramirez
DR. JUAN ALBERTO RAMÍREZ QUINTANA
MIEMBRO DEL COMITÉ TUTORIAL

DR. JESÚS ALFONSO MEDRANO HERMOSILLO
MIEMBRO DEL COMITÉ TUTORIAL

MTRA. ALMA DELIA CORRAL SÁENZ
MIEMBRO DEL COMITÉ TUTORIAL

DR. LUIS FRANCISCO CORRAL MARTÍNEZ
MIEMBRO DEL COMITÉ TUTORIAL

EDUCACIÓN SECRETARÍA DE EDUCACIÓN PÚBLICA | TECNOLÓGICO NACIONAL DE MÉXICO
INSTITUTO TECNOLÓGICO DE CHIHUAHUA
DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

2025 Año de La Mujer Indígena

Instituto Tecnológico de Chihuahua
División de Estudios de Posgrado e Investigación

CARTA CESIÓN DE DERECHOS

En la ciudad de Chihuahua el día 29 de septiembre de 2025, el que suscribe, C. JESÚS JOSHUA MUÑOZ PACHECO con número de control G23061083, de la Maestría en Ciencias en Ingeniería Electrónica, adscrita a la División de Estudios de Posgrado e Investigación del Instituto Tecnológico de Chihuahua, manifiesta que es autor intelectual del presente trabajo de Tesis bajo la dirección del el Dr. Juan Alberto Ramírez Quintana y cede los derechos del trabajo titulado "Reconocimiento de emociones y estados afectivos en tiempo real mediante expresiones faciales con un modelo de aprendizaje de máquina embebido", al Tecnológico Nacional de México y/o Instituto Tecnológico de Chihuahua para su difusión, divulgación, transmisión, reproducción, así como su digitalización con fines académicos y de investigación.

C. JESÚS JOSHUA MUÑOZ PACHECO

Instituto Tecnológico de Chihuahua
División de Estudios de Posgrado e Investigación

DECLARACIÓN DE ORIGINALIDAD

En la ciudad de Chihuahua el día 29 de septiembre de 2025, el que suscribe, C. JESÚS JOSHUA MUÑOZ PACHECO de la Maestría en Ciencias en Ingeniería Electrónica, con número de control G23061083, adscrito a la División de Estudios de Posgrado e Investigación del Instituto Tecnológico de Chihuahua, manifiesta que es autor intelectual de la tesis titulada "Reconocimiento de emociones y estados afectivos en tiempo real mediante expresiones faciales con un modelo de aprendizaje de máquina embebido" bajo la dirección el Dr. Juan Alberto Ramírez Quintana; que el contenido es original y que las fuentes de información consultadas para su fundamentación están debidamente citadas y referenciadas.

C. JESÚS JOSHUA MUÑOZ PACHECO

Av. Tecnológico #2909 Col. 10 de Mayo
C.P. 31200, Chihuahua, Chihuahua. Tel. (614) 2012000 ext. 2157
e-mail: dir_chihuahua@tecnm.mx tecnm.mx http://itchihuahua.mx

Chihuahua, Chihuahua, octubre de 2025

Dra. Rosaura Ruiz Gutiérrez
Director/a de SECIHTI

Sirva la presente para saludarla e informarle que a la fecha he obtenido el grado **de Maestro en Ciencias en Ingeniería Electrónica** en la División de Estudios de Posgrado e Investigación del Instituto Tecnológico de Chihuahua, motivo por el cual agradezco todo el apoyo brindado por este Consejo que usted representa, con el otorgamiento de una beca que permitió dedicarme de manera exclusiva a la realización de mis estudios de posgrado y de esta manera lograr el objetivo principal del convenio establecido.

Sin otro particular quedo a sus órdenes no sin antes reiterar mi agradecimiento.
Atentamente

Jesus Joshua Muñoz Pacheco

AGRADECIMIENTOS

Sinceramente hay muchas personas a las que quisiera agradecer, primeramente, a mi familia por apoyarme en mis sueños siempre, por estar siempre presentes y por todo el amor que siempre me han dado, los amo.

A mis amigos, quienes no se hartaron de mis constantes quejas cuando tenía que presentar mis avances de tesis, y que siempre me acompañaron en las noches largas entre sesiones de juegos y reuniones que fueron tan alegres y significativas para mí, los amo chicos.

Quiero agradecer especialmente a mi asesor de tesis Dr. Juan Alberto Ramírez Quintana, quien es para mí un modelo a seguir debido a su trayectoria en la investigación y el desarrollo de áreas de innovación, muchas gracias por apoyarme cuando más lo necesitaba y por darme a conocer una oportunidad que abrió el camino que hoy creo que es en el que siempre he querido avanzar y también a los proyectos TecNM 19182.24-P y CHIH-PYR-2025-21864 por aportar con mi trabajo de investigación.

A lo largo de estos años he conocido personas maravillosas, algunas de ellas jamás las volveré a ver y otras ni si quiera las he conocido en persona, pero su interacción conmigo, la manera en la que sus palabras de aliento, gritos, ánimos, llanto, buenos y malos deseos me cambiaron fue muy significativa, hoy puedo decir que gracias a todas esas personas me convierto cada vez más en lo que siempre quise ser, en una persona que poco a poco se hace tan fuerte no solo para defenderse a sí mismo, si no que tiene la suficiente fuerza para ayudar a otros, poco a poco me convierto en la persona que siempre quise que me defendiera en los peores momentos que he tenido, y hoy es completamente extraño para mi pensar que todas interacciones, esas decepciones, esas alegrías, esos llantos, ese sudor y ese esfuerzo se vean reflejadas en una persona que siempre buscará mejorar.

Nunca alcanzaré la perfección, pero no busco ser perfecto, busco ser cada día una mejor versión de mí, porque yo y todos los seres del mundo merecen siempre mi mejor versión.

Y hay un último agradecimiento que quisiera hacer, aunque suene raro y confuso, mi expresión por medio de la palabra escrita siempre fue mi forma de decir todo lo que siento y debo decirme lo muy orgulloso que me siento de ser la persona que soy, en lo mucho que he mejorado, en lo mucho que he aprendido a amar a otros y a amarme a mí a pesar de la constante critica externa e interna, como dije no soy perfecto y sé que no le agradaré ni gustaré a todo el mundo pero eso no es importante ahora porque no necesito hacerlo, me amo y para seguir haciéndolo prometo seguir respetándome y cuidándome tanto física como mentalmente, prometo no rendirme en la búsqueda de conexiones emocionales con las personas tan asombrosas que hay allá afuera, y prometo seguir mis sueños… después de todo esto es solo el principio.

Por último, quisiera nombrar aquí a las personas que fueron más significativas para mí personalmente:

Abigail Muñoz, Erika Pacheco, Jesús Muñoz R., Irvin Rivera, Daniel González, Bryan Meza, Miguel Portillo, Isaac Corral y Daniela García.

RESUMEN

RECONOCIMIENTO DE EMOCIONES Y ESTADOS AFECTIVOS EN TIEMPO REAL MEDIANTE EXPRESIONES FACIALES CON UN MODELO DE APRENDIZAJE DE MÁQUINA EMBEBIDO

Jesus Joshua Muñoz Pacheco
Maestría en Ciencias en Ingeniería Electrónica
División de Estudios de Posgrado e Investigación del
Instituto Tecnológico de Chihuahua
Chihuahua, Chih. 2025
Director de Tesis: Dr. Juan Alberto Ramírez Quintana

La computación afectiva es una rama de la inteligencia artificial que estudia cómo las máquinas pueden reconocer, interpretar y simular emociones humanas. El estudio de esta disciplina es relevante porque los estados afectivos negativos pueden influir en la calidad y el rendimiento de una persona, llegando incluso a provocar desde bajo desempeño laboral hasta accidentes graves, especialmente cuando se opera maquinaria bajo dichas condiciones. Actualmente, existen modelos capaces de reconocer emociones básicas, pero presentan una alta complejidad computacional, lo que dificulta su uso en tiempo real en dispositivos móviles o en entornos que están fuera del acceso a sistemas de cómputo de alto rendimiento como servidores especializados.

Esta tesis presenta un método de reconocimiento de emociones y estados afectivos con imágenes de expresiones faciales y un modelo ligero de aprendizaje profundo para procesamiento en tiempo real en sistemas embebidos. El modelo se llamó *Little Expression Net* (LiExNet) y se trata de una red neuronal convolucional de bajo costo computacional que integra mecanismos de atención para mejorar la clasificación de emociones faciales. LiExNet se entrena y prueba con las bases de datos de CK+, JAFFE, KDEF y FER2013. Los resultados de la exactitud del modelo con las diferentes bases de datos fueron de aproximadamente 98.2 %, 71.8 %, 85.3 % y 78.2 % respectivamente. Posteriormente, el modelo se adapta mediante

fine-tunning al reconocimiento de estados afectivos, empleando la base de datos EMOTION-ITCH etiquetadas definidas por la prueba SAM (Self-Assessment Manikin).

Se utilizó la tarjeta de evaluación NVIDIA Jetson TX2 para las pruebas de inferencia en tiempo real en sistemas embebidos, ya que es un sistema que tiene a su disposición software para la facilitar la implementación de redes neuronales bajo APIs como *Keras/Tensorflow*. Los resultados obtenidos fueron una inferencia de aproximadamente 269.44 FPS, y un uso de la memoria RAM del 88 % en estado de rendimiento máximo de la tarjeta Jetson TX2.

ÍNDICE

ÍNDICE DE FIGURAS.

ÍNDICE DE TABLAS

GLOSARIO

Tiempo real: se refiere a un sistema que puede procesar información y responder a eventos casi instantáneamente después de su ocurrencia, con un retardo mínimo o nulo. La información disponible en tiempo real se genera, se procesa y se presenta de forma casi inmediata, lo que permite tomar decisiones rápidas y eficientes en situaciones donde la velocidad es crítica.

INTRODUCCIÓN

Los estados afectivos negativos pueden ocasionar una serie de inconvenientes en la vida cotidiana, como es el caso del estrés, la ansiedad y la depresión. Estas condiciones, traen dificultades que van desde bajo rendimiento en el ámbito profesional hasta accidentes graves debido al mal manejo de maquinaria pesada.

Existen grandes avances en el análisis de estados afectivos negativos, principalmente en reconocimiento de emociones mediante imágenes de expresiones faciales basados en aprendizaje profundo o *deep learning.* Los modelos más comunes en el estado del arte de esta forma de reconocimiento incluyen arquitecturas jerárquicas profundas [1], redes con mecanismos de atención de alta capacidad [2, 3] o Redes Neuronales Convolucionales (CNN por sus siglas en inglés) profundas como ResNet-152 y DenseNet-161 [4]. Estos modelos han generado buenos resultados, pero llegan a tener de 20 a 40 millones de parámetros y requieren millones de operaciones flotantes por imagen [5]. Dicho costo impide su despliegue en sistemas embebidos que deben ejecutar inferencia local y continua, condición indispensable en entornos donde los niveles de estrés y depresión deben vigilarse *in situ* y en tiempo real para la prevención de accidentes [6].

De acuerdo con el análisis del estado del arte realizado, las soluciones de bajo costo computacional basadas en Inteligencia Artificial (IA) para reconocimiento de emociones suelen ser escasas. Debido a este problema, es necesario desarrollar modelos ligeros que puedan implementarse en entornos donde no se tiene acceso a sistemas computacionales de alto rendimiento o conexiones a servidores especializados, como aplicaciones en sistemas móviles o remotos y que se procesen en tiempo real. Por lo tanto, en esta tesis se propone un modelo de reconocimiento de emociones y detección de estados afectivos negativos en tiempo real. El modelo se denominó *Litte Expression Net* (LiExNet), el cual contiene cuatro bloques de extracción de características mediante capas convolucionales, capas convolucionales separables en profundidad y canales eficientes de atención (ECA por sus siglas en inglés) [7]. La clasificación se realiza con una capa densa totalmente conectada seguida de una función de activación *softmax.*

LiExNet se propuso bajo la siguiente metodología. Primero, se realizó un análisis del estado del arte sobre modelos de IA en sistemas embebidos, modelos de reconocimiento de emociones mediante expresiones faciales y procesamiento de imágenes en tiempo real. Luego, se recopilaron y utilizaron las bases de datos y métodos más significativos en la literatura sobre expresiones faciales para su clasificación en emociones discretas. Con base en diversos experimentos, se desarrolló LiExNet, la cual preserva la precisión de modelos pesados en tareas de reconocimiento de emociones, pero con una reducción de 10 a 15 veces el consumo computacional. Este consumo habilita el despliegue de LiExNet en tecnologías de sistemas embebidos como NVIDIA Jetson TX2, pero sin sacrificar fotogramas por segundo ni autonomía energética. El entrenamiento de LiExNet se realizó en una computadora de propósito general con tarjeta gráfica de procesamiento, mientras que la validación se realizó con la misma computadora y una tarjeta de desarrollo NVIDIA Jetson TX2.

El resto de la tesis se divide de la siguiente manera: el capítulo uno comprende el marco teórico, el cual contiene conceptos importantes para el entendimiento de esta tesis. El capítulo dos presenta los antecedentes que respaldan la investigación realizada. El capítulo tres se centra en describir las bases de datos, métodos y experimentos que se requirieron para proponer LiExNet. El capítulo cuatro presenta LiExNet y el capítulo cinco describe la implementación en tempo real de LiExNet en un sistema embebido. El capítulo seis muestra los resultados de este trabajo de investigación y el capítulo siete resume las conclusiones derivadas de los hallazgos.

1. MARCO TEÓRICO

El presente trabajo busca ofrecer un análisis integral sobre el desarrollo de una red neuronal profunda para la identificación de emociones y estados afectivos que funcione en tiempo real. Para ello, es importante establecer los conceptos, teorías y enfoques que sustentan esta investigación.

Este capítulo se divide de la siguiente manera: la sección 1.1 presenta conceptos básicos de IA. La sección 1.2 describe las arquitecturas de redes neuronales. La sección 1.3 describe lo referente a emociones y la sección 1.4 engloba conceptos de estrés. Finalmente, la sección 1.5 muestra los conceptos referentes a tiempo real.

1.1 Inteligencia artificial

La inteligencia artificial (IA) es un concepto polisémico, ya que no existe definición única aceptada por los autores que estudian y desarrollan elementos relacionados con este tema. Para el tema que nos ocupa, tomamos a definición de La Real Academia Española, que describe a la IA como "disciplina científica que se ocupa de crear programas informáticos capaces de ejecutar operaciones comparables a las que realiza la mente humana, como el aprendizaje o el razonamiento lógico" [8]. Esta definición no abarca toda la amplitud del campo debido a que la IA engloba varias subdisciplinas, entre las que destacan:

- **Aprendizaje automático (*Machine Learning*, ML):** estudio de algoritmos que aprenden a partir de datos [9]. Sus tres paradigmas principales son:

  - **Aprendizaje supervisado:** el modelo se entrena con pares entrada-salida conocidos [9].
  - **Aprendizaje no supervisado:** el sistema descubre patrones a partir de las entradas [9].
  - **Aprendizaje por refuerzo:** un agente optimiza su comportamiento recibiendo recompensas o castigos del entorno [10].

- **Aprendizaje profundo (*Deep Learning*, DL):** rama del ML basada en redes neuronales profundas que aprenden representaciones jerárquicas de la información [11].

- **Visión por computadora:** área que emplea algoritmos para extraer, analizar y procesar información de imágenes o video, de modo que el sistema pueda tomar decisiones o ejecutar acciones [12].

Estas últimas subdisciplinas (aprendizaje profundo y visión por computadora) son esenciales para el presente trabajo, pues proporcionan los modelos capaces de interpretar las expresiones faciales y clasificarlas en emociones humanas, lo que permite detectar indicadores de estrés en tiempo real.

1.2 Redes neuronales profundas

Las redes neuronales profundas son modelos computacionales inspirados en la estructura y funcionamiento de sistemas neuronales biológicos, compuestos por múltiples capas de procesamiento jerárquico que permiten aprender representaciones de datos a diferentes niveles de abstracción [11].

Entre los principales tipos de redes neuronales profundas se encuentran:

Redes Neuronales Convolucionales (CNN, por sus siglas en inglés): Son arquitecturas especializadas en el procesamiento de datos con una estructura en forma de cuadrícula, como las imágenes. Utilizan capas convolucionales que aplican filtros para extraer características espaciales jerárquicas [11].

Redes Neuronales Recurrentes (RNN): estas redes están diseñadas para procesar datos secuenciales, manteniendo información en el tiempo mediante conexiones recurrentes. Son ampliamente utilizadas en procesamiento de lenguaje natural, análisis de series temporales y reconocimiento de voz [13].

Transformers: son modelos que utiliza mecanismos de atención para modelar dependencias a largo plazo entre elementos de la secuencia [14].

Existen CNNs en la literatura que ya tienen arquitecturas definidas y que tienen buenos resultados para diversos propósitos. Las arquitecturas seleccionadas se han utilizado ampliamente en el reconocimiento de expresiones faciales y han demostrado un alto rendimiento en tareas de clasificación de emociones en imágenes. Su inclusión en este trabajo permite establecer una línea base robusta que facilita la comparación con modelos propuestos de mayor complejidad. Además, estas redes sirven como punto de partida para adaptar soluciones eficientes que puedan ejecutarse en tiempo real sobre sistemas embebidos, que es el objetivo principal de este trabajo de investigación [15][16][17].

1.2.1 MobileNet

Esta arquitectura fue concebida con el objetivo de optimizar la eficiencia en cómputo y en el uso de la memoria, de modo que pudiera emplearse en dispositivos móviles o sistemas embebidos con recursos de hardware limitados.

Esta arquitectura hace uso de convoluciones separables en profundidad o *depthwise convoluctions* por su nombre en inglés, que separan la operación de convolución en dos pasos: un primer proceso denominado convolución *depthwise*, donde se aplica cada filtro a un único canal de entrada, y un segundo proceso *pointwise* (1×1), en el que se toman las salidas de la etapa anterior para ajustar la dimensionalidad. Esta estrategia reduce significativamente la cantidad de parámetros y el costo computacional respecto a las redes convolucionales tradicionales que utilizar convoluciones completas. Además, MobileNetV1 recurre a la función de activación ReLU y a la normalización por lotes para estabilizar el entrenamiento.

Estas características hacen que la arquitectura sea ampliamente empleada en reconocimiento de imágenes, detección de objetos y segmentación semántica con requerimientos de baja latencia, como se ilustra en la Figura 1.1.

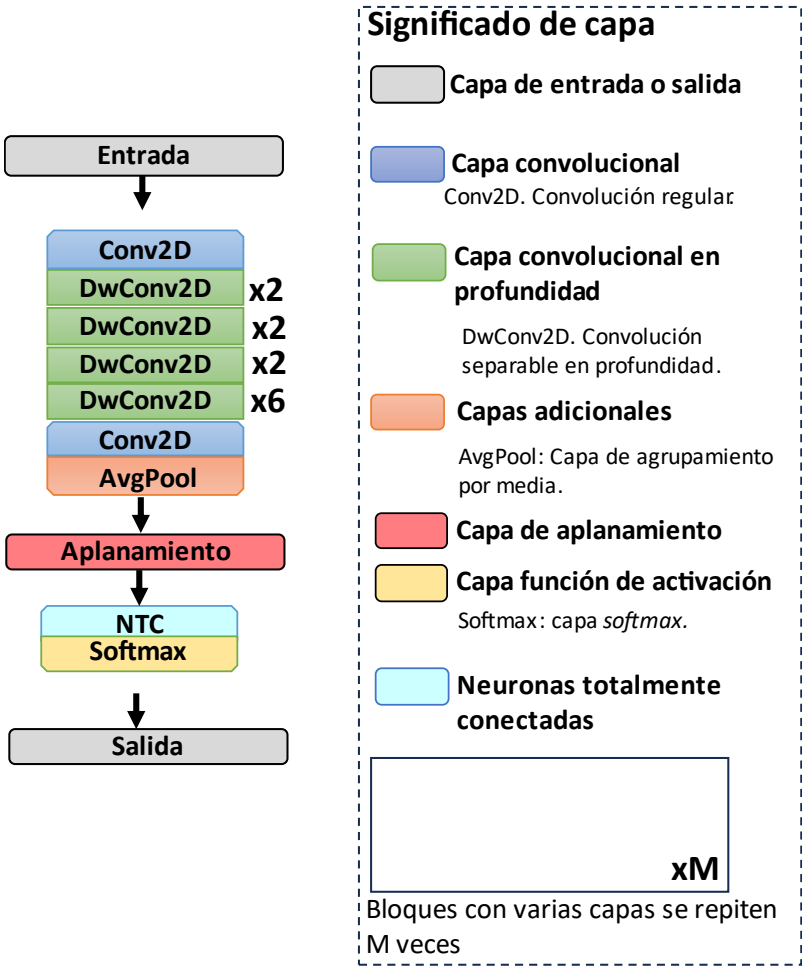


Figura 1.1 Diagrama representativo de la arquitectura de MobileNetV1.

1.2.2 ResNet50

ResNet50 es una red profunda diseñada para afrontar el problema del desvanecimiento de gradiente que puede surgir en redes con muchas capas, que consiste en que en cuanto más profunda es una red, las primeras capas de estas suelen no aprender mucho a lo largo de las épocas de entrenamiento.

Su innovación principal radica en la introducción de los bloques residuales, que facilitan la propagación del gradiente a través de conexiones de atajo o *skip connections* por su nombre

en inglés. Estas conexiones permiten sumar la entrada original de un bloque a la salida, evitando que el error se diluya a medida que la profundidad de la red aumenta. ResNet50 consiste en 50 capas y utiliza, en cada bloque, convoluciones 1×1 y 3×3 para reducir y restaurar la dimensionalidad de los mapas de características, seguido de la correspondiente normalización y activación para estabilizar el entrenamiento.

Esta arquitectura ha demostrado alta precisión en tareas de clasificación de imágenes, detección de objetos y segmentación semántica, su arquitectura puede verse tal en la Figura 1.2.

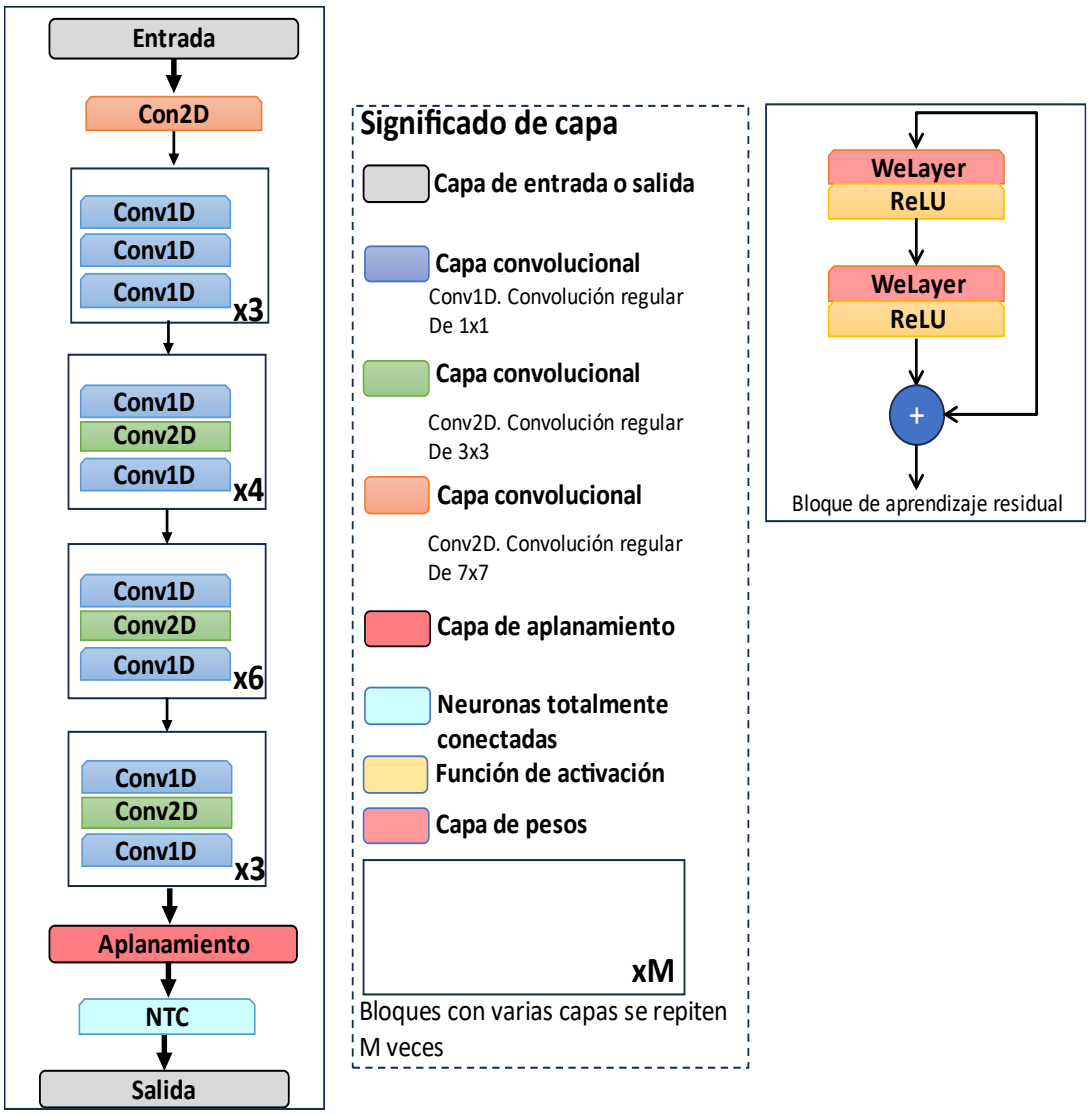


Figura 1.2 Diagrama representativo de la arquitectura de ResNet50.

1.2.3 EfficientNetB0

EfficientNetB0 es el modelo base de la familia EfficientNet, reconocida por lograr un balance óptimo entre precisión y uso de recursos computacionales. Su diseño se sustenta en un enfoque de escalamiento compuesto que equilibra, de manera uniforme, la profundidad, el ancho y la resolución de la red.

Al compararla con otras arquitecturas, EfficientNetB0 destaca por utilizar menos parámetros y requerir menor cantidad de operaciones de punto flotante (FLOPs) sin sacrificar la capacidad de generalización. Asimismo, se basa en los denominados MBConv (*Mobile Inverted Bottleneck Convolution*), inspirados en MobileNetV2, que combinan convoluciones *depthwise* 3×3 o 5×5 con convoluciones 1×1 para la expansión y compresión de canales, además de recurrir a conexiones de salto opcionales.

Se emplean funciones de activación más avanzadas, como Swish o SiLU, con el fin de mejorar la eficiencia del aprendizaje. El diagrama de su arquitectura puede observarse en la Figura 1.3 y se utiliza sobre todo en aplicaciones de visión por computadora donde la eficiencia y la velocidad de procesamiento son cruciales para el propósito en que se usará la red.

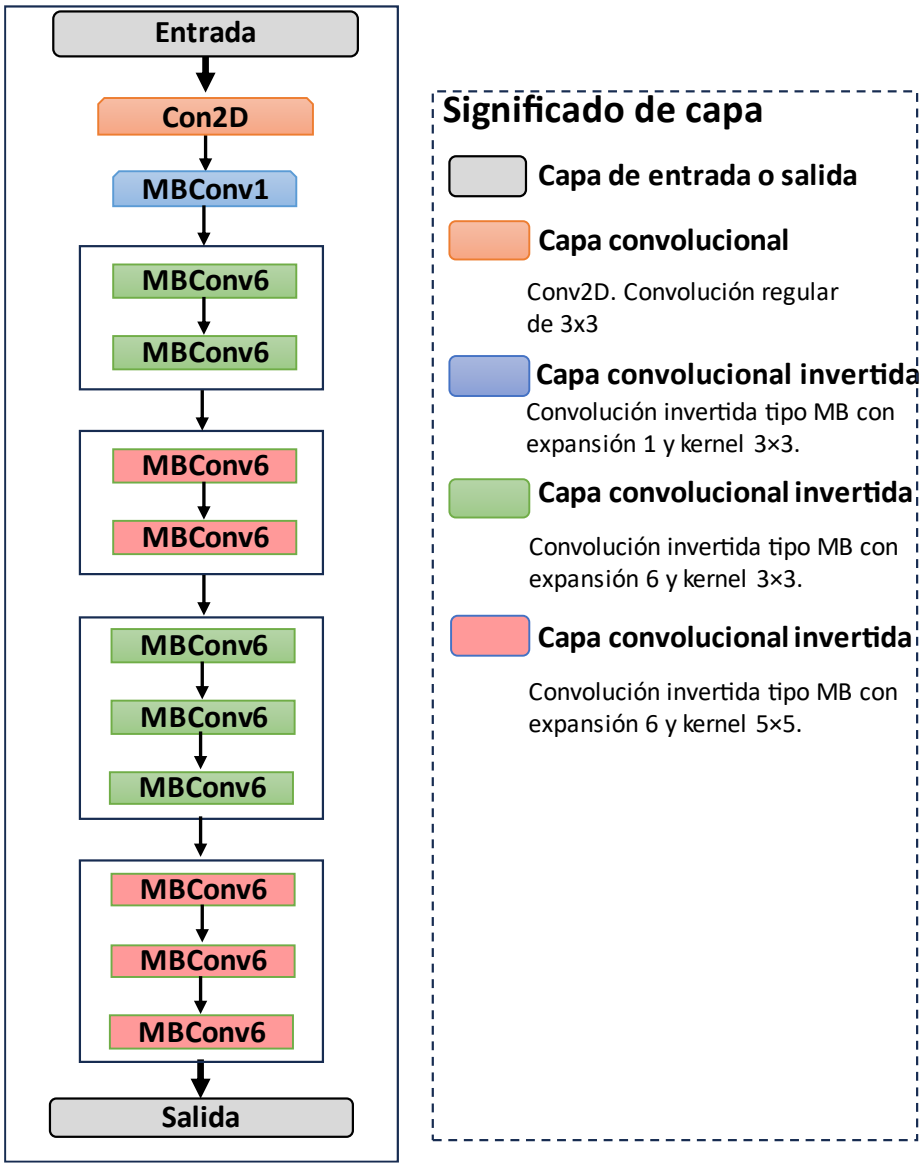


Figura 1.3 Diagrama representativo de la arquitectura de EfficientNetB0.

1.2.4 SqueezeNet

El objetivo principal de SqueezeNet recurre a los denominados *Fire Modules*, compuestos por una etapa de compresión (*squeeze*) y otra de expansión (*expand*). En la etapa de compresión se emplean convoluciones 1×1 para disminuir el número de canales, reduciendo el costo computacional.

En la etapa de expansión, se utilizan convoluciones 1×1 y 3×3 en paralelo de esta manera los canales se amplían nuevamente y así es posible extraer múltiples tipos de características. Esta estrategia permite que el modelo ocupe poco espacio en memoria y, por lo tanto, sea especialmente indicado para dispositivos con capacidades limitadas. El diagrama representativo de SqueezeNet se muestra en la Figura 1.4.

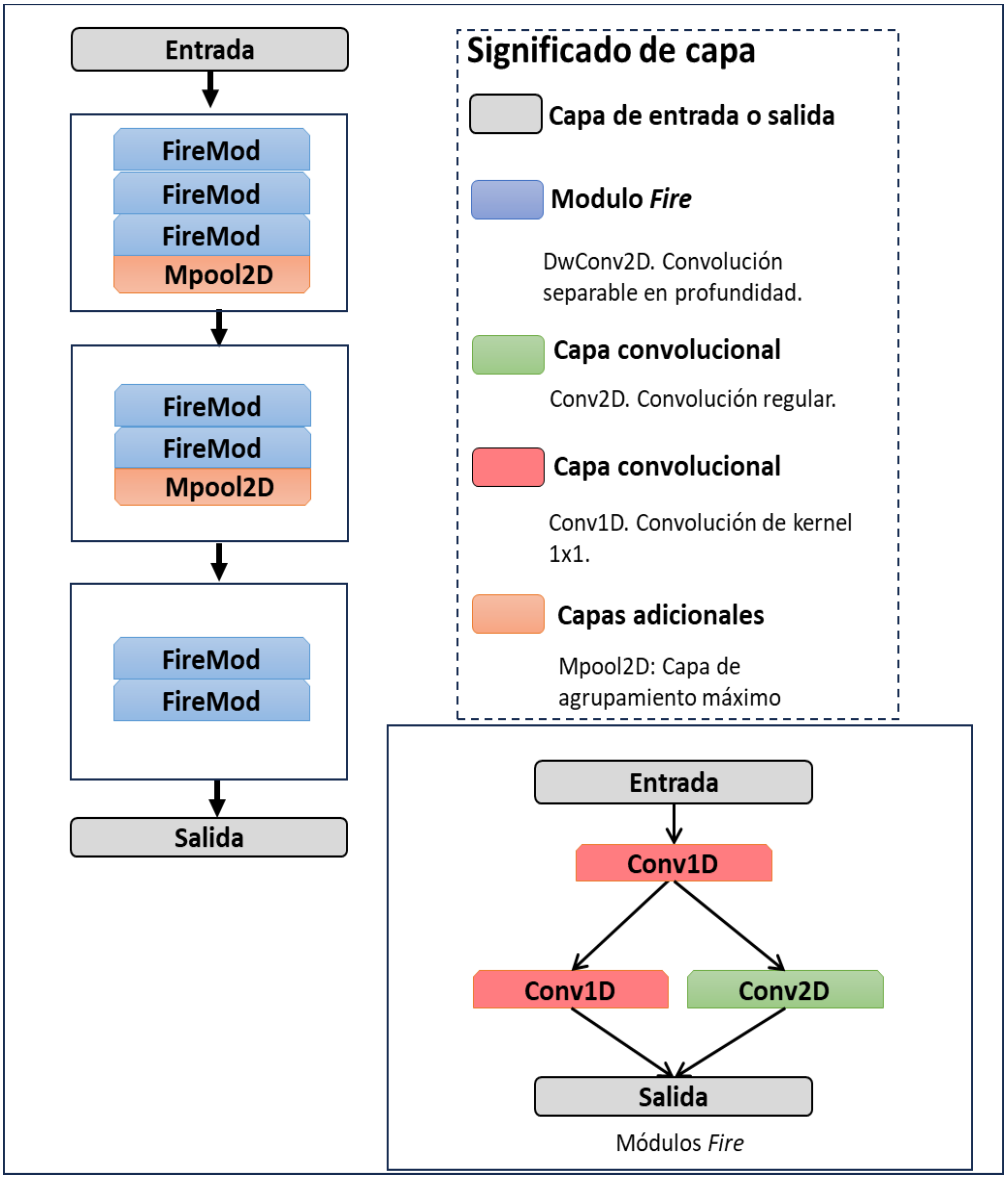


Figura 1.4 Diagrama representativo de la arquitectura de SqueezeNet.

1.2.5 ShuffleNet

ShuffleNet fue desarrollada con la meta de ofrecer un modelo altamente eficiente para dispositivos de baja potencia. Para lograrlo, recurre a la técnica de cambio de canal o *channel shuffle*, la cual mezcla los canales de entrada entre distintos grupos de convolución para garantizar que cada grupo procese información diferente en las siguientes capas, después estas características pasan a capas convoluciones separables en profundidad y capas de convoluciones agrupadas para después ser concatenadas. Esta estrategia maximiza la reutilización de características y facilita el flujo de gradientes con un coste computacional reducido.

ShuffleNet ha demostrado ser muy útil en aplicaciones que necesitan un alto rendimiento en tiempo real y disponen de recursos de hardware escasos, como se aprecia en la Figura 1.5.

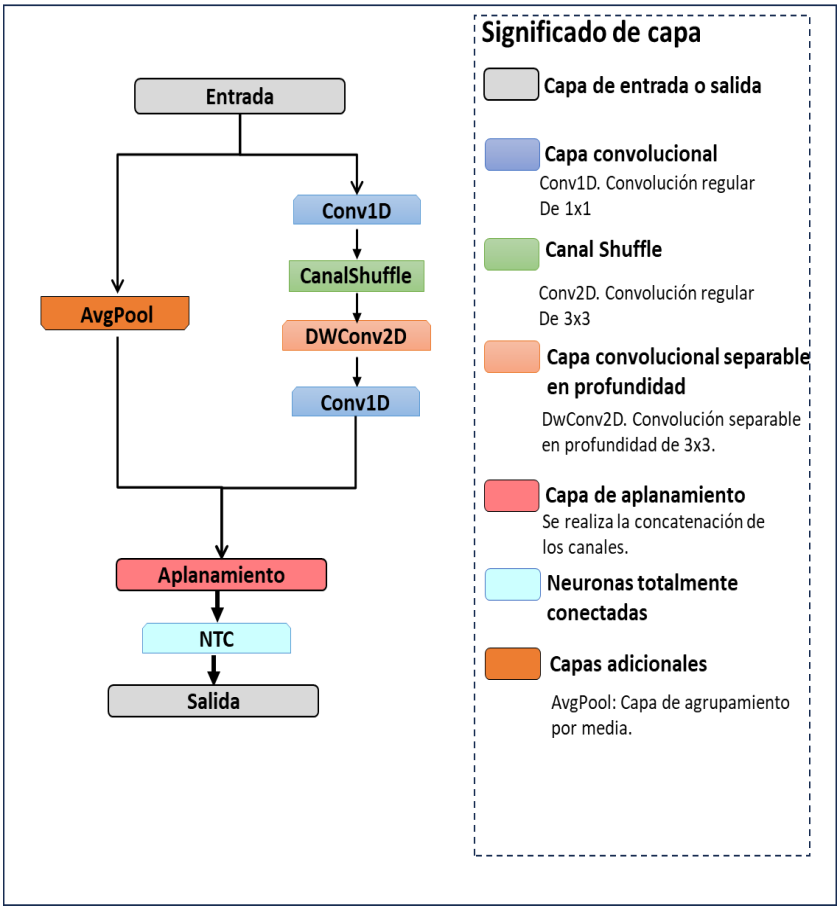


Figura 1.5 Diagrama representativo de la arquitectura de ShuffleNet.

1.2.6 VGG16

VGG16 se compone de 16 capas y se caracteriza por su simplicidad y por seguir un patrón muy uniforme. Cada bloque se integra por varias capas convolucionales de tamaño 3×3 seguidas de una operación de *pooling* (2×2) que reduce la resolución espacial de las características. Al final, se ubican capas totalmente conectadas.

Debido a su gran tamaño tiene un alto consumo de memoria además de un tiempo alto de inferencia debido a su gran cantidad de parámetros, sin embargo, esta arquitectura ha sido muy utilizada en clasificación de imágenes a gran escala. Su diagrama general se muestra en la Figura 1.6.

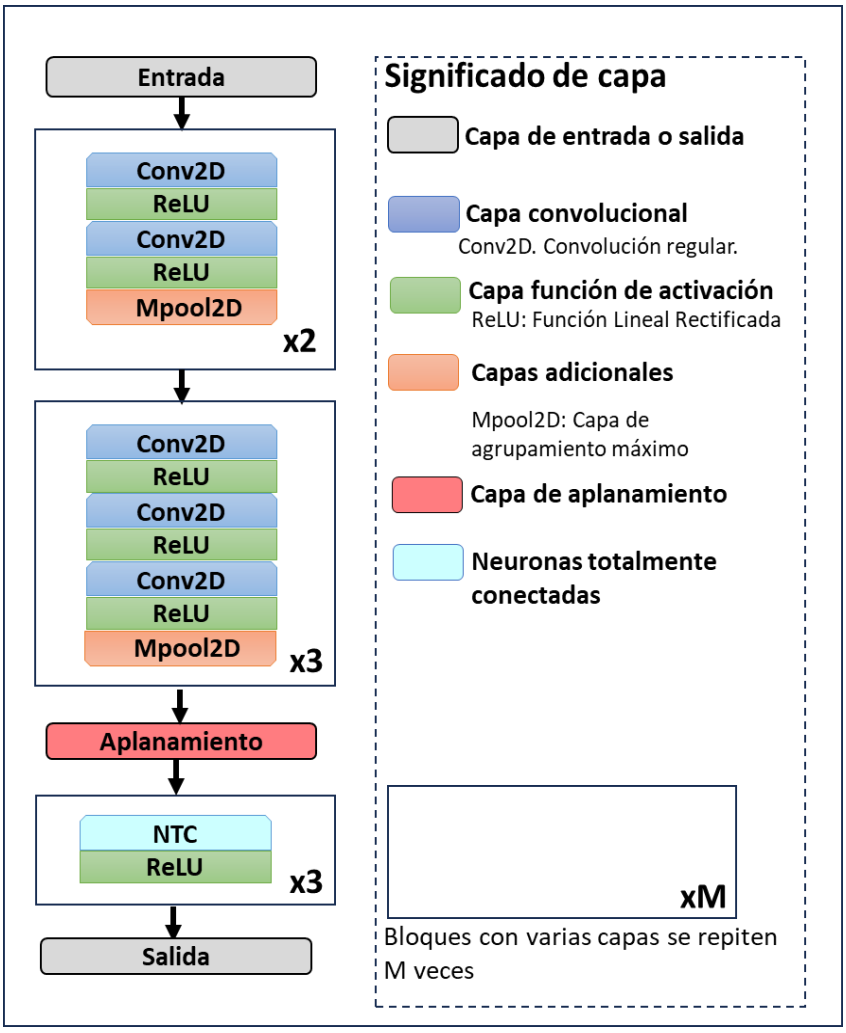


Figura 1.6 Diagrama representativo de la arquitectura de VGG16.

1.2.7 InceptionV1

InceptionV1, o GoogLeNet, introdujo la idea de los bloques *Inception*, que permiten procesar información a diferentes escalas en una misma capa. En cada bloque se utilizan convoluciones 1×1, 3×3 y 5×5 de forma paralela, junto con operaciones de *pooling*, estas características de cada bloque se concatenan utilizando convoluciones 1×1 que actúan como cuellos de botella para reducir la dimensionalidad y, por consiguiente, los requisitos de cómputo.

Este enfoque aprovecha la diversidad de tamaños de filtros para extraer características de distinta granularidad. La Figura 4.7 muestra la estructura básica de un bloque *Inception* y su funcionamiento dentro de la red.

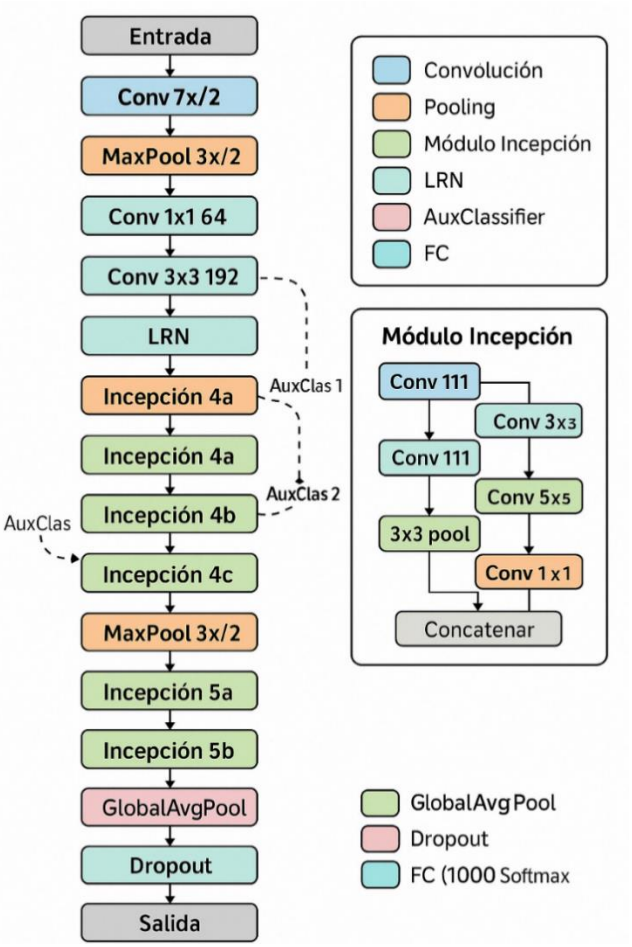


Figura 1.7 Diagrama representativo de la arquitectura de InceptionV1.

1.2.8 DenseNet121

DenseNet121 se basa en conexiones densas, donde cada capa se vincula con todas las capas previas dentro del mismo bloque. En lugar de sumar las salidas, como ocurre en ResNet, DenseNet las concatena, lo que favorece la reutilización de características y el flujo de gradiente, reduciendo el problema del desvanecimiento de gradiente que ocurre en modelos muy profundos a lo largo de las épocas de entrenamiento.

Los bloques densos (*Dense Blocks*) suelen contener capas compuestas de convoluciones 3×3 o 1×1, seguidas de normalización por lotes y función de activación. Entre un bloque y otro, se introducen capas de transición (*Transition Layers*) con convoluciones 1×1 y *pooling*, diseñadas para controlar la explosión en el número de canales y mantener la eficiencia de la arquitectura.

DenseNet121 es apreciada en tareas de clasificación y segmentación de imágenes, ya que ofrece una buena compensación entre rendimiento y eficiencia, su arquitectura se observa en la Figura 1.8.

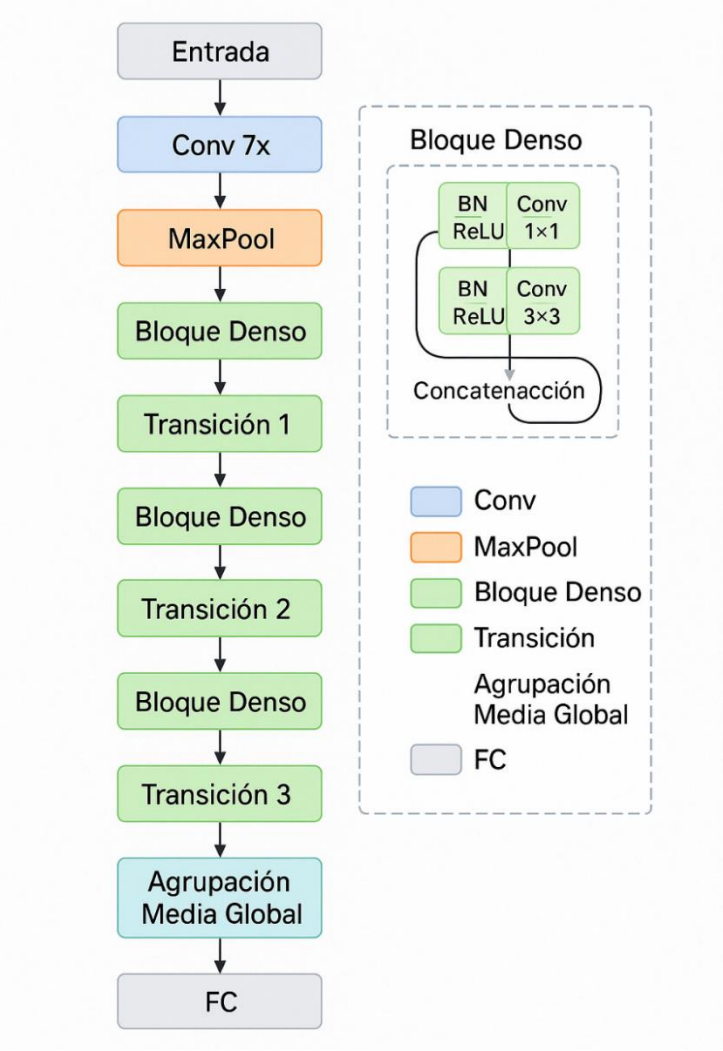


Figura 1.8 Diagrama representativo de la arquitectura de DenseNet121.

1.3 Emociones

Las emociones son respuestas psicológicas y fisiológicas a estímulos internos o externos a nosotros. Estas respuestas se ven reflejadas en el rostro haciendo que pequeñas características como la posición de la boca, cejas o los ojos se puedan interpretar como un comportamiento que demuestra una cierta emoción. Debido a que las emociones cumplen un papel crucial en la manera en que nos relacionamos con el mundo que nos rodea, influyendo en decisiones, comportamientos y percepciones, es importante considerar que cada emoción puede jugar un papel importante a la hora de tomar decisiones y esto puede ocasionar accidentes si no se tiene un debido cuidado.

En su momento, el psicólogo Paul Ekman propuso una clasificación de emociones la cual divide las mismas en 7 emociones básicas [18], aunque varios autores han participado en otro tipo de clasificación que no es atribuible a un solo investigador, sino que ha sido un enfoque ampliamente utilizado en el estudio de la psicología y la teoría de las emociones que dividen las emociones en positivas, negativas y neutras, es el caso de Barbara Fredrickson en su obra *"Positivity"* [19], que habla de cómo las emociones positivas expanden la atención y los recursos de una persona a través de su teoría del 'ampliar y construir' (broaden-and-build), además de Richard S. Lazarus, quien en su obra *"Emotion and Adaptation"* [20] propuso enfoques sobre la valoración cognitiva de emociones, diferenciando las respuestas emocionales según cómo se percibe y evalúa una situación y Robert Plutchik en su obra *"Emotion: A Psychoevolutionary Synthesis"* [21], que introdujo su 'rueda de las emociones' para clasificar emociones primarias que pueden combinarse para formar emociones más complejas, inspirando enfoques modernos para clasificar emociones en términos de valencia (positivas o negativas) e intensidad.

A partir de estas investigaciones se han desarrollado bases de datos que se ha consolidado como un referente en la clasificación de emociones faciales, utilizando siete categorías basadas en las expresiones emocionales universales propuestas por Ekman. Estas categorías incluyen tanto emociones positivas como negativas y sirven para el entrenamiento y evaluación de modelos de reconocimiento de emociones en el ámbito de la visión por computadora. A continuación, se detallan las características y distinciones de dichas emociones.

Neutralidad:

Expresión neutral:

- Representa un estado sin emociones aparentes.
- Su función principal es servir como punto de referencia o comparación para identificar y diferenciar otras expresiones emocionales.

Emociones positivas:

Alegría:

- Se manifiesta mediante una sonrisa visible.
- Las mejillas se levantan como resultado de la activación de los músculos faciales asociados al placer.
- Aparecen arrugas alrededor de los ojos, especialmente conocidas como "patas de gallo", indicando una sonrisa genuina (sonrisa de Duchenne).

Sorpresa:

- Se caracteriza por una apertura repentina de los ojos, que se agrandan notablemente.
- Las cejas se elevan de forma pronunciada, lo que refleja un aumento en la atención ante un estímulo inesperado.
- La boca se abre, generalmente sin emitir sonido, como una expresión instintiva de asombro o impacto.

Emociones negativas:

Tristeza:

- Se refleja en la caída de las comisuras de la boca, lo que da un aspecto decaído al rostro.
- Las cejas muestran un arqueo leve hacia arriba en el centro, lo que sugiere una expresión de sufrimiento o melancolía.
- Esta emoción suele estar asociada a una disminución en la energía facial y corporal.

Ira:

- Se evidencia a través del fruncimiento del ceño, lo que crea líneas visibles entre las cejas.
- Hay una notable tensión muscular en todo el rostro, especialmente en la mandíbula y la frente.
- La mirada se vuelve fija e intensa, con una expresión de confrontación o desafío.

Disgusto:

- Se identifica por la aparición de una arruga leve en la nariz, como si se intentara bloquear un olor desagradable.
- Los labios se mantienen apretados, a menudo formando una línea recta o curvada hacia abajo.

- Se percibe tensión en la zona de la boca, lo que transmite rechazo o repulsión.

Miedo:

- Los ojos se abren ampliamente, reflejando un estado de vigilancia y búsqueda de información visual.
- Las cejas se arquean hacia arriba y se juntan ligeramente, generando una expresión de preocupación.
- La boca se mantiene entreabierta, a menudo en preparación para una posible reacción de huida o grito.
- Esta expresión comunica una respuesta automática ante una amenaza percibida.

Existen también bases de datos basadas en el *Facial Action Coding System* (FACS), un sistema desarrollado por Ekman y Friesen para codificar expresiones faciales a partir de movimientos musculares específicos. A diferencia de otras este tipo de bases de datos incorporan un conjunto más amplio de expresiones, incluyendo ocho categorías emocionales que abarcan tanto emociones positivas como negativas.

Emociones positivas:

Felicidad:

- Se manifiesta mediante una sonrisa amplia que involucra tanto la boca como los músculos alrededor de los ojos.
- Las mejillas se elevan, creando un efecto de plenitud en el rostro.
- Los ojos suelen entrecerrarse levemente, lo cual es indicativo de una expresión genuina y espontánea de alegría.

Sorpresa:

- Se caracteriza por una apertura pronunciada de los ojos, lo que refleja un aumento repentino de atención.
- Las cejas se elevan marcadamente, intensificando la expresión de asombro.
- La boca se abre de manera notable, a menudo sin producir sonido, como respuesta automática ante lo inesperado.

Emociones negativas:

Tristeza:

- Se reconoce por la caída de las comisuras de la boca, dando una apariencia abatida al rostro.
- Las cejas se arquean hacia el centro, formando una expresión que refleja sufrimiento o pena.
- Esta emoción suele acompañarse de una disminución general en el tono muscular facial.

Ira:

- Se expresa mediante el fruncimiento del ceño, generando líneas verticales entre las cejas.
- Los labios se mantienen apretados, reflejando contención emocional y tensión.
- Todo el rostro presenta una tensión visible, especialmente en la mandíbula, lo que transmite confrontación o agresividad.

Disgusto:

- Se identifica por una contracción visible en la nariz, como si se intentara rechazar un olor desagradable.
- Las cejas se fruncen, reforzando la expresión de desagrado.
- Los labios están apretados, lo que sugiere rechazo, incomodidad o repulsión.

Miedo:

- Los ojos se abren exageradamente, indicando una reacción inmediata de alerta.
- Las cejas se levantan y se posicionan de forma arqueada, generando una expresión de preocupación o pánico.
- La boca se mantiene ligeramente entreabierta, como preparación para una respuesta rápida ante el peligro (por ejemplo, huida o grito).

Desprecio:

- Se distingue por una expresión más sutil en comparación con otras emociones negativas.
- Generalmente, se observa un levantamiento unilateral del labio superior (sólo de un lado).
- Esta expresión comunica desaprobación, desdén o un sentimiento de superioridad frente a otra persona o situación.

El uso de bases de datos basadas en FACS permite un análisis más detallado de las micro expresiones faciales, lo que la hace una base de datos clave en estudios de visión por computadora y en el desarrollo de modelos de reconocimiento emocional, especialmente en aplicaciones de interacción humano-máquina y monitoreo de estados afectivos.

1.4 Estados afectivos

Los estados afectivos son condiciones emocionales transitorias que reflejan cómo una persona se siente en un momento determinado. A diferencia de las emociones básicas los estados afectivos abarcan un efecto continuo de activación y valencia [22].

Estados afectivos negativos prolongados pueden activar el eje hipotálamo-hipófisis-adrenal (HHA), generando cortisol, lo que mantiene al cuerpo en un estado de alerta constante llevando a la persona a tener estrés [23]. Además, estados afectivos negativos como la inquietud, el miedo o la sensación de descontrol pueden llevar a trastornos de ansiedad generalizada o ataques de pánico [24] y en el peor de los casos un predominio de estados afectivos negativos sostenidos puede desencadenar síntomas depresivos [25].

Sin embargo, la detección de los estados afectivos en las personas mediante expresiones faciales no sería de mucha ayuda si el sistema no funcionara en tiempo real, debido a que las interacciones que pueden intervenir en las acciones de una persona que enfrenta una situación estresante deben ser rápidas para prevenir incidentes. Bajo este supuesto, se quiere realizar una red que funcione en tiempo real, si bien tiempo real es un tema que muchos relacionan con la inmediatez de las acciones de una máquina esto no es del todo cierto, realmente el concepto de tiempo real hace referencia a que los procesos funcionen en tiempos determinados sin que el sistema sea interrumpido por otro proceso, esto hace que un sistema tenga tiempos determinados para cada función y así no existan interrupciones que puedan realizar un mal funcionamiento.

1.5 Tiempo real

Aunque es posible utilizar sistemas de cómputo generales como máquinas que alberguen un sistema en tiempo real esto no siempre es la mejor opción, en especial porque este tipo de sistemas tienen tiempos de respuesta más largos en general por el uso de capas de software extra que generan abstracción y por lo tanto tiempos de respuesta entre ciclo del procesador, por lo que la mejor opción para albergar sistemas de este tipo son los sistemas embebidos que en sí, son computadoras de propósito específico cuya única tarea será realizar las acciones pertinentes para que el sistema en tiempo real funcione en tiempo y forma.

Existen muchos tipos de sistemas embebidos, realmente los núcleos de este tipo de sistemas tienen claras diferencias debido a que tanto su configuración o programación es diferente, un claro ejemplo de estos son los sistemas que tienen como núcleo un procesador basado en ARM® que son programados mediante programación explícita a diferencia de los sistemas con núcleo FPGA o CPL que son configurados mediante síntesis de hardware, y aunque estos sistemas tienen su grado de complejidad para su configuración/programación en si hay sistemas que requieren un grado más de investigación y desarrollo para albergar sistemas complejos, como lo es una red neuronal artificial en tiempo real como lo son los sistemas basados en FPGA.

Aunque existen varios modelos en el estado del arte que clasifican emociones [26] y a su vez estos son usados para la detección de estrés, estos modelos generalmente son extensivos con miles o millones de parámetros que les permiten extraer las características más pequeñas de las expresiones faciales pero estos modelos, en general, son inviables para sistemas en tiempo real debido a que al tener tantos parámetros y ser tan profundos necesitarían de mucho tiempo de procesamiento y en sistemas embebidos pequeños esto provocaría que el sistema tuviera tiempos empalmados o que el proceso durara más de lo que se requeriría en la detección del estrés para, por ejemplo, correr un subsistema de control.

2. ANTECEDENTES

A continuación, se presentan los artículos más importantes que respaldan este documento de investigación. Este capítulo se divide de la siguiente manera: la sección 2.1 describe los trabajos referentes al procesamiento en tiempo real, mientras que la sección 2.2 muestra los trabajos que se relacionan a la clasificación de emociones y estados afectivos.

2.1 Procesamiento en tiempo real

W. Hu. et al. proponen en [27] Vis-TOP un procesador de superposición para varios modelos de *Transformers* visuales. Se utilizó un ZCU102 de Xilinx para implementar el procesador superpuesto. Como resultado, el rendimiento TOP es 1.5 mayor en comparación con los aceleradores *Transformers* existentes hasta el momento, por último, el rendimiento por procesamiento digital de señales es entre 2.2 y 11.7 veces mayor que otros.

H. Khan et al. [28] presentan NPE, este es un procesador superpuesto basado en FPGA capaz de ejecutar de manera eficiente una variedad de modelos PNL. Mediante un Xilinx Zynq Z-7100 FPGA se crea el procesador y se experimenta el algoritmo propuesto. Como resultado de los experimentos se demuestra que NPE utiliza 3 veces menos recursos FPGA en comparación con aceleradores específicos de red BERT, NPE proporciona una solución basada en FPGA rentable y energéticamente eficiente para el PNL en el borde.

F. Muhammad et al. [29] implementan un sistema de clasificación de géneros musicales en tiempo real basado en aprendizaje profundo en un Zynq UltraScale+ MPSoC ZCU104 con la plataforma Vitis IA con el objeto de diseñar un algoritmo que utilice pipelining y paralelismo para implementar un modelo CNN y LSTM. El resultado que se obtuvo en términos de precisión de las pruebas, donde el modelo CNN solo perdió menos del 1 % de la precisión de las pruebas en el SoC y en términos de ejecución el SoC sobrepasó el procesador Intel Core con un tiempo de 5.13 segundos contra 25.96 segundos. También, el SoC fue mejor con 192 imágenes por segundo frente a las 38 del Intel Core lo que significa que puede procesar más cuadros por segundo lo que es más adecuado para aplicaciones de procesamiento de video en tiempo real.

El trabajo realizado por F. Ghaffar [30], tiene como objetivo proponer un sistema de visión embebido para la adquisición y procesamiento de vídeo añadiendo aceleradores de hardware para extraer algunas características de la imagen. En este artículo, se da especial interés en el diseño e implementación de un sistema de visión integrado, el cual facilita la transmisión de vídeo desde la cámara al monitor y el procesamiento de hardware a través de FPGA-SoC en tiempo real.

2.2 Reconocimiento de emociones

C. Jain et al. [31] desarrollaron un trabajo que tiene como objetivo detectar rostros a partir de una imagen determinada, extrayendo los rasgos faciales y clasificándolos en seis emociones básicas: felicidad, miedo, enfado, asco, neutral, tristeza. En dicho trabajo, se implementan los enfoques del filtro Gabor o transformar imágenes usando histograma de gradientes orientados (HOG por sus siglas en inglés) y transformada de ondas discretas (DWT por sus siglas en inglés) par, para mejorar la clasificación de los datos, logrando el mejor resultado pasando las imágenes de entrenamiento a través de HOG, seguido de la caracterización por SVM, logrando así una precisión promedio del 85 % de su modelo.

A.A. Alsheikhy et al. [32] proponen un sistema eficiente de reconocimiento facial de siete emociones básicas: enfado, disgusto, miedo, alegría, tristeza, sorpresa y neutral. El sistema utiliza una red neuronal convolucional, la cual predice y asigna probabilidades a cada emoción. En este sistema se pasa cada imagen por el algoritmo de detección de rostros para incluirla en el conjunto de datos de entrenamiento y luego se duplicaron los datos usando varios filtros en cada imagen. Las imágenes preprocesadas son de un tamaño de 80×10080×100 pixeles y se utilizaron tres capas convolucionales, seguidas de una capa de agrupación y tres capas densas. El modelo se entrenó combinando las bases de datos JAFFE y KDEF. El 90 % de los datos se usó para entrenamiento y el resto para pruebas. Se logró una precisión máxima del 78.1 % utilizando el conjunto de datos combinado.

Gupta et al. [33] analizaron técnicas de extracción de características basadas en características prosódicas o características espectrales. Se descubrió que Coeficientes Cepstrales en las Frecuencias de Mel extraen las mejores características para reconocer contenido emocional. Como se observa en la figura 2.1, en este trabajo se propone un modelo red neuronal convolucional estocástica profunda óptimo para espectrograma sin procesar que resultó en una precisión no ponderada del 61 % para el espectrograma sin procesar (con ruido) y del 79 % para el espectrograma limpio (sin ruido).

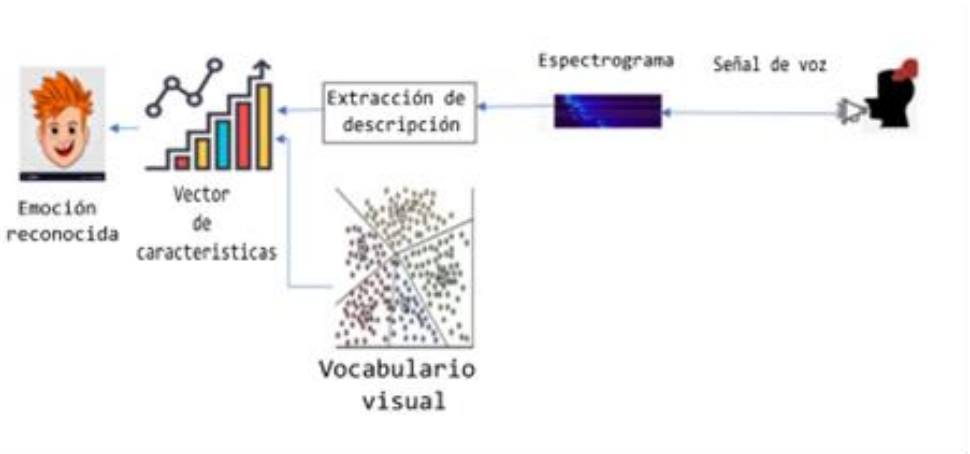


Figura 2.1 Reconocimiento de emociones a partir de la generación de espectrogramas de señales de voz [33].

B. Phavish et al. [34] exploran el reconocimiento de expresiones faciales humanas mediante un enfoque de aprendizaje profundo usando un algoritmo de red neuronal convolucional, utilizando un conjunto de datos etiquetados con alrededor de 32,298 imágenes con múltiples expresiones faciales para entrenamiento y prueba. La fase previa al entrenamiento implica un subsistema de detección de rostros con eliminación de ruido, incluida la extracción de características. Se genera un modelo de clasificación utilizado para la predicción, el cual puede identificar siete emociones del FACS. Los resultados presentan una precisión del 79.8 % para el reconocimiento de las siete emociones humanas básicas, sin la aplicación de técnicas de optimización.

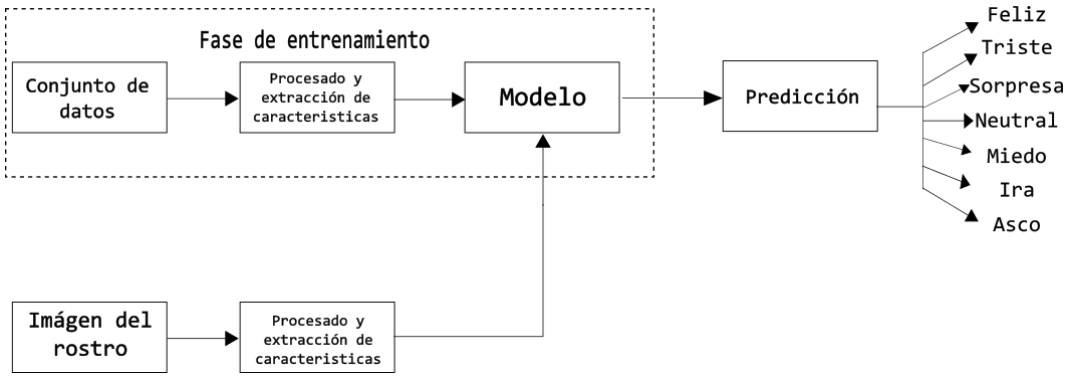


Figura 2.2 Arquitectura propuesta [34].

Amel Ali Alhussan et al. proponen en [35] un modelo de reconocimiento de expresiones faciales (FERM) basado en una Máquina de Vectores de Soporte (SVM) optimizada. Se utilizó el conjunto de datos AffectNet para probar el rendimiento del modelo. Como resultado, la precisión alcanzada fue del 99 % y la puntuación F1 fue del 98 %, demostrando una mejora significativa en la categorización de emociones en comparación con otros modelos. La optimización se realizó mediante búsqueda en cuadrícula para ajustar los hiperparámetros de la máquina de vectores de soporte, y se empleó análisis discriminante lineal para mejorar el proceso de clasificación.

Rajesh Singh et al. proponen en [36] un modelo híbrido para el reconocimiento de expresiones faciales en videos utilizando una combinación de una red neuronal convolucional tridimensional y una red de memoria a largo plazo convolucional (ConvLSTM por sus siglas en inglés). La arquitectura del modelo incluye tres capas de 3D-CNN, una capa de *max-pooling* 3D, un bloque ConvLSTM con 16 unidades, una capa totalmente conectada, y una capa de clasificación *softmax*. Este modelo se evaluó en los conjuntos de datos CK+, SAVEE, y AFEW. Específicamente, alcanzó una precisión del 95.10 % en el conjunto de datos CK+, del 98.83 % en SAVEE, y del 43.86 % en AFEW, demostrando un desempeño competitivo en el reconocimiento de expresiones faciales en tiempo real sin utilizar datos externos o técnicas de fusión de modelos.

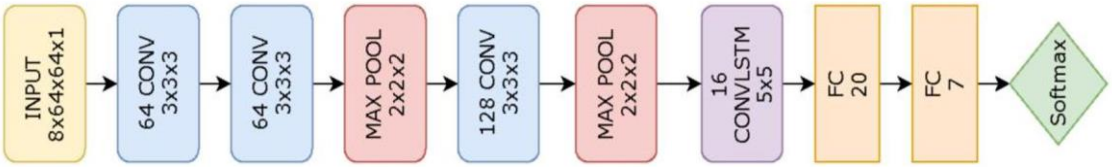


Figura 2.3 Representación esquemática del modelo híbrido propuesto de CNN 3D y CNN LSTM [36].

Naveen Kumar H N et al. proponen en [37] un sistema automático de reconocimiento de expresiones faciales (AFER) que combina características de textura y forma de regiones faciales prominentes. La arquitectura emplea descriptores como *Local Binary Pattern*, HOG, y *Local Phase Quantization* para extraer información discriminativa de las imágenes faciales. Se utiliza una Máquina de Vectores de Soporte Multiclase para la clasificación. Este modelo fue probado en los conjuntos de datos CK+, KDEF, y JAFFE, alcanzando una precisión del 94.2 % en CK+ y 93.7 % en KDEF, mejorando significativamente los métodos basados en características manuales existentes.

Dr. Shubhangi Milind Joshi et al. proponen en [38] un nuevo algoritmo para el procesamiento de imágenes en tiempo real, que utiliza técnicas avanzadas de procesamiento paralelo y extracción de características para mejorar la velocidad y precisión sin comprometer la calidad del procesamiento. El algoritmo se probó en áreas como sistemas autónomos, imágenes médicas, vigilancia y aplicaciones multimedia. Los resultados mostraron una mejora del 30 % en la velocidad de procesamiento en comparación con métodos tradicionales, manteniendo o mejorando la precisión hasta un 99 %. El algoritmo destaca por su capacidad de manejar ruido y características variadas de imágenes en diversas condiciones operativas.

Abey Litty propone en [39] un enfoque para el procesamiento de imágenes en tiempo real en bioinformática utilizando técnicas de aprendizaje profundo acelerado por GPU. El artículo se centra en aplicaciones clave como la imaginería genómica, el fenotipado celular y la patología de enfermedades, donde los modelos de aprendizaje profundo, como las redes CNN y las RNN, se optimizan para la aceleración en GPU. Los resultados mostraron que la aceleración en GPU reduce el tiempo de procesamiento de horas a segundos y mejora la

precisión en el análisis de imágenes bioinformáticas. El modelo propuesto logró una precisión del 95.8 % en secuencias genómicas, del 93.5 % en imágenes microscópicas y del 97.2 % en imágenes histopatológicas, con un rendimiento significativamente más rápido que los métodos tradicionales.

Khare et al. proponen en [40] una revisión sistemática del reconocimiento de emociones utilizando inteligencia artificial, abarcando estudios entre 2014 y 2023. El artículo analiza múltiples modalidades para la detección emocional, incluyendo señales físicas (como el habla y las expresiones faciales) y fisiológicas. A través de un análisis exhaustivo de 142 artículos seleccionados bajo las directrices PRISMA, los autores identifican las técnicas más utilizadas para la extracción de características, modelos de aprendizaje automático y profundo, así como los conjuntos de datos más relevantes. Los resultados indican que el uso de técnicas de descomposición no lineal y la fusión de información entre distintas modalidades mejoran significativamente la precisión de los sistemas de reconocimiento emocional, alcanzando precisiones cercanas al 100 % en algunos estudios. Además, el artículo destaca la necesidad de bases de datos multimodales, modelos explicables y sistemas portables como futuras direcciones de investigación en este campo.

3. MATERIALES Y MÉTODOS

En este esté capítulo se describirán las bases de datos y experimentos que sirvieron como base para el desarrollo de la investigación. Los experimentos consistieron en aplicar las arquitecturas vistas en la sección 1.2 con las bases de datos de expresiones faciales más populares en la literatura.

Este capítulo se organiza de la siguiente manera: la sección 3.1 describe las bases de datos utilizadas en esta tesis. La sección 3.2 reporta los experimentos realizados con las diferentes arquitecturas para el reconocimiento de emociones. Finalmente, la sección 3.3 presenta conclusiones que derivaron en la propuesta de LiExNet.

3.1. Bases de datos

Las bases de datos empleadas en esta tesis reúnen una cantidad significativa de imágenes de expresiones faciales representadas como $I(m,n)^{RGB}$ $d$onde m y n es la posición espacial de cada píxel y RGB indica los canales de color rojo, verde y azul. Cada muestra $I(m,n)^{RGB}$ se normalizan mediante la siguiente expresión:

$$I_{NORM}(m,n) = \frac{I(m,n)^{RGB}}{255} \tag{3.1}$$

donde $I(m,n)$ representa la intensidad del píxel en cada canal de color (rojo, verde o azul) e $I_{NORM}(m,n)$ es la imagen resultante con valores en el rango [0,1]. Posteriormente, cada muestra se redimensiona a una resolución fija de 224×224 píxeles con el fin de estandarizar las dimensiones de entrada a las redes neuronales artificiales descritas en la Sección 1.2. De este modo, se garantiza la coherencia espacial y cromática necesaria para el entrenamiento y la inferencia en tiempo real.

Las bases de datos seleccionadas para los experimentos fueron:

- EMOTION-ITCH

- *Japanese Female Facial Expression* (JAFFE)
- *Extended Cohn-Kanade* (CK+)
- *Karolinska Directed Emotional Faces* (KDEF)
- *Facial Expression Recognition* 2013 (FER2013)

Las bases de datos CK+, JAFFE, KDEF y FER2013 se eligieron por ser las más populares en la literatura para el reconocimiento automático de emociones faciales. Estas bases de datos están consolidadas en la literatura por contar con un balance adecuado de muestras y porque ofrecen anotaciones bien definidas de las siete emociones universales: ira, disgusto, miedo, alegría, tristeza, sorpresa y neutral. En conjunto, permiten evaluar el rendimiento de los modelos bajo condiciones variadas de captura, resolución y diversidad demográfica.

Por otro lado, EMOTION ITCH es un repositorio propio desarrollado en el TecNM - Instituto Tecnológico de Chihuahua. Asimismo, al haber sido registrada sin condiciones controladas de laboratorio y con dispositivos de adquisición EEG de acceso abierto, esta base aproxima mejor el entorno operativo final en el que se busca implementar el sistema propuesto.

3.1.1. EMOTION-ITCH

EMOTION-ITCH es una base de datos desarrollada en el Instituto Tecnológico de Chihuahua (ITCH) y se compone de imágenes de expresiones faciales de 26 sujetos, con 5 imágenes correspondientes a cada tipo de emoción por sujeto. Esta base de datos clasifica las emociones en tres grandes categorías: neutral, positivas y negativas. Además de las imágenes, la base de datos contiene información complementaria sobre el estado emocional de cada participante como EGG y Audio. Las etiquetas de cada emoción son validadas mediante el cuestionario SAM (*Self-Assessment Manikin*).

Los participantes fueron citados tras su jornada laboral o en estado de descanso, y se registró si se encontraban activos laboralmente o no, con el objetivo de utilizar esta variable

como un segundo ground truth (GT), el cual se realizó para reforzar la detección de estados afectivos negativos relacionados con el entorno laboral.

La Figura 3.1 muestra algunas de las imágenes pertenecientes a la base de datos EMOTION-ITCH.

Figura 3.1 Muestras de la base de datos EMOTION-ITCH.

3.1.2. JAFFE

JAFFE es una colección de imágenes que contiene expresiones faciales de 10 mujeres japonesas que se desarrolló con el objetivo de investigar el reconocimiento de emociones a partir de la expresión facial. Esta base de datos se ha utilizado ampliamente en investigaciones de análisis facial debido a la alta calidad y la estandarización de las expresiones capturadas. La composición de JAFFE es la siguiente:

- 213 imágenes de expresiones faciales.
- 10 mujeres japonesas como sujetos de prueba.
- Entre 3 a 4 imágenes por sujeto para cada emoción.
- La base de datos JAFFE clasifica las emociones en las seis categorías de Eckman y neutralidad: Neutral, alegría (positiva), tristeza (negativa), ira (negativa), sorpresa (positiva), disgusto (negativa), miedo (negativa).

La Figura 3.2 muestra algunas de las imágenes pertenecientes a la base de datos JAFFE.

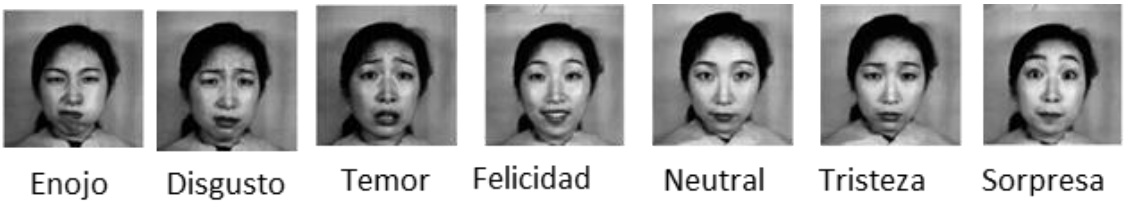


Figura 3.2 Muestras de la base de datos JAFFE [41].

3.1.3. CK+

La base de datos CK+ (*Extended Cohn-Kanade Dataset*) es una colección de secuencias de videos que contienen expresiones faciales de individuos en diferentes emociones. Esta base de datos se ha utilizado ampliamente en investigaciones sobre reconocimiento de emociones y análisis de expresiones faciales debido a su gran tamaño y la inclusión de múltiples emociones. La composición de este conjunto de datos es la siguiente:

- 593 secuencias de video que capturan la transición desde una expresión neutral hasta una expresión emocional completa.
- 123 sujetos de diversas edades y etnias, lo que proporciona una mayor variabilidad en las características faciales.
- Las secuencias se capturaron desde posiciones frontales bajo condiciones controladas de iluminación y postura.

La base de datos CK+ clasifica las emociones en ocho categorías, basadas en el sistema de codificación FACS (*Facial Action Coding System*) de Ekman, abarcando tanto emociones positivas como negativas:

- Neutral, felicidad (positiva), tristeza (negativa), ira (negativa), sorpresa (positiva), disgusto (negativa), miedo (negativa), desprecio (negativa).

La Figura 3.3 muestra algunas de las imágenes pertenecientes a la base de datos CK+.

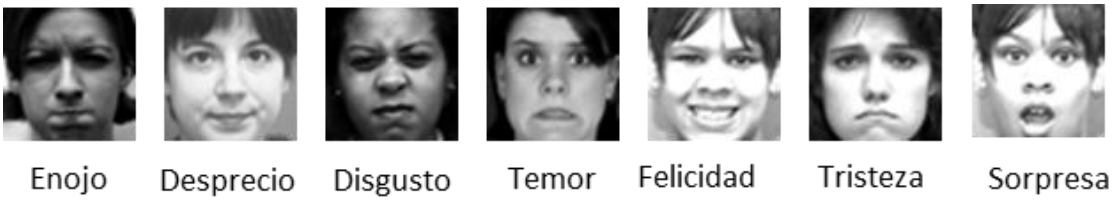


Figura 3.3 Muestras de la base de datos CK+ [42].

3.1.4. Base de Datos KDEF

La base de datos *Karolinska Directed Emotional Faces* (KDEF) es un conjunto de imágenes diseñado para el estudio del reconocimiento de emociones y expresiones faciales. Fue desarrollado por el *Karolinska Institute* en Suecia, con el objetivo de proporcionar un recurso estandarizado para investigaciones en psicología, neurociencia y visión por computadora. La base de datos incluye imágenes de individuos de diferentes géneros y con expresiones faciales variadas en condiciones controladas. La composición de este conjunto de datos es la siguiente:

- 4,900 imágenes de expresiones faciales.
- 70 sujetos (35 hombres y 35 mujeres).
- Cada sujeto presenta 7 emociones en 5 ángulos diferentes (frontal y cuatro variaciones laterales).

La base de datos KDEF clasifica las emociones en siete categorías distintas:

- Neutral, felicidad (positiva), tristeza (negativa), ira (negativa), sorpresa (positiva), disgusto (negativa), miedo (negativa).

KDEF es ampliamente utilizada en estudios de percepción emocional, reconocimiento de rostros y afectividad, y se ha aplicado en modelos de inteligencia artificial para mejorar la detección de emociones en imágenes estáticas.

La Figura 3.4 muestra algunas de las imágenes pertenecientes a la base de datos KDEF.

Figura 3.4 Muestras de la base de datos KDEF [43].

3.1.5. Base de Datos FER2013

La base de datos *Facial Expression Recognition* 2013 (FER2013) es un conjunto de datos creado específicamente para la clasificación automática de expresiones faciales en imágenes de rostros. Fue introducida en la competencia ICML 2013 *Challenges in Representation Learning* y se ha convertido en una referencia clave en el desarrollo de modelos de aprendizaje profundo para reconocimiento de emociones. FER2013 es notable por su tamaño y la diversidad de imágenes en diferentes condiciones. La composición de este conjunto de datos es la siguiente:

- 35,887 imágenes en escala de grises de 48x48 píxeles.
- Imágenes extraídas de diversas fuentes web, lo que introduce variabilidad en condiciones de iluminación, poses y calidad.
- Se divide en conjuntos de entrenamiento, validación y prueba.

La base de datos FER2013 clasifica las emociones en siete categorías:

- Neutral, felicidad (positiva), tristeza (negativa), ira (negativa), sorpresa (positiva), disgusto (negativa), miedo (negativa).

FER2013 presenta un mayor grado de variabilidad y ruido en las imágenes que JAFFE, KDEF y CK+. Por lo tanto, esta base de datos es un desafío para el entrenamiento de modelos de reconocimiento de emociones en entornos no controlados. Sin embargo, para abordar los

desafíos de desbalance de clases y baja calidad de imágenes presentes en bases de datos comunes como FER2013 y gracias a que los autores Sukhrob Bobojanov et al [44] se generó una nueva base de datos balanceada mediante técnicas de aumento de datos y limpieza de imágenes con bajo contraste u oclusiones.

La Figura 3.5 muestra algunas de las imágenes pertenecientes a la base de datos FER2013.

Figura 3.5 Muestras de la base de datos FER2013[44].

3.2. Experimentación de reconocimiento de emociones

Esta sección presenta los experimentos realizados con las redes neuronales descritas en el Capítulo 1 con las bases de datos de JAFFE, CK+, KADEF, FER2013 y EMOTION-ITCH. El objetivo principal de esta fase experimental es identificar qué arquitecturas son más efectivas para extraer las características adecuadas para la tarea de clasificación de emociones.

Los experimentos se llevaron a cabo utilizando modelos tanto con pesos preentrenados como sin ellos, el primer experimento se llevó a cabo para determinar cuáles de las arquitecturas tenía mejor desempeño para el reconocimiento de emociones, mientras que el segundo experimento se utilizó para conocer identificar las características más relevantes para diseñar un modelo propio, ligero y específico para la tarea de clasificación de emociones, sin requerir necesariamente el uso de pesos preentrenados.

3.2.1. Entrenamiento de modelos utilizando pesos pre-entrenados

El uso de pesos obtenidos de una base de datos más grande permite aprovechar características previamente aprendidas de estas bases de datos, lo que resulta particularmente útil cuando se trabaja con bases limitadas. Este enfoque facilita una mejor generalización y acelera la convergencia del entrenamiento, reduciendo la probabilidad de sobreajuste.

Se ha optado por mostrar únicamente los resultados correspondientes a la base de datos JAFFE por razones de espacio y porque esta base permite ilustrar de manera clara las ventajas y limitaciones del uso de pesos preentrenados.

3.2.1.1. Resultados con JAFFE

Esta sección presenta los resultados obtenidos mediante pesos preentrenados sobre la base de datos JAFFE. Para ello, se emplearon tres arquitecturas ampliamente utilizadas: MobileNetV1, ShuffleNet y VGG16, todas adaptadas a la tarea de clasificación de expresiones faciales mediante la sustitución de sus capas superiores por capas densas personalizadas.

El entrenamiento de los modelos evaluados en esta etapa se realizó utilizando implementaciones preexistentes en *Keras/TensorFlow*, las cuales incluyen pesos preentrenados con la base de datos de ImageNet [45]. ImageNet es una base de datos que contiene más de 14 millones de imágenes etiquetadas y distribuidos en miles de categorías, las cuales se refieren a los objetos que contienen cada una de las imágenes. Estas imágenes se utilizan para preentrenar modelos de visión por computadora con el objetivo de aprender representaciones generales que luego puedan adaptarse a tareas específicas con conjuntos de datos más pequeños. A esta estrategia de aprovechamiento de conocimiento previamente adquirido se le conoce como transferencia de aprendizaje o *transfer learning* y *fine-tunning*, y ha demostrado mejorar significativamente el rendimiento y la velocidad de convergencia en tareas especializadas.

3.2.1.2. MobileNetV1

El modelo MobileNetV1 mostró un desempeño sólido al ser entrenado con la base de datos JAFFE. La precisión en entrenamiento alcanzó valores superiores al 95 %, mientras que en validación se estabilizó alrededor del 87 %, lo que indica una buena capacidad de generalización. La pérdida disminuyó rápidamente y se mantuvo baja durante el resto del entrenamiento, evidenciando una convergencia eficiente. La matriz de confusión revela un buen reconocimiento de la mayoría de las emociones, con aciertos perfectos en clases como HA, FE y DI. Las confusiones observadas entre clases como SA y SU, o entre neutral y HA, son coherentes con la similitud que se tiene en las expresiones faciales, especialmente considerando el tamaño reducido del conjunto de datos, los resultados pueden verse de manera gráfica en la Figura 3.6.

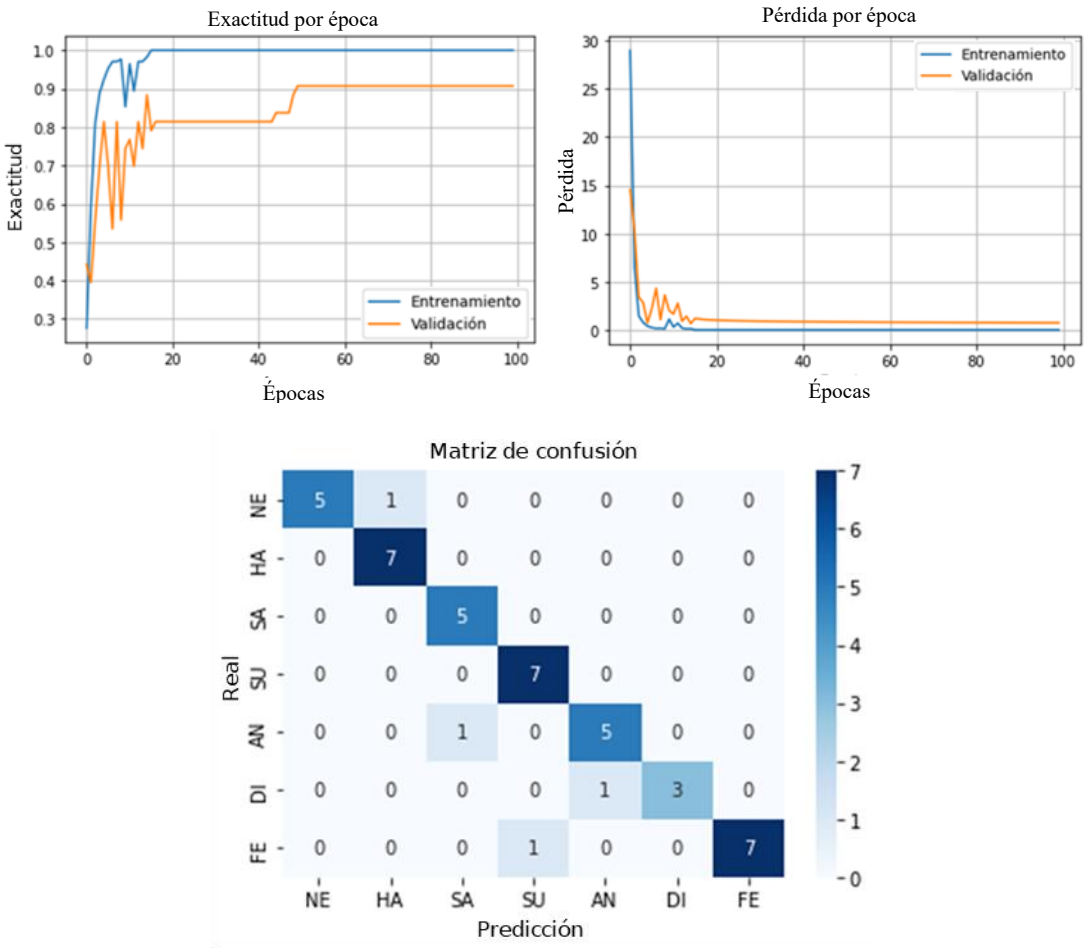


Figura 3.6 Resultados del modelo MobileNetV1 con JAFFE.

3.2.1.3. ShuffleNet

El modelo *ShuffleNet* alcanzó una precisión de validación cercana al 87 %, mostrando una curva de entrenamiento estable y sin signos claros de sobreajuste, aunque con una ligera diferencia entre las curvas de entrenamiento y validación. La pérdida se redujo progresivamente, con una convergencia adecuada en ambas fases. En la matriz de confusión, se observa una clasificación correcta en emociones como NE, SU, DI y SA, mientras que HA y FE presentan mayores confusiones. En particular, la clase FE se confundió con SA, SU y DI, lo que sugiere una menor diferenciación de las características faciales asociadas a esta emoción en el conjunto de entrenamiento, los resultados pueden verse de manera gráfica en la Figura 3.7.

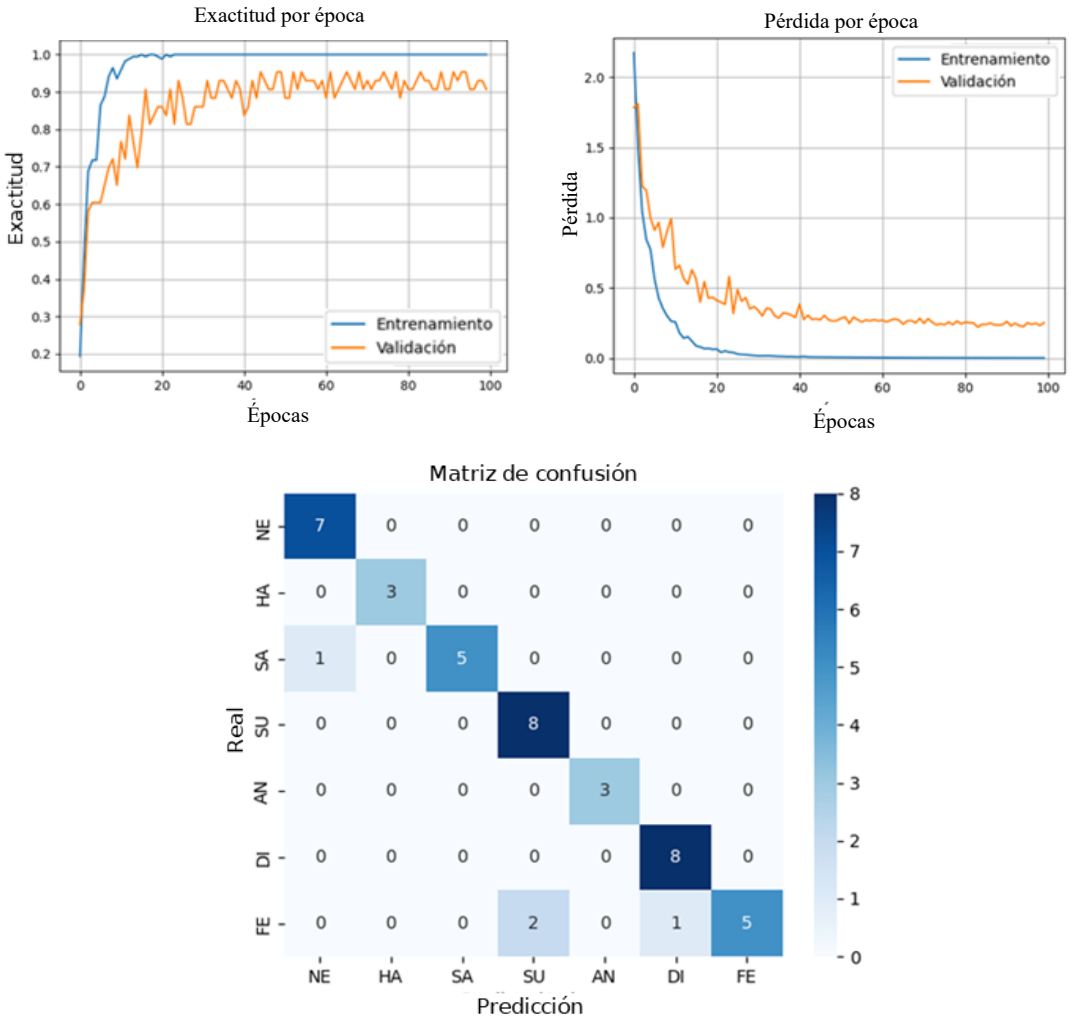


Figura 3.7 Resultados del modelo ShuffleNet con JAFFE.

3.2.1.4. VGG16

El modelo VGG16 logró una precisión de validación cercana al 90 %, con una curva de entrenamiento que muestra una rápida convergencia y valores estables a lo largo de las épocas. La pérdida disminuyó consistentemente tanto en entrenamiento como en validación, alcanzando valores bajos que indican una optimización adecuada. La matriz de confusión refleja un desempeño robusto en emociones como neutral, SU y DI, con clasificaciones correctas en la mayoría de las muestras. Sin embargo, se observan confusiones en la clase FE, que fue identificada erróneamente como SA, SU y DI, al igual que en los modelos anteriores, los resultados pueden verse de manera gráfica en la Figura 3.8.

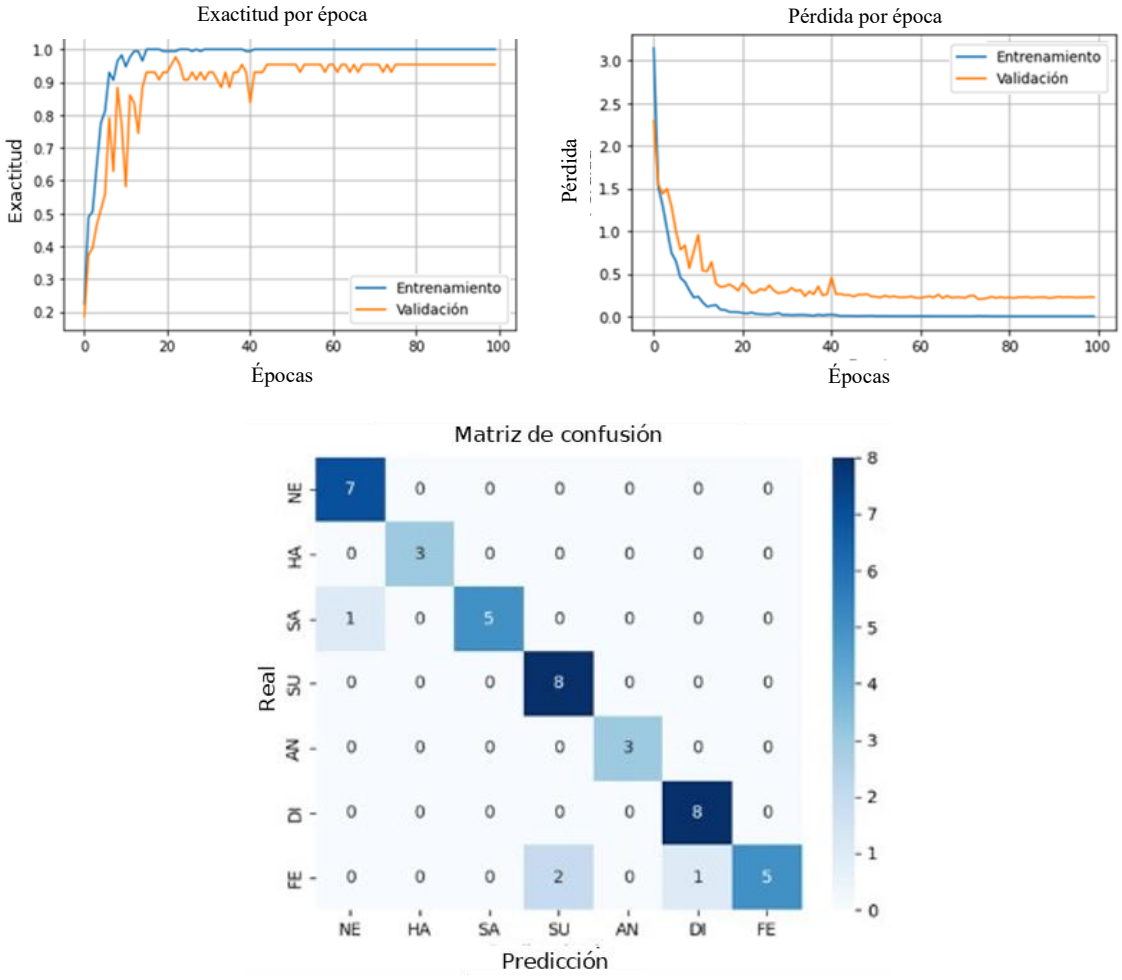


Figura 3.8 Resultados del modelo VGG16 con JAFFE.

3.2.1.5. Conclusión del experimento

Los resultados obtenidos al aplicar transferencia de aprendizaje con pesos de ImageNet sobre la base de datos JAFFE permiten concluir que el uso de modelos conocidos como MobileNetV1, ShuffleNet y VGG16 resulta eficaz para la clasificación de expresiones faciales, incluso en escenarios con bases de datos limitadas.

Específicamente, MobileNetV1 se destacó por su excelente capacidad de generalización, manteniendo una alta precisión de entrenamiento y validación, con especial acierto en emociones como felicidad (HA), miedo (FE) y disgusto (DI). ShuffleNet, a pesar de su estructura más ligera, mantuvo un rendimiento competitivo, aunque presentó mayores confusiones en clases con expresiones faciales similares, como el miedo. Por su parte, VGG16 alcanzó el mayor rendimiento en validación, evidenciando un comportamiento estable y preciso en la mayoría de las emociones, aunque replicó las dificultades comunes con la clase FE.

En conjunto, los tres modelos demostraron ser capaces de aprender representaciones discriminativas útiles, siendo la estrategia de *fine-tuning* una herramienta valiosa para mejorar el desempeño en bases de datos pequeñas. Sin embargo, también se identifican limitaciones comunes en la distinción de emociones con rasgos faciales sutiles, lo cual sugiere que, para mejorar la precisión general, podrían explorarse estrategias complementarias como el uso de mecanismos atención especializada, un resumen de este experimento puede verse en la Tabla 3.1.

Tabla 3.1 Resumen de resultados con pesos preentrenados sobre la base de datos JAFFE

Modelo	**Precisión en entrenamiento**	**Precisión en validación**	**Principales aciertos**	**Principales confusiones**
MobileNetV1	>95 %	~87 %	HA, FE, DI	SA↔SU, NE↔HA
ShuffleNet	>92 %	~87 %	NE, SU, DI, SA	FE↔SA, SU, DI
VGG16	>95 %	~90 %	NE, SU, DI	FE↔SA, SU, DI

3.2.2. Entrenamiento de modelos sin pesos preentrenados

La siguiente etapa de los experimentos consistió en el desarrollo y evaluación de modelos sin el uso de pesos aleatorios, con el fin de compararlos directamente con aquellos modelos entrenados utilizando pesos de la base de datos ImageNet. Esta comparación se llevó a cabo con el objetivo de analizar en profundidad la estructura del modelo y evaluar su impacto en el rendimiento del aprendizaje en la tarea de clasificación de emociones. Los modelos elegidos para este análisis fueron los siguientes:

- MobileNetV1
- VGG16
- DenseNet121 (versión simplificada)

Estos modelos fueron seleccionados por su balance entre precisión, eficiencia computacional y capacidad de extracción de características en la clasificación de emociones.

3.2.2.1. MobileNetV1

MobileNetV1 mostró una curva de entrenamiento con mejora progresiva en precisión y reducción sostenida de la pérdida, lo que indica una optimización adecuada del modelo en ese conjunto. Sin embargo, durante la validación, la precisión se mantuvo en un rango bajo (entre 0.2 y 0.5) y la pérdida tendió a incrementarse con las épocas, lo que evidencia un problema de sobreajuste, es decir, el modelo aprendió a representar muy bien los datos de entrenamiento, pero no logró generalizar correctamente a los datos no vistos.

Esta limitación puede deberse a que MobileNet, al ser una red ligera, tiene dificultades para capturar la complejidad de las expresiones faciales en JAFFE, donde las diferencias entre emociones son sutiles, dichos resultados pueden verse en la Figura 3.9.

Modelo sin pesos preentrenados

Modelo con pesos preentrenados

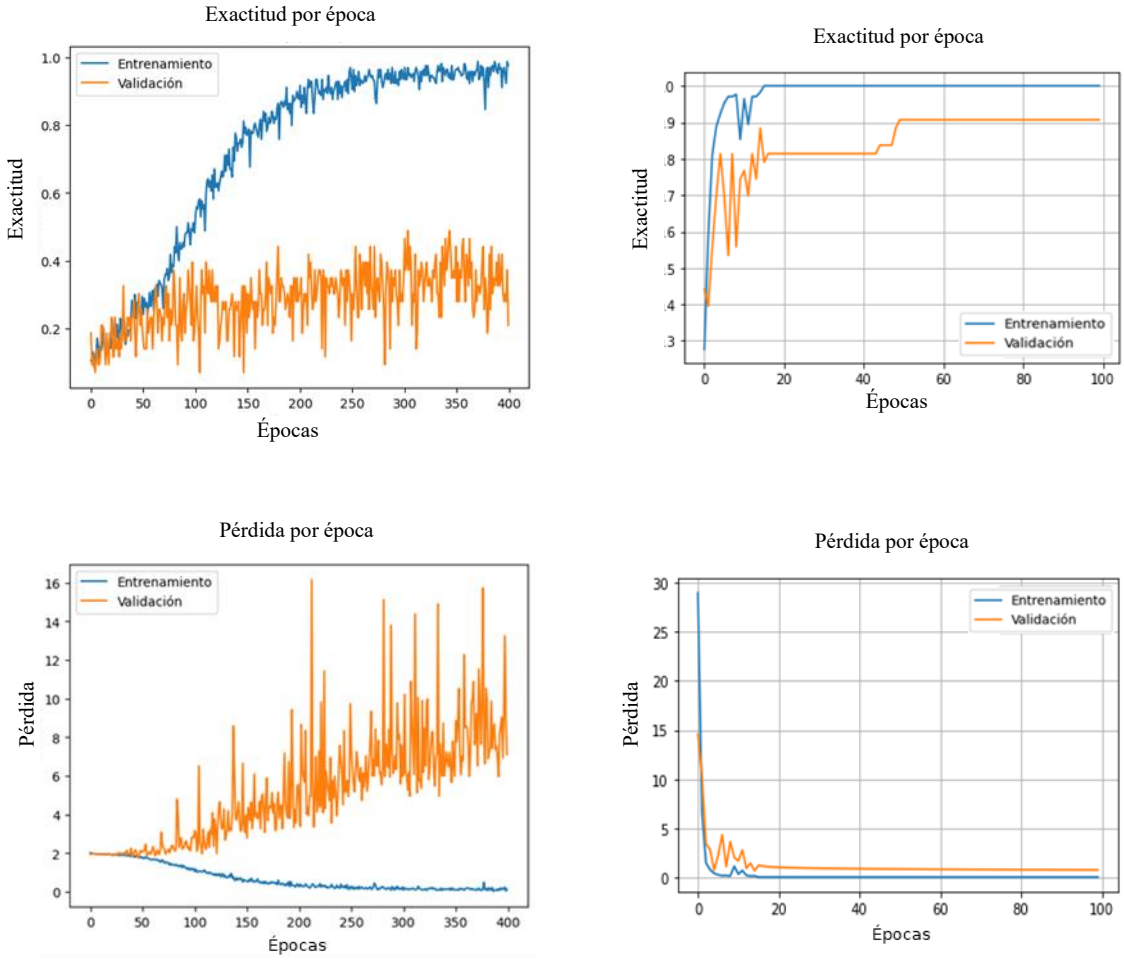


Figura 3.9 Comparación de los resultados del modelo MobileNetV1 con JAFFE.

Por otro lado, MobileNetV1 en CK+ muestra un desempeño positivo en entrenamiento, alcanzando cerca del 90 % de precisión. Sin embargo, en validación, la precisión fluctúa entre 0.5 y 0.9, indicando dificultades de generalización y posible sobreajuste. Mientras que la pérdida en entrenamiento disminuye de manera constante, la pérdida en validación presenta variaciones significativas, lo que sugiere sensibilidad a ciertas muestras.

La matriz de confusión muestra un rendimiento desigual entre clases, con buena clasificación en "AN" y "SA", pero dificultades en "HA" y "NE". Además, se observan errores recurrentes en "FE" y "SU", reflejando problemas para diferenciar emociones similares.

Este comportamiento puede deberse a la naturaleza ligera de MobileNet, diseñada para eficiencia en dispositivos con recursos limitados. Sin embargo, en una base de datos pequeña como CK+, la distribución de las clases y el número de muestras pueden no ser suficientes para garantizar estabilidad en la clasificación, lo que explica las fluctuaciones observadas en validación, dichos resultados pueden verse en la Figura 3.10.

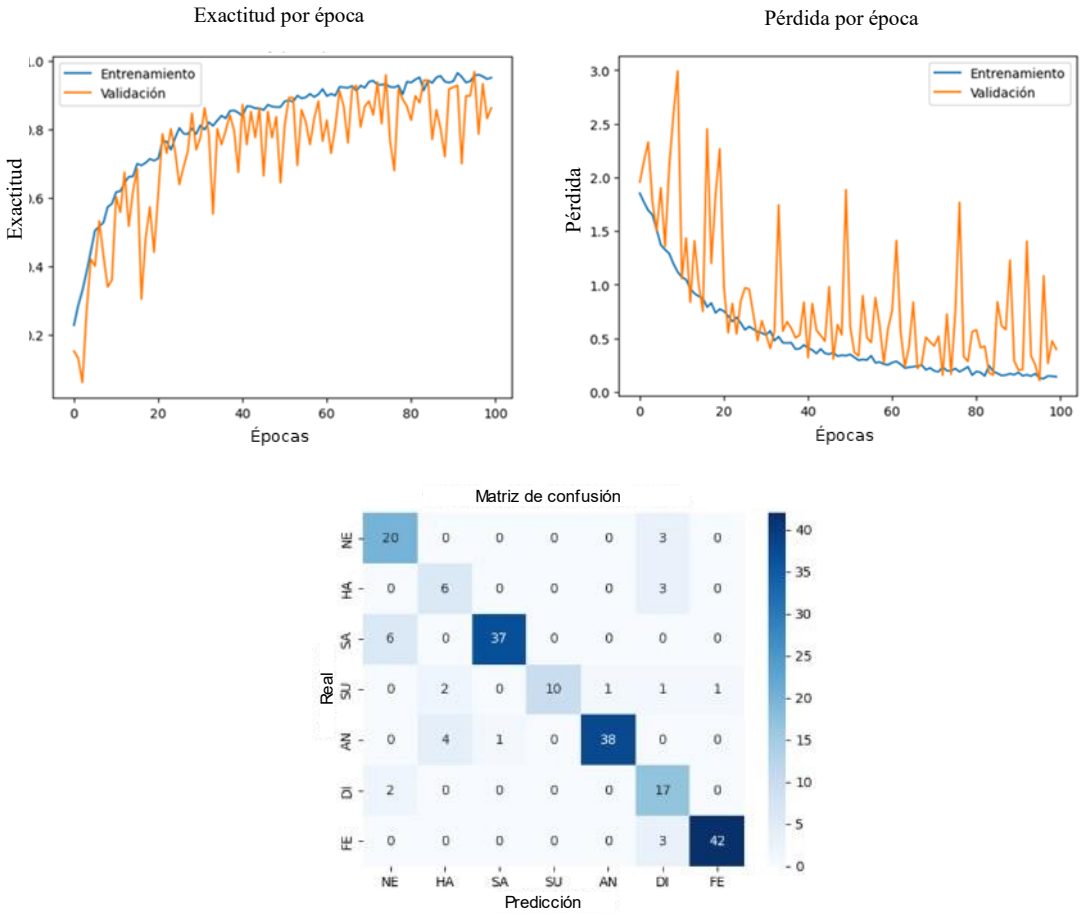


Figura 3.10 Valores propios del modelo sin contexto MobileNet con CK+.

3.2.2.2. VGG16

El análisis de VGG16 en JAFFE muestra un rendimiento sólido y consistente, alcanzando alta precisión en entrenamiento y validación desde las primeras épocas. Aunque presenta leves caídas en precisión y picos de pérdida en validación alrededor de la época 300, el modelo se recupera rápidamente, indicando estabilidad.

Este desempeño se debe a la arquitectura profunda de VGG16, que extrae patrones complejos sin sobre ajustarse, permitiéndole generalizar mejor que modelos más ligeros en la tarea de reconocimiento de expresiones faciales, dichos resultados pueden verse en la Figura 3.11.

Modelo sin pesos preentrenados

Modelo con pesos preentrenados

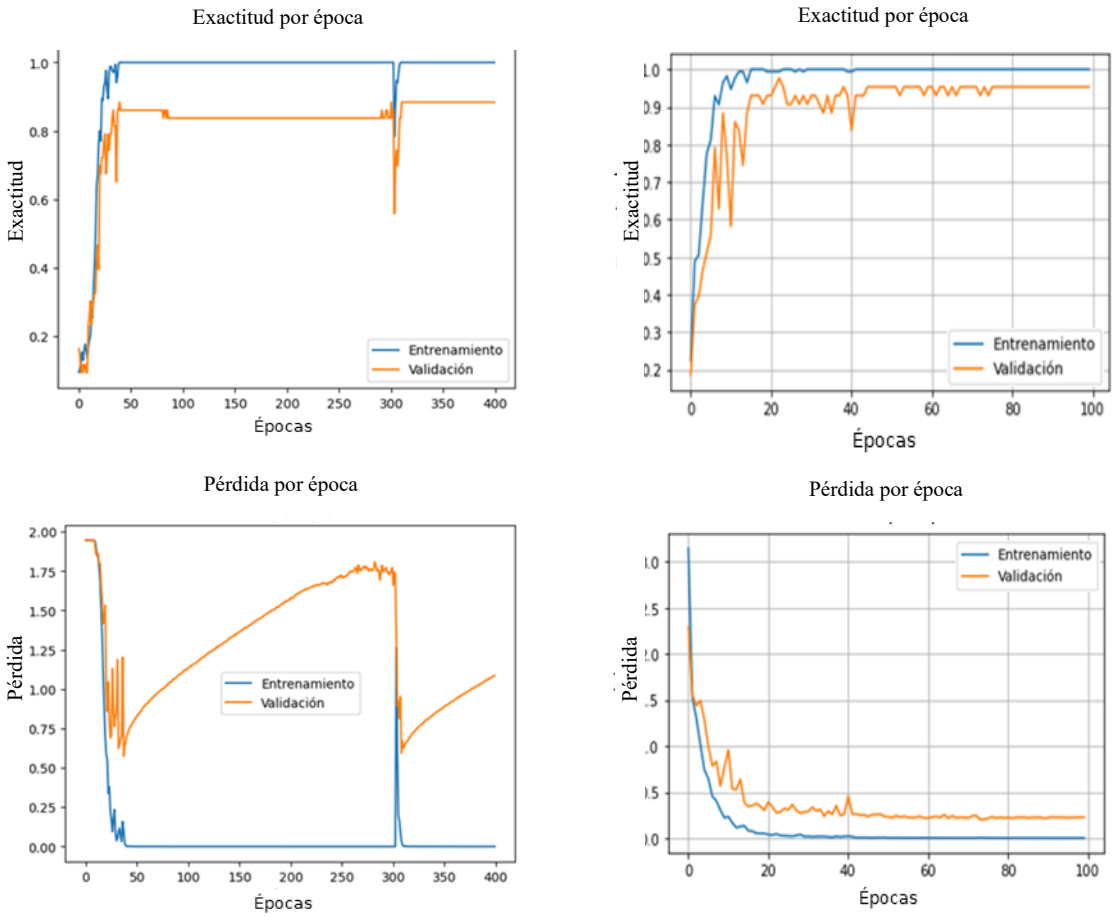


Figura 3.11 Comparación de los resultados del modelo VGG16 con JAFFE.

Por otro lado, el análisis de VGG16 en CK+ muestra un desempeño excepcional, alcanzando casi el 100 % de precisión en entrenamiento y validación desde las primeras épocas. La estabilidad en las curvas de precisión y pérdida indica ausencia de sobreajuste, con una rápida convergencia y solo leves fluctuaciones iniciales.

La matriz de confusión confirma una clasificación precisa en casi todas las clases. Este rendimiento óptimo se atribuye a la capacidad de VGG16 para capturar variaciones sutiles en las expresiones emocionales y a la calidad de CK+, que facilita una buena generalización. Además, la estabilidad en las métricas sugiere que los hiperparámetros están bien ajustados, asegurando un entrenamiento eficiente y robusto, dichos resultados pueden verse en la Figura 3.12.

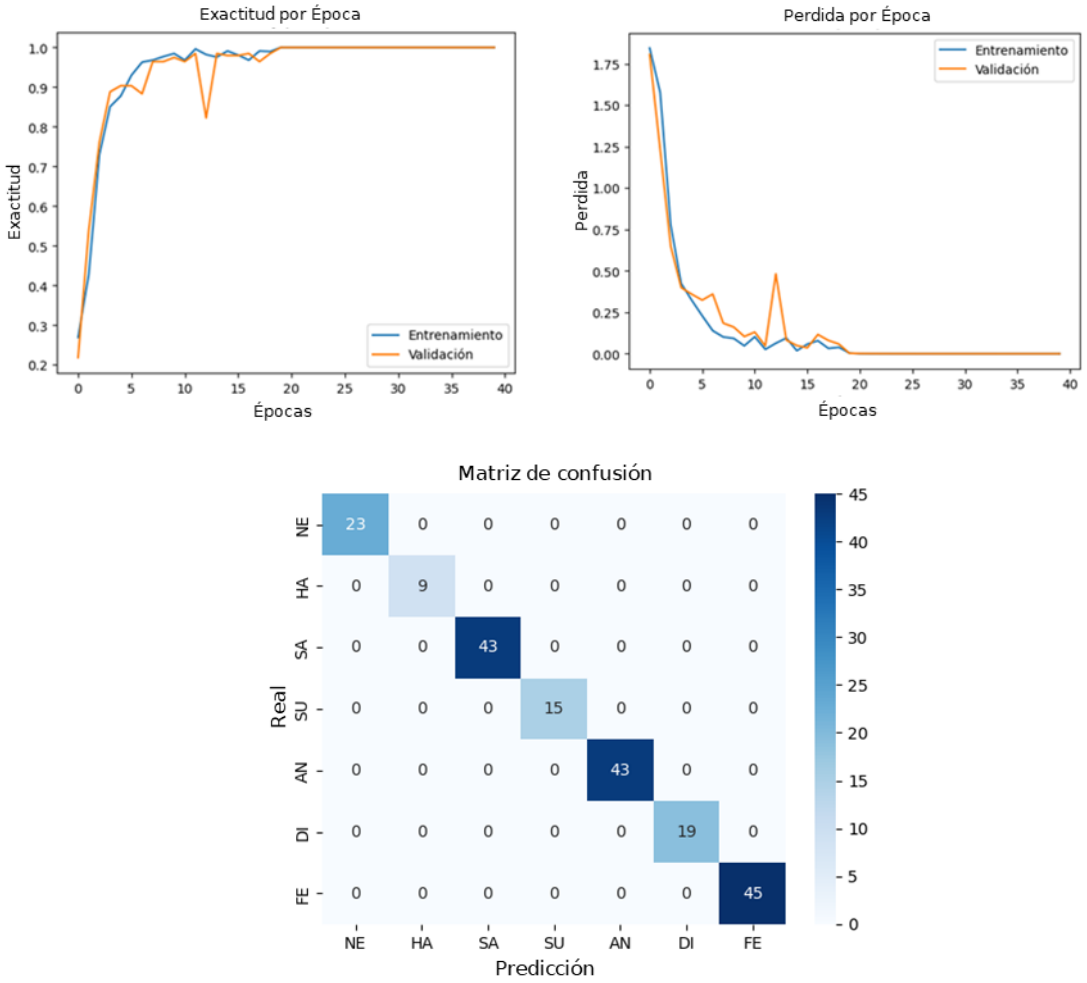


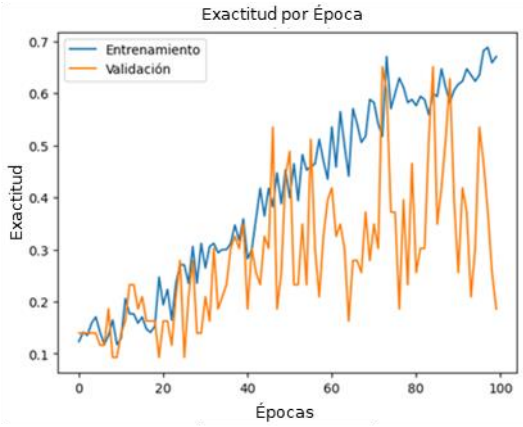
Figura 3.12 Comparación de los resultados del modelo VGG16 con CK+.

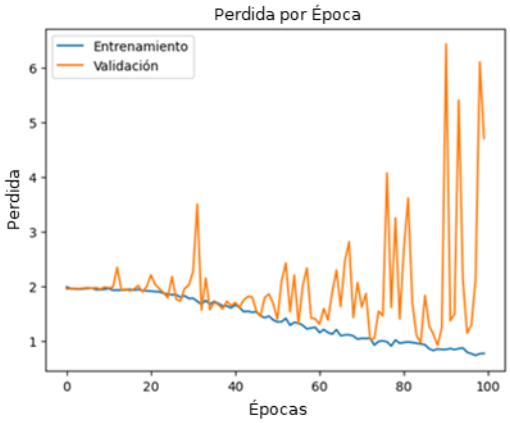
3.2.2.3. DenseNet121

El análisis de DenseNet en JAFFE muestra problemas de generalización y precisión. Aunque el modelo mejora en entrenamiento, la precisión en validación es inestable y no supera consistentemente el 0.6, con una pérdida elevada hasta las últimas épocas, lo que sugiere sobreajuste o dificultad para identificar patrones en validación.

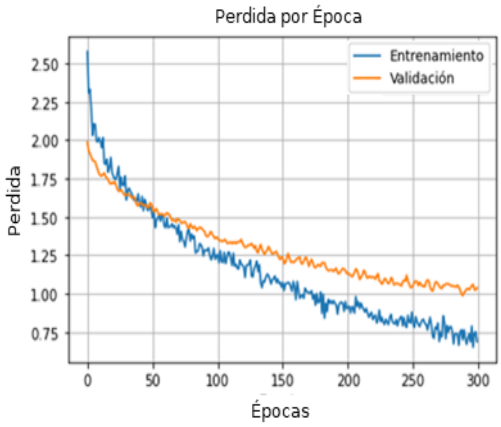
Estos problemas pueden deberse a la complejidad de DenseNet, que podría sobreajustarse en una base de datos pequeña como JAFFE. Además, el desbalance de clases puede haber generado sesgo en las predicciones. Finalmente, la alta conectividad en este modelo, útil en grandes conjuntos de datos, podría estar generando redundancia en los patrones aprendidos debido a la limitada cantidad de datos disponibles, dichos resultados pueden verse en la Figura 3.13.

Modelo sin pesos prenetrenados





Modelo con pesos preentrenados



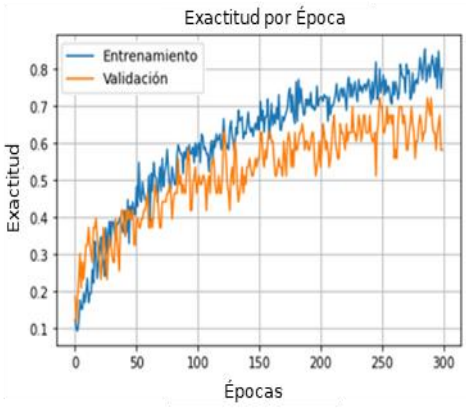


Figura 3.13 Comparación de los resultados del modelo DenseNet121 con JAFFE.

Por otro lado, DenseNet en CK+ muestra problemas graves de generalización. Aunque alcanza casi el 100 % de precisión en entrenamiento, la precisión en validación se mantiene baja (~0.2), con fluctuaciones mínimas, lo que indica un sobreajuste extremo. La pérdida en validación es errática y presenta valores elevados en varias épocas, confirmando la incapacidad del modelo para generalizar.

La matriz de confusión refleja un rendimiento deficiente, con predicciones sesgadas hacia ciertas clases como "FE" y "AN", mientras que ignora otras como "NE" y "HA".

Este comportamiento puede deberse a la complejidad de DenseNet, optimizada para grandes volúmenes de datos, lo que provoca sobreajuste en una base pequeña como CK+, dichos resultados pueden verse en la Figuras 3.14.

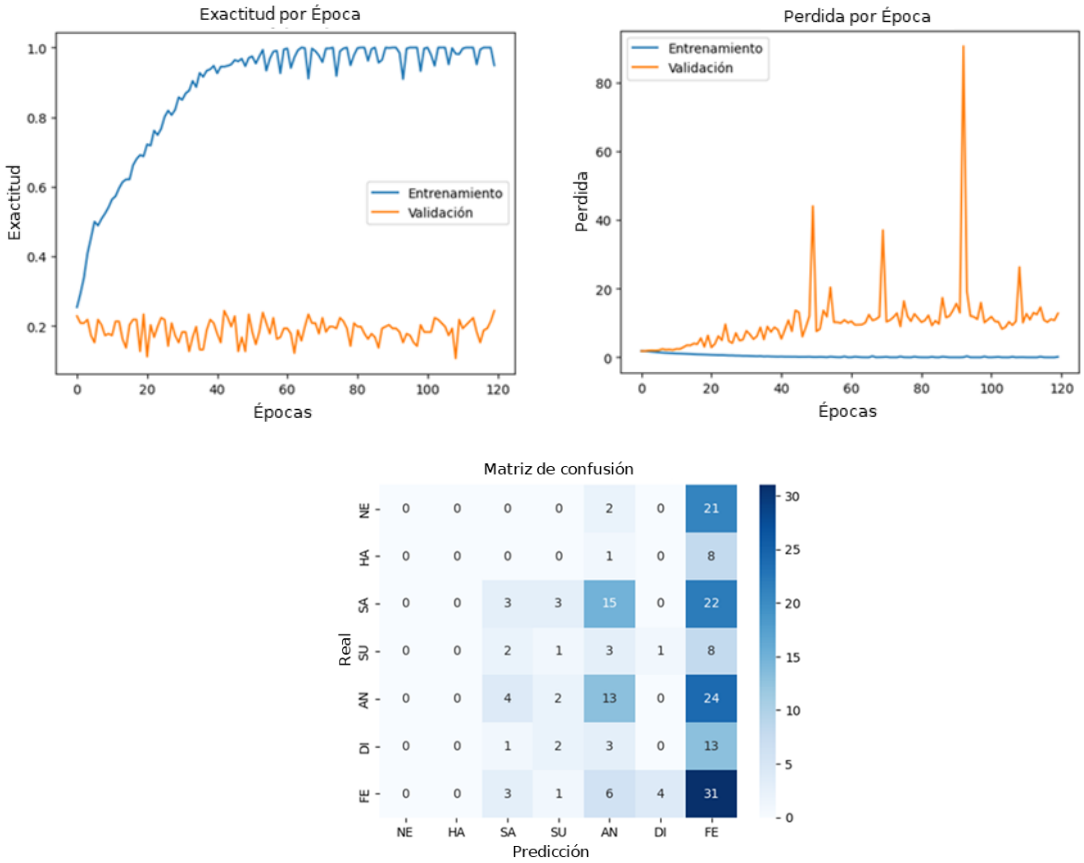


Figura 3.14 Valores propios del modelo sin contexto DenseNet con CK+.

3.2.2.4. Conclusión del experimento

El entrenamiento desde cero evidenció que la ausencia de pesos preentrenados con alguna base de datos como ImageNet afecta significativamente la capacidad de generalización, especialmente en bases de datos pequeñas.

MobileNetV1 y DenseNet121 presentaron sobreajuste y bajo rendimiento en validación, destacando la necesidad de preentrenamiento o estrategias adicionales para mejorar su desempeño. En contraste, VGG16 mantuvo una alta precisión y estabilidad en ambos conjuntos, demostrando ser la arquitectura más robusta sin necesidad de transferencia previa, esto tiene sentido debido a su estructura profunda le permitió extraer patrones complejos sin caer en el sobreajuste, destacándose como la arquitectura más robusta en este escenario.

Estos resultados confirman que la elección del modelo y su complejidad deben estar alineadas con el tamaño y calidad del conjunto de datos, y que el preentrenamiento sigue siendo una herramienta clave en escenarios con datos limitados, un resumen de los resultados puede verse en la Tabla 3.2.

3.3. Discusión de los experimentos

Antes de proponer una arquitectura propia, se llevó a cabo una serie de experimentos comparativos con MobileNetV1, ShuffleNet, VGG16 y DenseNet121 sobre las bases de datos JAFFE, CK+, KDEF y FER2013. El propósito fue doble; primeramente, identificar qué familias de redes extraen con mayor eficacia los patrones faciales relevantes cuando el volumen de datos es limitado y también saber la relación entre exactitud y coste computacional que impone cada arquitectura, a fin de guiar el diseño de un modelo ligero pensado para sistemas embebidos.

Para los modelos con pesos entrenados se reutilizaron pesos iniciales entrenados en ImageNet, remplazando las capas superiores por clasificadores densos adaptados a siete emociones. Los resultados evidencian que las redes profundas (VGG16) capturan

características más discriminativas, pero las arquitecturas eficientes (MobileNetV1, ShuffleNet) ofrecen un balance atractivo para hardware restringido.

Tabla 3.2 Resumen de resultados sin pesos aleatorios.

Modelo	**Base de datos**	**Precisión en entrenamiento**	**Precisión en validación**	**Principales observaciones**
MobileNetV1	JAFFE	Alta (>90 %)	Baja (0.2–0.5)	Buen aprendizaje en entrenamiento, pero severo sobreajuste; dificultad para generalizar debido a arquitectura ligera.
MobileNetV1	CK+	Alta (~90 %)	Inestable (0.5–0.9)	Buen inicio, pero sensibilidad a muestras causa fluctuaciones; errores en emociones similares.
VGG16	JAFFE	Alta (>90 %)	Alta (~90 %)	Estable con ligeras caídas puntuales; buena generalización gracias a arquitectura profunda.
VGG16	CK+	Muy alta (~100 %)	Muy alta (~100 %)	Excelente rendimiento, estabilidad y generalización; clasificaciones casi perfectas.
DenseNet121	JAFFE	Alta	Irregular (<0.6)	Sobreajuste y pérdida alta; posible redundancia de conexiones por bajo volumen de datos.
DenseNet121	CK+	Muy alta (~100 %)	Muy baja (~0.2)	Sobreajuste extremo; modelo no logra generalizar; pérdida errática y predicciones sesgadas.

Por otro lado, en los modelos entrenados sin pesos entrenados la comparación confirma que el uso de pesos preentrenados es útil cuando se dispone de pocos datos y que las arquitecturas ligeras requieren mecanismos adicionales (atención, regularización) para sostener la precisión.

Con esto podemos concluir que VGG16 ofrece la mejor precisión, pero su elevado número de parámetros y FLOPs la descartan para ejecución en tiempo real en sistemas de bajos recursos. Pero además de las arquitecturas grandes, MobileNetV1 y ShuffleNet mantienen un costo bajo y muestran buen desempeño con transferencia de aprendizaje, lo que sugiere adoptar sus bloques de convolución separable y *channel shuffle* como base.

Pero las confusiones recurrentes entre expresiones faciales similares indican que la simple reducción de parámetros no basta; se requieren módulos de atención y estrategias de regularización (L2, dropout) para mejorar la discriminación.

Estas observaciones motivaron el desarrollo de LiExNet, un modelo que combina la eficiencia de MobileNetV1 con bloques de Atención Eficiente por Canal (ECA) y técnicas de regularización, reduciendo la profundidad total sin sacrificar capacidad de representación.

4. LITTLE EXPRESSION NET - LiExNet

LiExNet es un modelo que fue desarrollado combinando elementos de la arquitectura de MobileNetV1 en su bloque de extracción de características, pero reduciendo su número de capas convolucionales e incluyendo capas ECA, capas de regularización L2 y *dropout*. Su diseño permite mantener un bajo número de parámetros sin comprometer la capacidad de representación, haciéndolo ideal para sistemas embebidos y dispositivos con recursos limitados. La arquitectura del modelo puede verse en la Figura 4.1.

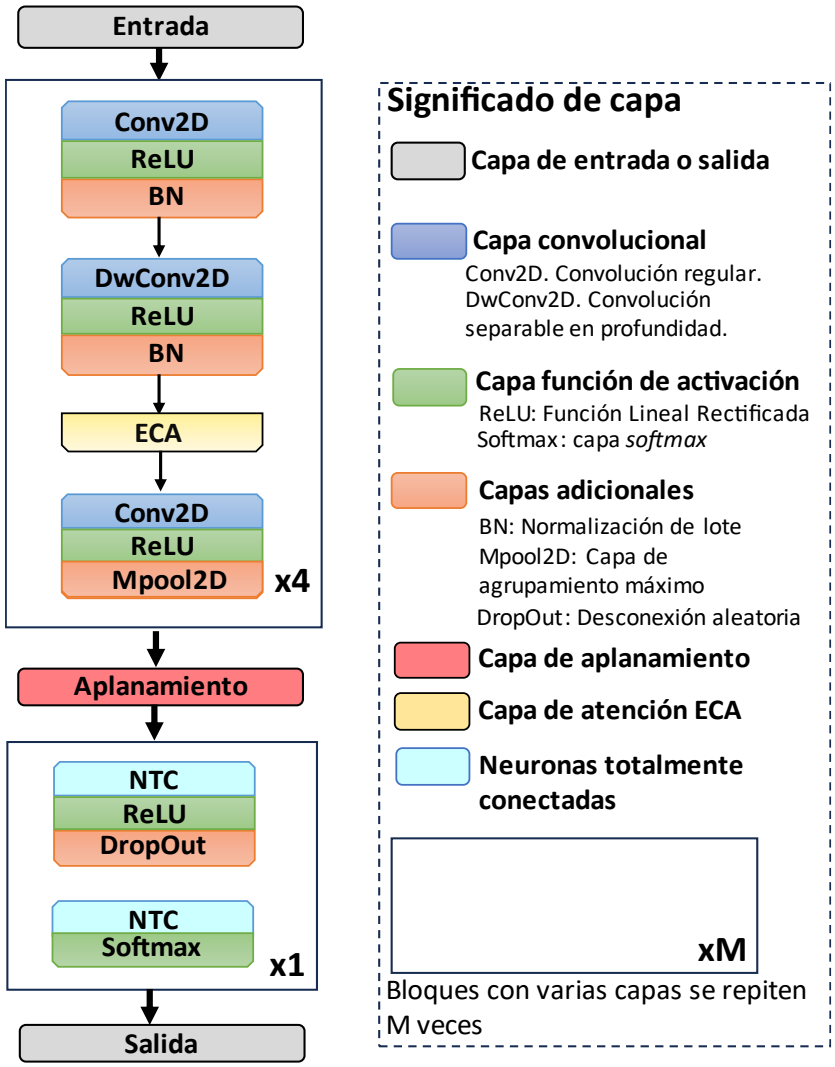


Figura 4.1 Arquitectura del modelo LiExNet

La entrada del sistema es una imagen facial proveniente de una base de datos o capturada en tiempo real. La extracción de características se realiza mediante cuatro bloques que combinan capas convolucionales separables en profundidad, funciones de activaciones de unidad lineal rectificada (ReLU por sus siglas en inglés), canales de atención eficiente, y operaciones de normalización o agrupación. El resultado es un conjunto de características que se convierte en un vector mediante una capa de aplanamiento. Este vector es procesado por el bloque de clasificación para identificar la emoción expresada. El resultado final puede emplearse en tareas de control o comunicación. En las siguientes secciones se describen los componentes de LiExNet.

4.1. Entrada

La red recibe una imagen facial a color $I(m,n)^{RGB}$ en formato entero de 8 bits de una expresión facial que puede provenir de una de las bases de datos que se utilizaron para este proyecto de investigación que son FER2013, KDEF, CK+ o JAFFE, o bien puede provenir de una captura en tiempo real. Cada canal se pasa a formato flotante y se normaliza dividiendo por 255, con lo que se crea un tensor

$$X_r(m,n) = \frac{I(\mathrm{m},\mathrm{n})^{RGB}}{255}, \qquad r \in \{R,G,B\}, \tag{4.1}$$

cuyos valores quedan en el rango [0,1]. Para preservar la relación espacial entre los rasgos faciales y aprovechar convoluciones profundas entrenadas sobre entradas cuadradas, el tensor se redimensiona a 224x224x3.

4.2. Extracción de características

Antes de exponer la secuencia interna de los cuatro bloques convolucionales, se definen con los operadores fundamentales que la componen.

Convolución

Una convolución se entiende de manera que, sea una entrada tridimensional representada por el tensor $X \in R^{H \times W \times C_{\text{in}}}$ donde H y W representan el alto y ancho de la imagen o matriz de características de entrada respectivamente y C_{in} es el número de canales.

Sea un conjunto de filtros $K \in R^{k_H \times k_W \times C_{\text{in}} \times C_{\text{out}}}$ donde k_H y k_W es el tamaño del filtro y C_{out} es el número de filtros o canales de salida.

La salida de la operación convolucional se define como:

$$Y_{i,j,c} = \sum_{m=0}^{k_H-1} \sum_{n=0}^{k_W-1} \sum_{d=0}^{C_{\text{in}}-1} X_{i+m,j+n,d} \cdot K_{m,n,d,c} + b_c \;\; \text{ó} \;\; Y_c = X * K_c + b_c \qquad (4.2)$$

Donde $Y_{i,j,c}$ representa el valor en la posición (i, j) del canal c de la salida, por otro lado $K_{m,n,d,c}$ es el valor del filtro aplicado en la posición (m, n) del canal de entrada d hacia el canal de salida c, además se le agrega un sesgo b_c correspondiente al canal c.

Capa de agrupamiento

Dentro del proceso de extracción de características, se emplea también la capa de agrupamiento máximo, conocida en inglés como *max pooling*, cuya función principal es reducir la dimensionalidad espacial de los mapas de activación sin añadir nuevos parámetros al modelo. Esta operación permite conservar la información más significativa de cada región local, eliminando al mismo tiempo valores redundantes o poco relevantes, lo cual mejora la eficiencia computacional y refuerza la capacidad del modelo para generalizar.

Matemáticamente, dada una entrada $\boldsymbol{A} \in R^{H \times W \times C}$, donde H y W representan las dimensiones espaciales del mapa de características y C es el número de canales, la operación de *max pooling* sobre una ventana de tamaño $p \times p$ con paso s se define como:

$$M_{i,j,c} = \max_{\substack{0 \le m < p \\ 0 \le n < p}} A_{i \cdot s+m,\ j \cdot s+n,\ c} \tag{4.3}$$

Donde $M_{i,j,c}$ representa el valor de salida en la posición *(i,j)* del canal c, es importante saber que se selecciona el valor máximo dentro de la ventana local de tamaño $p \times p$ y que el desplazamiento entre ventanas viene dado por el *stride* s.

Función de activación

La función de activación ReLU, introduce no linealidad en el modelo y permite que las redes profundas aprendan representaciones más complejas. Esta función está definida matemáticamente como:

$$\mathrm{ReLU}(x) = \max(0, x) \tag{4.4}$$

Donde sí la entrada x es positiva, la salida es igual a x, pero si la entrada es negativa, la salida es cero.

Convolución separable en profundidad

Las convoluciones integran la información espacial y de canales en una sola operación, aumentando la expresividad de la red, pero también el número de parámetros y el costo computacional. Para equilibrar ambos aspectos, el modelo emplea convoluciones separables en profundidad, las cuales reducen de forma notable los parámetros sin mermar su poder de extracción de características.

Dado un conjunto de filtros $\boldsymbol{K} \in R^{k_H \times k_W \times C_{\text{in}}}$, donde se utiliza un filtro independiente por canal, la salida se calcula como:

$$Z_{i,j,d} = \sum_{m=0}^{k_H-1} \sum_{n=0}^{k_W-1} X_{i+m,j+n,d} \cdot K_{m,n,d} + b_d \quad \text{ó} \quad Z_d = X_d * K_d + b_d \tag{4.5}$$

Donde $Z_{i,j,d}$ representa el valor en la posición (i, j) del canal d de la salida. La operación se realiza de forma independiente por canal, por lo que no hay mezcla entre los distintos canales de entrada y también se agrega un sesgo b_d correspondiente el canal d.

Para obtener una salida con múltiples canales y permitir la combinación de la información aprendida por canal, se utiliza posteriormente una convolución punto a punto (*pointwise convolution*) de tipo 1×1, descrita por:

$$Y_{i,j,c} = \sum_{d=0}^{C_{\text{in}}-1} Z_d \cdot P_{d,c} + b_c \tag{4.6}$$

Donde $P_{d,c} \in R^{C_{\text{in}} \times C_{\text{out}}}$ son los pesos de la convolución 1×1, aquí se realiza una combinación lineal de todos los canales de la salida intermedia Z para obtener el canal de salida c.

Estas definiciones se utilizan en los cuatro bloques para obtener características faciales.

4.2.1. Convolución estándar + ReLU + max-pooling

$X_r(m, n)$ se considera la capa $l = 1$ y es procesada mediante filtros 3×3 con los cuatro bloques convolucionales con la siguiente secuencia. Primero se obtiene la convolución + RELU dados por:

$$C_{l,r}(m, n) = ReLU\big(W_l * Y_{l-1,r}(m, n) + \beta_l\big), \qquad l \in \{1,5,9,13\}, \ \ r = 1, \dots, R \tag{4.7}$$

Después se aplica *max-pooling* de 2×2 dado para reducir la resolución a la mitad y mitigando el sobreajuste, lo cual está dado por:

$$Y_{l,r} = g[C_{l,r}]. \tag{4.8}$$

4.2.2. Convolución depth-wise + ReLU + max-pooling

Cada canal se filtra de manera independiente, bajando drásticamente el número de multiplicaciones mediante una convolución clásica de la misma dimensión:

$$C_{l,r}(m,n) = ReLu\big(W_{l,r} * Y_{l,r}(m,n) + \beta_{l,r}\big),\ l \in \{2,6,10,14\}. \tag{4.9}$$

El resultado vuelve a pasar por *max-pooling* 2×2 que se puede ver en la ecuación 4.8.

4.2.3. ECA

Para que la red se centre en canales realmente discriminativos se emplea ECA, cuya lógica se resume en tres pasos. Primero se obtiene el *global average pooling* condensa la respuesta espacial de cada canal:

$$s_r = \frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N} Y_{l-1,r}(m,n). \tag{4.10}$$

Luego, se realiza una convolución 1-D, cuyo tamaño de núcleo se ajusta en función del número de canales **M**, lo que modela dependencias inter-canal definidas por:

$$k = \psi(M) = \left|\frac{\log_2 M}{\gamma} + b\right|_{\text{odd}}, \quad \gamma = 2,\ b = 1 \tag{4.11}$$

$$z = \sigma(W_{\text{ECA}} * s), \tag{4.12}$$

donde σ es la sigmoide. Finalmente, cada mapa de activación se recalibra con el peso z_r:

$$\widetilde{Y_{l,r}} = z_r\, C_{l,r}. \tag{4.13}$$

ECA incrementa la exactitud hasta 1.2 puntos porcentuales en sin aumentar los demasiado los FLOPS y con menos de un kilobyte de parámetros.

4.2.4. Convolución estándar + ReLU + *max-pooling*

Una segunda convolución tradicional dada por la Ecuación 4.2, donde $l \in \{4,8,12,16\}$ con el objetivo de aumentar el número de características extraídas antes de su transformación en vector.

4.3. Transformación a vector

El mapa tridimensional de activaciones de tamaño 8x8x96 se transforma para crear un vector que conserve la información aprendida por todos los filtros:

$$Z(k) = \widetilde{Y_{16,r}}(m,n), \qquad k = m \cdot n \cdot r. \tag{4.14}$$

Este vector denso de 64 elementos actúa como una representación compacta de la expresión facial y alimenta el clasificador denso que sigue.

4.4. Clasificación.

El bloque de decisión consta de dos capas totalmente conectadas (FCN), estas funcionan de la siguiente manera:

4.4.1. Capa densa + ReLU + *dropout*

$Z(k)$ es la entrada de 256 neuronas con modelo FCN y se aplica *dropout* al 30 %, reduciendo el sobreajuste Esto se puede expresar de la siguiente manera:

$$h_r = ReLU\big(W_{1,r}Z + \beta_{1,r}\big), \qquad r = 1, \dots, 256. \tag{4.15}$$

4.4.2. Capa densa de salida + *softmax*

La segunda capa totalmente conectada transforma el vector oculto $h \in R^{256}$ en un vector de *logits* $z \in R^{7}$, uno por cada emoción. De esta manera, esta capa se define como:

$$z_i = W_{2,i}\, h + \beta_{2,i}, \qquad i = 1, \dots, 7. \tag{4.16}$$

Dado que LiExNet categoriza emociones por medio de clasificación multiclase (siete emociones), se aplica la función *softmax* para convertir dichos *logits* en una distribución de probabilidad. La función *softmax* se define como:

$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{7} e^{z_j}} \quad \text{para } i = 1,2, \dots, 7. \tag{4.17}$$

Donde (z_i) es la activación del nodo ***i*** antes del *softmax*, así 1 representa la probabilidad de que la entrada pertenezca a la emoción ***i***.

Finalmente, se selecciona como salida la clase correspondiente al índice de mayor probabilidad de la siguiente forma:

$$\hat{y} = \arg\max_{i} \ \text{softmax}(z_i) \tag{4.18}$$

Este valor $\hat{y} \in \{0,1,\dots,6\}$ representa la emoción identificada a partir de la imagen de entrada.

4.5. Fine-tunning para detección de estados afectivos en tiempo real

A partir de la experiencia obtenida con la red LiExNet, previamente entrenada con bases de datos enfocadas en siete emociones, se propone ahora utilizar el modelo para una tarea de clasificación binaria empleando la base de datos EMOTION-ITCH.

$$Y = \{Negativo, Positivo\} \tag{4.19}$$

Se utiliza un subconjunto de 260 imágenes que se redimensionaron a 224x224x3 debido a que la entrada debe de corresponder con el modelo LiExNet, estas muestras corresponden a las etiquetas de positivas y negativas siendo estas 130 imágenes para cada clase. Además, se tiene una lista que muestra que sujetos trabajan de manera activa, el interés práctico de esta lista era reforzar la detección de expresiones negativas en sujetos que se encontraban trabajando.

Para lograr que esta lista de sujetos tenga relevancia para el modelo en la clasificación se optó por un esquema de muestreo por pesos:

$$w_i = \begin{cases} w, Si\ la\ imagen\ "i"\ corresponde\ a\ un\ sujeto\ que\ trabaja \\ 1, en\ otro\ caso, \end{cases} \tag{4.20}$$

donde $w \in \{1, 1.5,2,3,4\}$ es el factor de peso que muestra si un sujeto trabaja, esta información se extrajo directamente de la lista utilizando *pandas* y *openpyxl.*

Durante el *fine-tuning* se optimizo la perdida mediante la entropía cruzada ponderada:

$$\mathcal{L}(\theta) = -\frac{1}{N}\sum_{i=1}^{N} w_i \sum_{c \in \mathcal{Y}} y_{i,c} \log\big(p_\theta(\, c \mid x_i\,)\big), \tag{4.21}$$

donde x_i representa la *i-ésima* imagen para obtener la probabilidad predicha por la red con parámetros θ, en una codificación *one-hot* de la clase real $y_{i,c} \in \{0,1\}$ utilizando los pesos definidos en (4.20).

La fase de *fine-tuning* siguió un esquema en tres etapas (3+10+40 épocas) con descongelado progresivo de capas y amortiguación de la tasa de aprendizaje mediante *Cosine Decay Restarts* (CDR) [46]. La idea central de usar CDR es no usar una tasa de aprendizaje fija, sino hacer que oscile de forma controlada: empezamos alto, bajamos suavemente hasta

casi 0 y, justo antes de que el optimizador deje de funcionar por caer en mínimos locales, volvemos a subir para salir del mínimo local y seguir explorando la superficie de pérdida.

Las métricas que se obtuvieron para cada modelo fueron:

- Exactitud

$$\text{Acc} = \frac{1}{|\mathcal{D}_{\text{val}}|} \sum_{(x,y)\in\mathcal{D}_{\text{val}}} 1[\hat{y}(x) = y] \tag{4.22}$$

- Recall de la clase negativa

$$\text{Recall}_{\text{Neg}} = \frac{\text{TP}_{\text{Neg}}}{\text{TP}_{\text{Neg}} + \text{FN}_{\text{Neg}}}; \tag{4.23}$$

- Área bajo la curva

$$\text{AUC} = \int_0^1 \text{TPR}(t)\ \text{dFPR}(t), \tag{4.24}$$

El ajuste con $\omega = 1.5$ proporcionó la mayor capacidad de discriminación global ($AUC = 0.975$), superando al baseline ($\omega = 1.0$, $AUC = 0.966$).

Por el contrario, $\omega = 2$ obtuvo la mejor combinación de exactitud y *recall* para la clase negativa:

$$Acc = 0.9423, \qquad Recall_{Neg} = 0.9231$$

mientras que valores más altos ($\omega \geq 3$) redujeron ambas métricas.

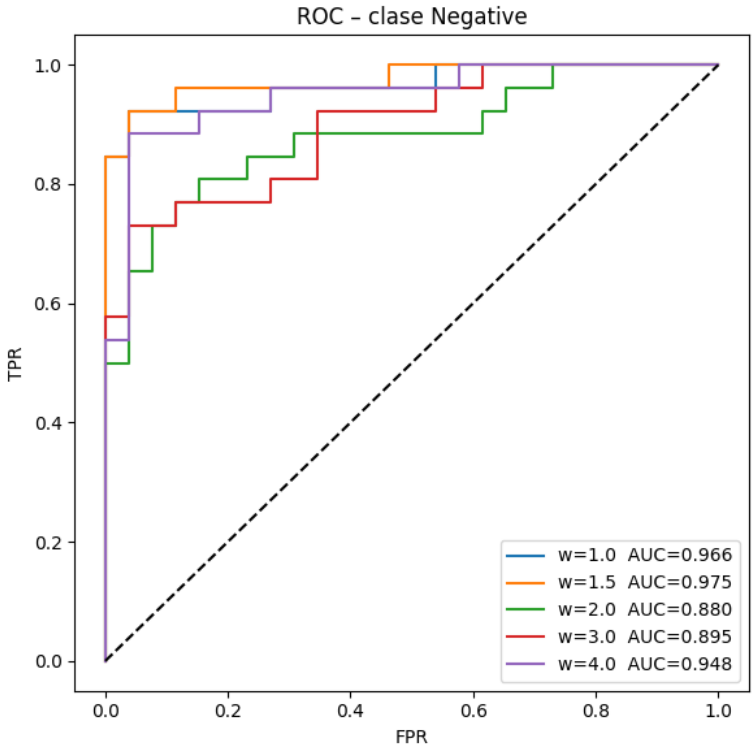


Figura 4.2 Gráfica AUC con los diversos ajustes de pesos w.

Para el modelo $\omega = 2$ se computaron mapas Grad-CAM en tres imágenes de personas, las cuales trabajan actualmente, que pueda observarse en la Figura 4.2.

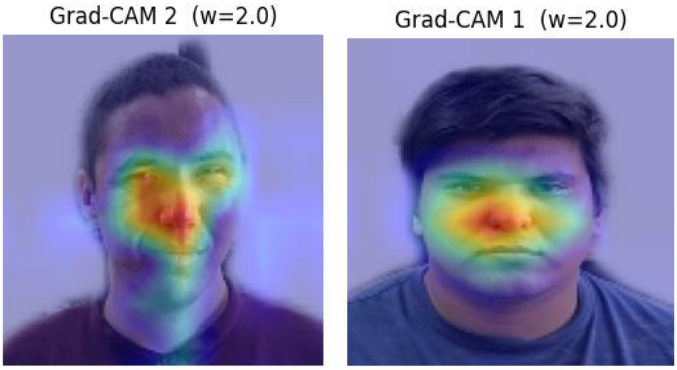


Figura 4.3 Imágenes mapas Grad-CAM para w = 2.

En todos los casos la activación máxima se localiza en la región óculo-nasal, consistente con literatura sobre reconocimiento de tristeza y enfado, y no se observan focos espurios provocados por la reponderación.

Una ponderación moderada ($\omega \approx 1.5$) mejora la discriminación global (AUC), mientras que $\omega = 2$ conserva el recall negativo sin sacrificar exactitud. Por el contrario, tener pesos excesivos ($\omega > 3$) inducen sobre-focalización en el subgrupo de trabajadores y degradan el rendimiento. Además, los mapas Grad-CAM indican que el refuerzo de pesos no desplaza la atención a regiones irrelevantes, manteniendo la interpretabilidad del detector facial.

Estos hallazgos sugieren que la incorporación de información contextual (estado laboral) mediante muestras de pesos puede afinar la detección de emociones negativas, pero requiere calibración cuidadosa del factor de peso para evitar sobre ajuste.

5. IMPLEMENTACIÓN EN TIEMPO REAL

El objetivo de LiExNet es ofrecer una solución ligera y eficiente para la clasificación de emociones en tiempo real, capaz de ejecutarse en dispositivos con recursos computacionales limitados. Por lo tanto, es necesario realizar pruebas en tecnologías destinadas a sistemas embebidos. Para esta tesis se eligieron las tarjetas NVIDIA Jetson TX2 y Xilinx UltraScale+ MPSoC. La tarjeta NVIDIA Jetson TX2 fue seleccionada porque ofrece una arquitectura optimizada para inferencia de modelos de inteligencia artificial, con soporte nativo para CUDA, TensorRT y aceleración mediante GPU. La UltraScale+ MPSoC se eligió porque integra núcleos ARM y lógica programable (FPGA), lo que permite analizar el modelo en un entorno de hardware más personalizado y de bajo nivel. Esta tarjeta es ideal para optimizaciones específicas en consumo, latencia y procesamiento paralelo, fundamentales para su despliegue en aplicaciones embebidas críticas.

Este capítulo se divide de la siguiente manera. La sección 5.1 reporta la implementación de LiExNet con la tarjeta NVIDIA Jetson TX2 y la sección 5.2 presenta el desarrollo con la tarjeta Xilinx UltraScale+ MPSoC.

5.1. NVIDIA Jetson TX2

En el ámbito de los sistemas embebidos para IA, la NVIDIA Jetson TX2 se ha consolidado como una plataforma de referencia. Esta tarjeta está diseñada para ofrecer un alto rendimiento computacional, pero manteniendo un consumo energético moderado que se ha vuelto popular en aplicaciones de visión por computadora, robótica y aprendizaje profundo en el borde.

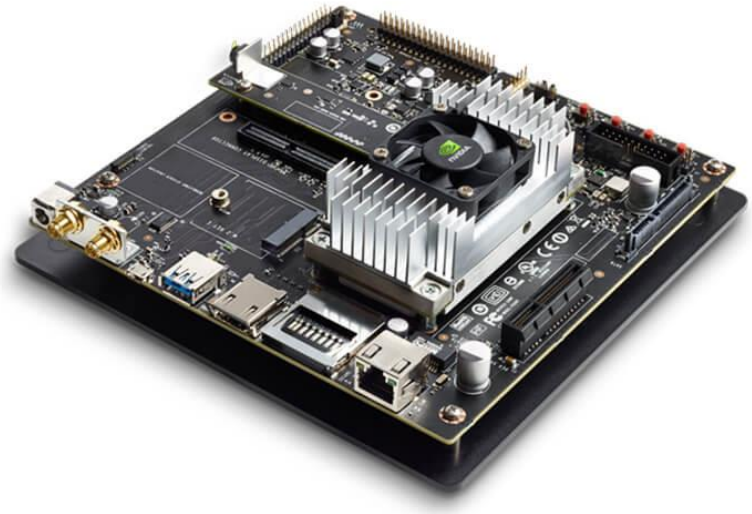


Figura 5.1 NVIDIA Jetson TX2

Lo que hace destacar a la *Jetson TX2* es su núcleo, donde se encuentra el *Tegra X2*, un sistema en un chip (SoC) que combina múltiples elementos: una CPU, una GPU y diversos periféricos, todos cuidadosamente diseñados para operar de manera conjunta y eficiente.

La CPU, presenta una arquitectura con dos núcleos *Denver 2* y cuatro núcleos *ARM Cortex-A57*. Por otra parte, la GPU utiliza la arquitectura *Pascal* de NVIDIA y cuenta con 256 núcleos CUDA.

Además, la Jetson TX2 no se limita a su GPU para tareas de IA. Integra también el *NVIDIA Deep Learning Accelerator* (NVDLA), un coprocesador especialmente diseñado para optimizar el rendimiento en la ejecución de modelos de aprendizaje profundo. Esta inclusión refuerza su enfoque en la inteligencia artificial, permitiendo una mayor eficiencia durante la inferencia, sin necesidad de consumir más recursos de los necesarios.

En términos de memoria, el módulo incorpora 8 GB de LPDDR4 con un ancho de banda de 58.4 GB/s, lo cual es más que suficiente para muchas aplicaciones intensivas en datos. Para el almacenamiento, ofrece 32 GB de memoria eMMC 5.1 y permite expansión mediante

tarjetas microSD o unidades USB, facilitando el desarrollo y prueba de prototipos sin preocuparse demasiado por limitaciones de espacio.

Un aspecto de esta plataforma es su compatibilidad con un ecosistema de herramientas enfocadas en IA para implementar, ajustar y ejecutar modelos de manera eficiente. Entre ellas se encuentran:

- **TensorRT**, para la optimización de inferencias en redes neuronales profundas.
- **CUDA**, que permite aprovechar al máximo el paralelismo de la GPU.
- **cuDNN**, una biblioteca específica para redes convolucionales.
- **DeepStream SDK**, ideal para el procesamiento de video en tiempo real, muy útil en sistemas de vigilancia o análisis visual en entornos dinámicos.

A nivel de software, la *Jetson TX2* funciona con Ubuntu 18.04 LTS y se apoya en *JetPack SDK*, una distribución desarrollada por NVIDIA que incluye los controladores, bibliotecas y herramientas necesarias para implementar LiExNet. También ofrece compatibilidad con bibliotecas como *OpenCV*, optimizadas para ejecución en GPU, lo que resulta útil para detectar estados emocionales en tiempo real.

La *Jetson TX2* alcanza hasta 1.3 TFLOPs en operaciones de precisión flotante de 16 bits (FP16). Su arquitectura le permite ejecutar redes neuronales convolucionales y manejar análisis de video sin mayores compromisos en términos de velocidad o consumo, haciéndola ideal para proyectos embebidos que requieren respuesta inmediata y procesamiento local.

5.1.1. Implementación de la red propuesta en el sistema embebido NVIDIA Jetson TX2

LiExNet se implementó en la NVIDIA Jetson TX2 con JetPack 4.6.4 (L4T 32.7.4), entorno que integra CUDA 10.2, cuDNN 8.2 y TensorRT 8.2. El modelo entrenado en Keras se exportó a formato TensorFlow-Lite y posteriormente se pasó a un formato compatible con TensorRT en precisión FP16. De esta manera, LiExNet pesa 159 kB y conservó los 40 923

parámetros de LiExNet con huella total en memoria de 0.16 MB. La complejidad computacional fue de 27,736,458 FLOPs, cifra que confirma el diseño ultraligero del modelo.

Como verificación del funcionamiento del modelo en el sistema Jetson TX2 se ejecutó una inferencia por lotes sobre diez imágenes de CK+ solo para mostrar que las métricas obtenidas del modelo fueran correctas. El programa cargó el modelo, transfirió las entradas a la GPU y devolvió las etiquetas y probabilidades, manteniendo la exactitud reportada en las pruebas de laboratorio. La Figura 5.2 muestra los resultados; solo se observó un desacierto (expresión de *disgust* clasificada como *happy* mostrada en rojo), situación coherente con los patrones de confusión analizados previamente para este par de emociones. El consumo energético durante la prueba se mantuvo en el modo de potencia máxima de la Jetson TX2 conocida como MAXN, sin registrar estrangulamientos térmicos ni caídas de frecuencia de reloj.

Figura 5.2 Inferencia de LiExNet en la NVIDIA Jetson TX2

5.1.2. Experimentación del modelo en tiempo real

Para evaluar el comportamiento de LiExNet en un escenario continuo se procesó un vídeo de resolución Ultra HD (3840 × 2160) almacenado localmente en la misma Jetson TX2. La resolución espacial de cada imagen se cambió a 128 × 128, se normalizó y se envió a TensorRT. La sesión de prueba arrojó un promedio de 269.44 FPS, valor que sitúa la latencia por cuadro por debajo de los 4 ms, lo que se considera tiempo real, ya que la literatura sobre Edge-AI establece que bastan ≈ 25 FPS (≈ 40 ms de extremo a extremo) para mantener la interacción humano-máquina fluida en tareas de visión por computadora [47].

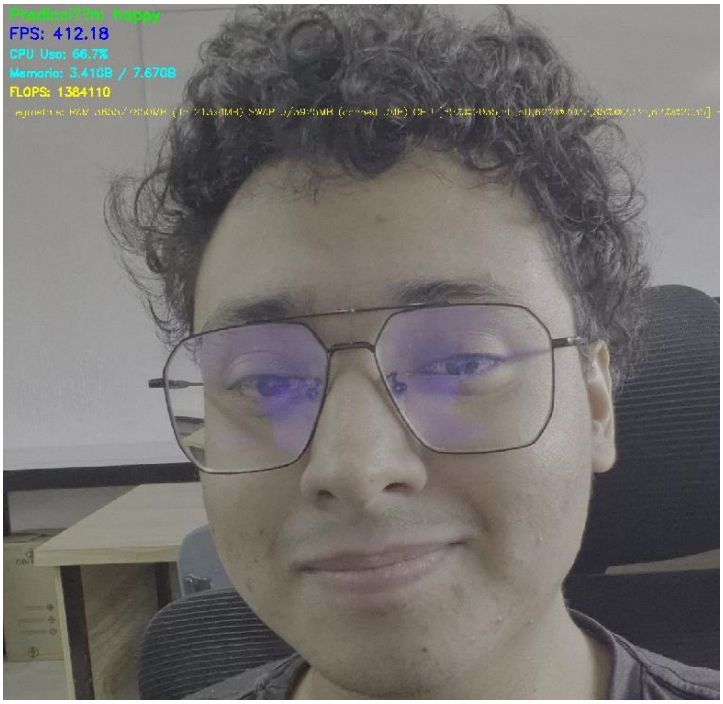


Figura 5.3 Procesamiento en tiempo real en Jetson TX2.

Los recursos utilizados por el sistema embebido se midieron utilizando la herramienta *tegrastats*, estas mediciones reflejan un 38.97 % de ocupación media de CPU, lo que deja margen para procesos concurrentes y un consumo RAM del 88.03 %, este valor se mantiene dentro del límite admisible para la plataforma con 8 GB de memoria. El subsistema gráfico apenas participó en la carga de trabajo, registrando un 2.64 % de actividad en la GPU, circunstancia especialmente valiosa en sistemas embebidos donde la GPU suele reservarse

para tareas de visión adicionales. Todos los ensayos se ejecutaron bajo el perfil de potencia MAXN, garantizando el mayor rendimiento de la TX2.

En conjunto, las métricas confirman que LiExNet sostiene un flujo de vídeo UHD a más de diez veces el umbral mínimo de 25 FPS establecido para aplicaciones interactivo-humanas, sin agotar la capacidad de cómputo del dispositivo y con un requerimiento insignificante de aceleración gráfica [48]. Estos resultados respaldan que el modelo puede utilizarse para aplicaciones embebidas exigentes tales como análisis de comportamiento en entornos industriales o sistemas de advertencia al conductor donde la baja latencia y la eficiencia energética son críticas.

Tabla 5.1 Métricas de desempeño en tiempo real de LiExNet en la NVIDIA Jetson TX

Métrica	**Valor**	**Unidad / Descripción**
FPS promedio	269.44	Cuadros por segundo (vídeo UHD)
Latencia promedio por cuadro	3.71	ms (1000/FPS)
CPU promedio	38.97	Porcentaje de uso
RAM promedio	88.03	% de utilización
GPU promedio	2.64	% de utilización
Modo de potencia	MAXN	Perfil de potencia Jetson TX2

5.2. Xilinx UltraScale+ MPSoC

Xilinx Zynq UltraScale+ MPSoC ZU2CG combina un procesador ARM de 64 bits con una FPGA en el mismo chip, lo que lo convierte en una solución versátil para aplicaciones que requieren tanto procesamiento general como procesamiento paralelo intensivo.

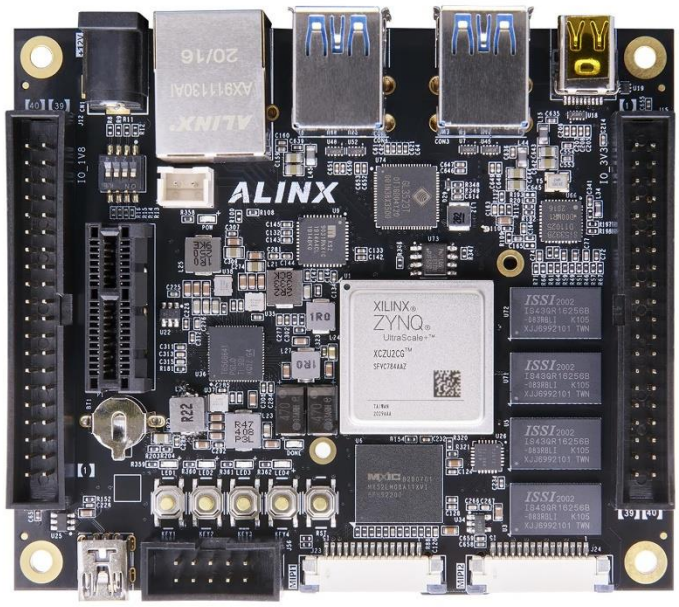


Figura 5.4 Xilinx Zynq UltraScale+ MPSoC ZU2CG

A nivel arquitectónico, el ZU2CG cuenta con un procesador dual-core ARM Cortex-A53, lo cual lo hace apto para correr sistemas operativos como Linux y ejecutar tareas generales de control. A esto se suma una FPGA basada en la arquitectura UltraScale+, con aproximadamente 53 mil células lógicas disponibles para implementar lógica personalizada, coprocesadores dedicados o interfaces específicas.

En cuanto a la memoria, esta plataforma incluye 1 GB de RAM DDR4, junto con 32 MB de memoria *flash* QSPI para el almacenamiento del sistema de arranque y otros datos esenciales. También ofrece soporte para memoria eMMC (de forma opcional) y expansión mediante tarjetas microSD, lo cual permite adaptarse a diferentes necesidades de almacenamiento según el tipo de aplicación.

Uno de los aspectos que hacen a esta plataforma especialmente atractiva para aplicaciones embebidas es su equilibrio entre potencia y consumo. El procesador ARM está optimizado para eficiencia energética, mientras que el consumo total del sistema puede oscilar entre los 3 y 8 watts, dependiendo del uso que se haga de la lógica programable. Esto permite que el

dispositivo sea utilizado en aplicaciones móviles o autónomas sin comprometer demasiado la autonomía.

Aunque el ZU2CG no cuenta con unidades dedicadas a inteligencia artificial como algunos SoC más recientes, su FPGA permite desarrollar aceleradores personalizados que pueden implementarse para tareas como la inferencia de redes neuronales o el procesamiento de imágenes en tiempo real. Herramientas como Vitis AI y OpenCL facilitan esta tarea, permitiendo aprovechar el paralelismo de la FPGA sin necesidad de configurar directamente el sistema mediante síntesis de hardware.

En cuanto al entorno de desarrollo, Xilinx proporciona un conjunto de herramientas robustas que permiten cubrir tanto el diseño del hardware como del software. Vivado es la herramienta principal para la síntesis y diseño de lógica en la FPGA, mientras que Vitis y el antiguo Xilinx SDK permiten el desarrollo en los procesadores ARM y la integración con la lógica programable.

En resumen, el Zynq UltraScale+ ZU2CG es una plataforma robusta que se sitúa en un punto medio ya que es suficientemente flexible para adaptarse a aplicaciones exigentes, pero también suficientemente eficiente para ser utilizada en sistemas embebidos de bajo consumo.

6. RESULTADOS

En este capítulo se presentan los resultados obtenidos con LiExNet respecto a la clasificación de emociones faciales. Los resultados se muestran en términos de entrenamiento y evaluación de LiExNet, validación cruzada y comparación de LiExNet con otros métodos.

La métrica seleccionada para esta evaluación fue la exactitud, ya que permite una comparación directa con trabajos previos y es ampliamente utilizada en modelos de clasificación por su capacidad para reflejar el desempeño general del sistema.

El capítulo se divide de la siguiente manera: la sección 6.1 muestra los resultados de LiExNet con las diferentes bases de datos de reconocimiento de emociones. La sección 6.2 presenta la validación cruzada de LiExNet con KDEF, CK+ y JAFFE. Finalmente, la sección 6.3 muestra la comparación de LiExNet con otros modelos de la literatura.

6.1. Resultados LiExNet

LiExNet fue diseñada para reducir de forma drástica el número de parámetros en el reconocimiento de expresiones faciales sin sacrificar exactitud, combinando convoluciones separables en profundidad, un número moderado de canales y módulos de atención eficiente (ECA). Para evaluar su eficacia se llevaron a cabo tres conjuntos de experimentos: desempeño global, validación cruzada y comparación con el estado del arte. Los resultados se obtuvieron en cuatro bases de datos de emociones descritas en el capítulo 2.

6.1.1. Desempeño en prueba

El desempeño de LiExNet es comparado de en base a las diferentes bases de datos que se utilizaron para comprobar y comparar su desempeño y cantidad de recursos utilizados. Los resultados obtenidos en exactitud final fueron:

- **CK+**: 98.2 %
- **KDEF**: 85.3 %

- **FER2013**: 77.8 %
- **JAFFE**: 72.0 %

Estos resultados demuestran que LiExNet mantiene un desempeño competitivo incluso en bases de datos con alta variabilidad o tamaño limitado.

En FER2013, LiExNet muestra una curva de aprendizaje clara con rápida convergencia. La precisión de validación alcanza aproximadamente 78 % hacia la época 80, estabilizándose después. La gráfica superior derecha de la Figura 6.1 refleja una diferencia sostenida entre las curvas de entrenamiento y validación, pero sin evidencia clara de sobreajuste, lo cual sugiere una buena capacidad de generalización en este conjunto más complejo.

El resultado robusto de FER2013 hizo que se utilizara como base para el entrenamiento de JAFE, dado el tamaño reducido del conjunto, aplicando *fine-tuning*. Como resultado de este entrenamiento el modelo logró una precisión promedio del 71 % en validación cruzada estratificada de cinco pliegues. Como se aprecia en la gráfica correspondiente inferior izquierda en la Figura 6.1.

Por otro lado, la base de datos KDEF mostró una curva estable que se mantuvo alrededor del 85 % de precisión tras las primeras 200 épocas. En la gráfica superior izquierda de la Figura 6.1, se observa una evolución paralela entre entrenamiento y validación, con ligeras oscilaciones en la precisión de validación, lo que indica un comportamiento predecible y robusto del modelo. Esta base de datos fue la que presentó la menor mejora en el desempeño del modelo a lo largo de los experimentos realizados en este trabajo de investigación.

Finalmente, LiExNet alcanzó su mejor desempeño en CK+, logrando una precisión cercana al 98 %. En la gráfica inferior derecha de la Figura 6.1, se observa una convergencia rápida, alrededor de la época 100 y estabilidad absoluta en ambas curvas a partir de la época 150. La variabilidad puntual de la curva de validación es mínima y no afecta el rendimiento global.

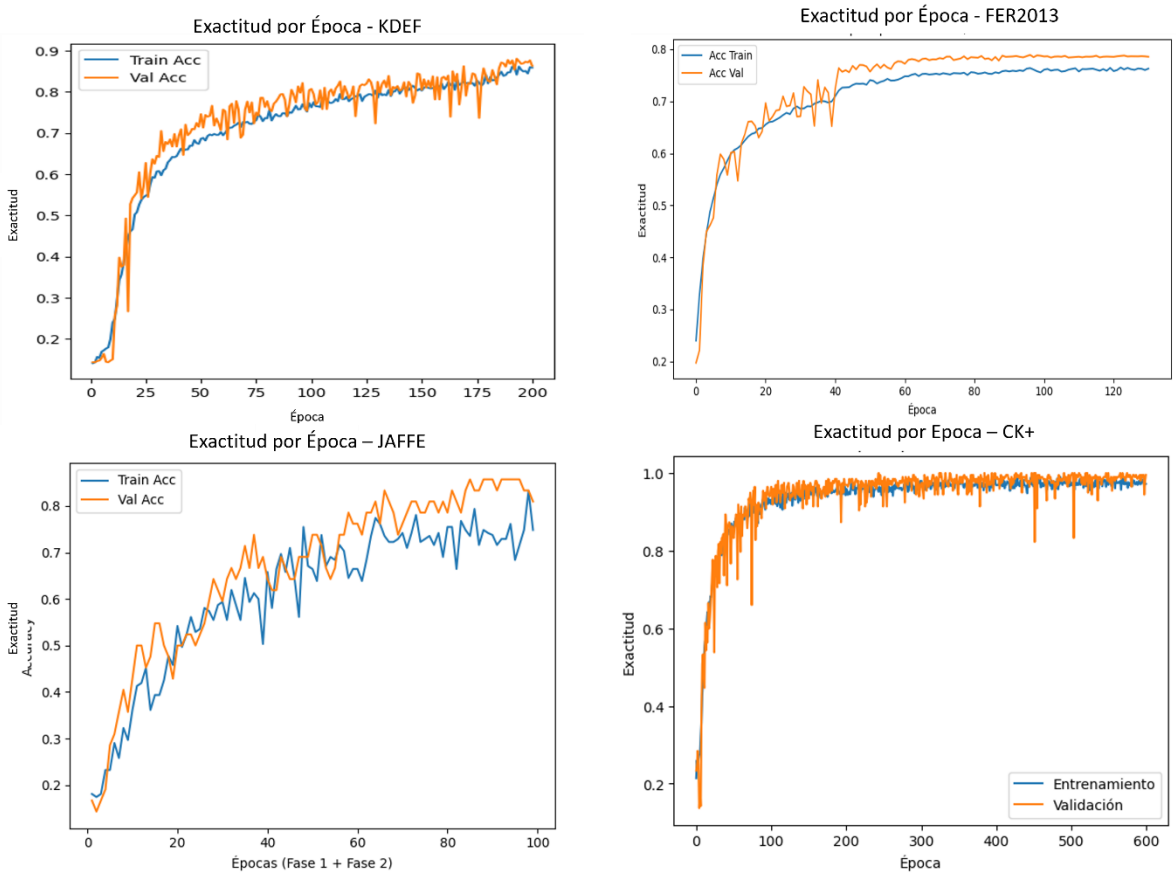


Figura 6.1 Gráficas de Exactitud de LiExNet para las diferentes bases de datos.

6.1.2. Desempeño en validación

El análisis de las matrices de confusión permite evaluar con mayor precisión qué emociones fueron correctamente clasificadas y cuáles presentaron mayores niveles de confusión para el modelo LiExNet en cada base de datos.

LiExNet logró un desempeño sobresaliente en CK+, con clasificaciones casi perfectas y apenas una leve confusión entre AN y DI, emociones que suelen confundirse normalmente por los modelos por ser tan similares en las expresiones faciales, estos resultados pueden observarse en la Figura 6.2 en la matriz inferior derecha. Por otro lado, FER2013 y KDEF obtuvieron buena precisión general, especialmente en emociones como HA y DI, aunque se presentó confusión recurrente entre SA y NE, un patrón común debido a la baja intensidad

emocional o la ambigüedad facial en estas clases, estos resultados pueden observarse en la Figura 6.2 en las dos matrices superiores.

Finalmente, en JAFFE el modelo mantuvo un rendimiento aceptable, con buen reconocimiento en SU y FE, pero con errores más marcados en clases como SA y HA. En general, las emociones con expresiones más distintivas fueron clasificadas correctamente, mientras que las de baja expresividad generaron mayor ambigüedad. Estos resultados confirman que la calidad, tamaño y balance del conjunto de datos influyen directamente en la precisión del modelo, especialmente al diferenciar emociones sutiles o similares, estos resultados pueden observarse en la Figura 6.2 en la matriz inferior izquierda.

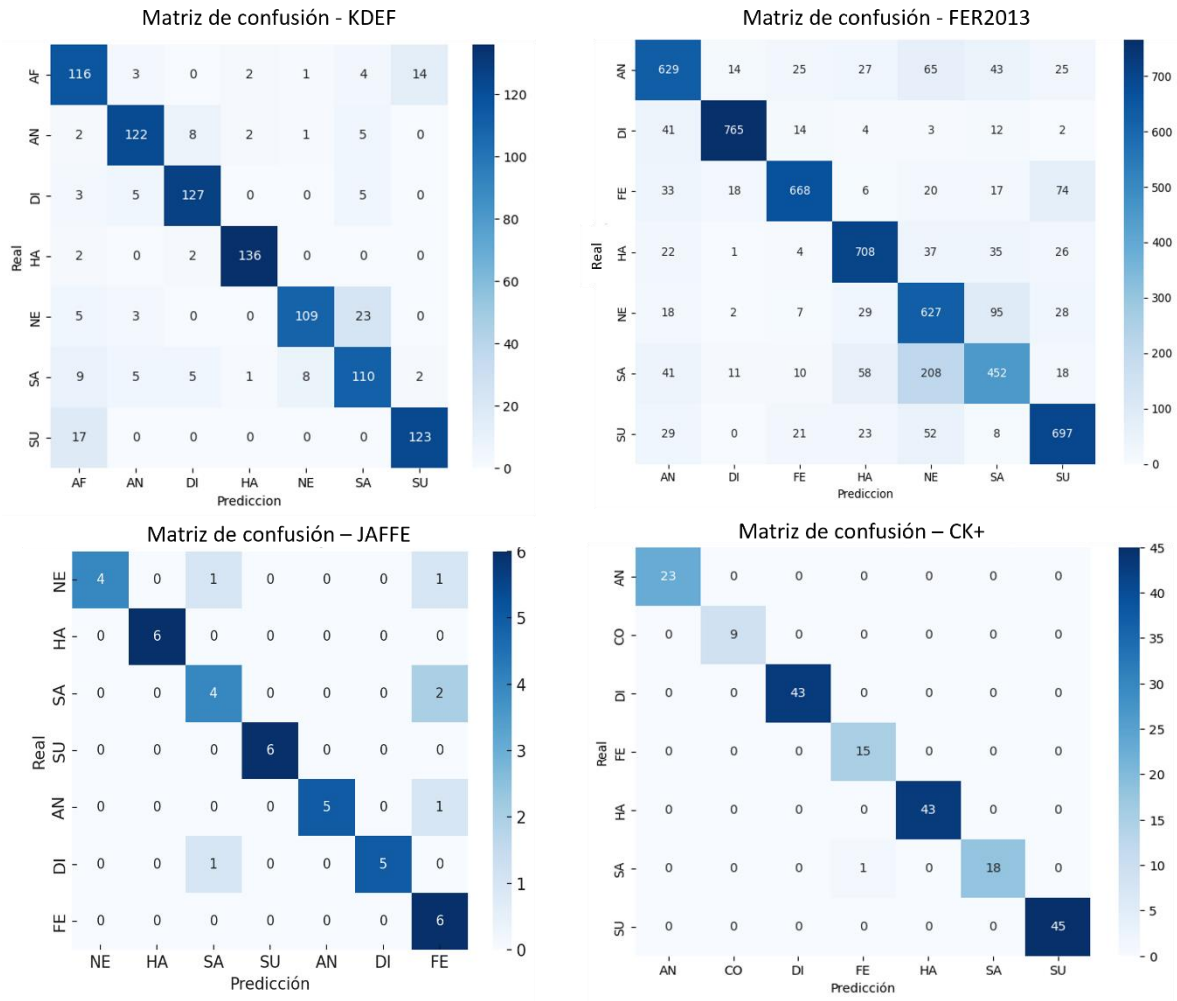


Figura 6.2 Matrices de confusión de LiExNet para las diferentes bases de datos.

Además, hay que hacer hincapié en que la ventaja decisiva de LiExNet se aprecia en el costo computacional ya que con apenas 1.78 GFLOPS y 40,000 parámetros, el modelo es entre 10 y 20 veces más liviano que VGG-16, aun así, supera a ambas redes en exactitud promedio en los tres conjuntos mayores (CK+, KDEF y FER2013). Además, su consumo máximo de RAM se mantuvo en 105 kB durante inferencia, frente a los ~68 MB requeridos por VGG-16.

En términos de velocidad, el modelo procesa 269.4 cuadros por segundo en la Jetson TX2; la latencia por imagen se mantiene por debajo de 4 ms, límite comúnmente aceptado para aplicaciones de tiempo real en visión embebida. La combinación de alta precisión y requisitos mínimos de memoria y cómputo valida la efectividad de la estrategia de convoluciones separables y módulos de atención ECA integrada en la arquitectura.

6.2. Validación cruzada

Con el objetivo de evaluar la capacidad de generalización de LiExNet se aplicó validación cruzada estratificada de cinco pliegues a los conjuntos CK+, KDEF y JAFFE, mientras que FER2013 se mantuvo con una partición fija.

CK+

En CK+, los cinco pliegues arrojaron exactitudes individuales de 94.4 %, 100 %, 97.4 %, 95.4 % y 98.0 %. El promedio resultante fue 97.0 % con una desviación estándar de apenas ±0.2 %, lo que indica una variación menor al 2 % entre repeticiones y confirma que el modelo reproduce su rendimiento incluso cuando se ve obligado a entrenarse con un subconjunto reducido de imágenes.

KDEF

Para KDEF, base más heterogénea en pose, iluminación y número de sujetos, las exactitudes por pliegue fueron 86.94 %, 86.02 %, 85.20 %, 82.02 % y 86.31 %, lo que se

traduce en una media de 85.3 % y una desviación estándar de ±1.7 %. Aun con la mayor variabilidad intrínseca del conjunto, la oscilación se mantuvo dentro de un margen estrecho (intervalo 82.0–86.9 %), lo que respalda la robustez del modelo y demuestra que la reducción drástica de parámetros no compromete la consistencia entre cada pliegue.

FER2013

En el caso de FER2013, que contiene más de 23 000 imágenes "in-the-wild", se optó por un único entrenamiento desde cero y una partición de prueba balanceada para no multiplicar el tiempo de cómputo. Tras un máximo de 100 épocas con early-stopping (paciencia 15–40), se alcanzó una precisión de 77.79 %. Aunque no se calcularon métricas de variabilidad pliegue a pliegue, esta cifra representa un desempeño sólido frente a ruido, variación de pose y condiciones de iluminación difíciles.

JAFFE

Finalmente, JAFFE se abordó mediante *fine-tuning* de los pesos aprendidos en FER2013 y también se sometió a un esquema de validación cruzada de cinco pliegues. Cada pliegue se entrenó en dos fases:

1. Fase de congelación parcial ("warm-up"): se congelaron los bloques convolucionales y solo se ajustaron las capas densas finales durante 20 – 30 épocas a un ritmo de aprendizaje bajo (1×10^{-4}).
2. Fase de descongelación completa: se liberaron todas las capas y se continuó el entrenamiento hasta 100 épocas con *early-stopping* (paciencia 15).

Las exactitudes obtenidas por pliegue fueron 72.09 %, 65.12 %, 74.42 %, 85.71 % y 61.90 %, lo que arroja una media de 71.8 % con desviación estándar de ± 0.5 %. Aun con el riesgo inherente de sobreajuste, las dos fases de entrenamiento por pliegue permitieron conservar rasgos discriminantes y sostener un desempeño competitivo para un corpus tan reducido.

En conjunto, estos experimentos demuestran que LiExNet mantiene un rendimiento estable y predecible en contextos muy distintos. La combinación de validación cruzada y pruebas fijas ofrece, por tanto, una evidencia sólida de la capacidad del modelo para generalizar sin sacrificar su ligereza computacional. Los resultados resumidos de la validación cruzada pueden verse en la Tabla 6.1.

Tabla 6.1 Exactitud en validación cruzada de LiExNet en CK+, KDEF y JAFFE.

Método	***k=1***	***k=2***	***k=3***	***k=4***	***k=5***	$\mu + \sigma$
CK+	94.40 %	100 %	97.40 %	95.4 %	98.0 %	97.05 % ± 0.0197
KDEF	86.94 %	86.02 %	85.20 %	82.02 %	86.31 %	85.30 % ± 1.7
JAFFE	72.09 %	65.12 %	74.42 %	85.71 %	61.90 %	71.80 % ± 0.5

6.3. Comparación con otros métodos del estado del arte

LiExNet obtuvo resultados comparables e incluso superiores a otros modelos en el estado del arte, los resultados pueden observarse en la Tabla 6.2 donde se compara el rendimiento de LiExNet con otros métodos de reconocimiento de emociones que utilizan principalmente los conjuntos CK+, KDEF, FER2013 y JAFFE. El modelo alcanza en promedio 98.2 % de exactitud en CK+ bajo validación cruzada, 85 % en KDEF, 72 % en JAFFE y 77 % en la versión balanceada de FER2013, lo que demuestra su competitividad en diferentes dominios y distribuciones de clase.

Revisiones más amplias de la literatura confirman que existen métodos que superan esas cifras, pero a costa de una carga computacional mucho mayor. El enfoque orientado a *anti-spoofing* descrito en [49], por ejemplo, obtiene resultados superiores gracias a un uso intensivo de recursos. De manera similar, la técnica basada en transformada wavelet con gradiente de [50] reporta excelentes métricas, sobre todo en CK+, pero incrementa considerablemente la complejidad de cálculo. Además, Varios trabajos recurren a arquitecturas profundas de gran escala como DenseNet-161 o modelos propios que superan

los 20 millones de parámetros [51] [52], lo que implica mayores demandas de memoria y tiempos de inferencia. Otros estudios [53][54][55][56] reducen todavía más la complejidad eliminando clases como *contempt* o neutral y entrenando con solo seis emociones, lo que dificulta una comparación directa con LiExNet.

Tabla 6.2 Comparación de las métricas de LiExNet con otros métodos

Método	Base de datos	Exactitud	Método	Base de datos	Exactitud
[49]	KDEF CK+	85 % 97.86 %	[58]	FER2013 JAFFE	74.64 % 98.44 %
[56]	CK+ JAFFE	91.11 % 63.33 %	[59]	CK+ JAFFE	96.46 % 91.27 %
[57]	CK+ FER2013 JAFFE	98.9 % 75.82 % 98.52 %	[53]	CK+ JAFFE	94.09 % 91.79 %
[60]	CK+	97.2 %	[61]	CK+ FER2013 JAFFE	98 % 70.02 % 92.8 %
[54]	CK+	98.38 %	[51]	KDEF JAFFE	98.78 % (Hold-out) 100 % (Hold out)
[62]	CK+ JAFFE	99.36 % 95.65 %	[63]	CK+ JAFFE	91.42 % 92.85 %
[64]	FER2013	72.35 %	[65]	CK+	98.5 %
[52]	CK+ JAFFE	93.24 % 95.23 %	[66]	FER2013	88.56 % (fused)
[67]	CK+	85 %	[68]	CK+ JAFFE	98.9 % 97.1 %
[69]	FER2013	54 %			
LiExNet	KDEF FER2013 CK+ JAFFE	85.3 % 77.8 % 98.2 % 72 %			

LiExNet mantiene las siete emociones básicas propuestas por Paul Ekman y añade la categoría neutral (presente en KDEF y FER2013) como estado de referencia. Contar con neutral es crucial para distinguir entre la presencia y la ausencia de expresividad facial en aplicaciones continuas o no controladas, reforzando así la validez práctica y psicológica del modelo. Aunque ciertos métodos del estado del arte superan la exactitud de LiExNet gracias

a transformaciones costosas o redes de gran profundidad, el modelo se posiciona como una alternativa eficiente y conceptualmente sólida para el reconocimiento facial de emociones.

7. CONCLUSIONES

Este capítulo sintetiza el trabajo desarrollado en este proyecto de investigación cuyo eje central fue el diseño, entrenamiento e implementación de LiExNet, una arquitectura ligera para el reconocimiento emociones y estados afectivos en tiempo real mediante expresiones faciales.

La metodología que se utilizó para el desarrollo del modelo que es LiExNet se basó principalmente en experimentación de arquitecturas conocidas en la literatura, el objetivo de estos experimentos era conocer como las diferentes arquitecturas podían aportar al objetivo de reconocimiento de emociones, de esta manera se conocerían los aspectos más útiles y las debilidades de cada modelo con el fin de diseñar un modelo adaptado a la tarea específica del reconocimiento de emociones, que fuera eficiente y tuviera métricas competitivas con otros métodos de la literatura.

Posteriormente, se adaptó el aprendizaje que se tenía con el reconocimiento de emociones a estados afectivos mediante *fine-tunning* mediante una base de datos propia de la institución que avala este proyecto de investigación.

Finalmente se comprobó su funcionalidad en sistemas de bajos recursos utilizando una tarjeta de desarrollo NVIDIA Jetson TX2, implementando además el modelo en tiempo real, alcanzando aproximadamente 270 FPS, resultados mayores a los necesarios para que el sistema fuera considerado tiempo real que son aproximadamente 25 FPS, esto demostró que LiExNet no solo es una red lo suficientemente eficiente para un sistema embebido con recursos utilices para la implementación de redes neuronales como lo son una unidad de gráficos, si no que tiene suficiente margen de ganancia para dispositivos con recursos aún menos que el sistema Jetson Tx2.

En síntesis, LiExNet ofrece:

- Equilibrio entre rendimiento y ligereza ya que proporciona tasas de acierto competitivas con una fracción de los parámetros y FLOPs de redes más grandes.
- Cobertura emocional completa, porque respeta la taxonomía clásica de Ekman e incorpora neutral, lo que favorece la comparación interdisciplinaria y la validez con sistemas respaldados psicológicamente.
- Portabilidad a sistemas embebido debido a su bajo costo computacional el cual permite la inferencia en tiempo real en plataformas como la NVIDIA Jetson TX2.

LiExNet es un modelo que aporta tanto al área de reconocimiento de emociones mediante expresiones faciales, por ser capaz de superar modelos en sus resultados, como al área de computación afectiva, donde se utilizan algoritmos capaces de entender o identificar emociones humanas para interactuar de manera más ergonómica con el ser humano. Finalmente, LiExNet es una aportación a modelos profundos que se desempeñan de manera eficiente en entornos donde no existe una conexión a servidores especializados o que necesitan estar en constante movimiento y los componentes computacionales no pueden ser excesivamente grandes debido a la fragilidad de estos y en su lugar deben de utilizar sistemas de cómputo de recursos limitados y/o de propósito único como sistemas de control.

Sin embargo, LiExNet no alcanzó cifras significativas para bases de datos como lo son FER2013 y JAFFE que fueron bases de datos para las cuales se obtuvieron los peores resultados, y si bien LiExNet si superó a algunos modelos con FER2013, este no lo hizo con JAFFE. Si bien este no es un objetivo del presente trabajo de investigación, si puede ser un área de oportunidad de mejora para una versión futura del modelo.

Finalmente, LiExNet también presenta oportunidades de extensión hacia otras tareas de clasificación relacionadas con el estado emocional, como la detección de estrés, depresión o ansiedad. Gracias a su arquitectura eficiente, la incorporación de capas adicionales o adaptaciones específicas implicaría un incremento mínimo en la cantidad de parámetros, lo

que permitiría mantener un bajo costo computacional. Esto abre la posibilidad de desarrollar sistemas de detección en tiempo real para estas condiciones.

8. BIBLIOGRAFÍA

[1] A. Mollahosseini, B. Hasani y M. H. Mahoor, "AffectNet: A Database for Facial Expression, Valence, and Arousal Computing in the Wild," *IEEE Transactions on Affective Computing*, vol. 10, no. 1, pp. 18–31, Jan.–Mar. 2019, doi: 10.1109/TAFFC.2017.2740923.

[2] S. Woo, J. Park, J. Y. Lee y I. S. Kweon, "CBAM: Convolutional Block Attention Module," en *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19, doi: 10.1007/978-3-030-01234-2_1.

[3] X. Wang, R. Girshick, A. Gupta y K. He, "Non-local Neural Networks," en *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 7794–7803, doi: 10.1109/CVPR.2018.00813.

[4] K. He, X. Zhang, S. Ren y J. Sun, "Deep Residual Learning for Image Recognition," en *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[5] M.A.H. Akhand, S. Roy, N. Siddique, M.A.S. Kamal, T. Shimamura, Facial emotion recognition using transfer learning in the deep CNN, Electronics 10(9) (2021) 1036

[6] J. Lee, H. Lee y M. Shin, "Driving Stress Detection Using Multimodal Convolutional Neural Networks with Nonlinear Representation of Short-Term Physiological Signals", Sensors, vol. 21, n.º 7, p. 2381, marzo de 2021. Accedido el 12 de mayo de 2025. [En línea]. Disponible: https://doi.org/10.3390/s21072381

[7] Q. Wang, B. Wu, y col., "ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks," en Proc. CVPR, 2020.

[8] Real Academia Española, "inteligencia artificial", *Diccionario de la lengua española*, [En línea]. Disponible en: https://dle.rae.es/inteligencia

[9] T. M. Mitchell, Machine Learning. New York, NY, USA: McGraw Hill, 1997.

[10] R. S. Sutton y A. G. Barto, Reinforcement Learning: An Introduction, 2.ª ed. Cambridge, MA, USA: MIT Press, 2018.

[11] I. Goodfellow, Y. Bengio y A. Courville, Deep Learning. Cambridge, MA, USA: MIT Press, 2016.

[12] R. Szeliski, Computer Vision: Algorithms and Applications, 2.ª ed. Cham, Switzerland: Springer, 2022.

[13] C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006, pp. 373–389.

[14] J. Eisenstein, Natural Language Processing, The MIT Press, 2019, pp. 124–131

[15] A. Howard et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," arXiv preprint arXiv:1704.04861, 2017.

[16] K. He, X. Zhang, S. Ren y J. Sun, "Deep Residual Learning for Image Recognition," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* (CVPR), pp. 770–778, 2016.

[17] G. Huang, Z. Liu, L. van der Maaten y K. Q. Weinberger, "Densely Connected Convolutional Networks," Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 4700–4708, 2017.

[18] P. Ekman, "Emotions Revealed: Recognizing Faces and Feelings to Improve Communication and Emotional Life", 2nd ed., New York: Holt Paperbacks, 2007.

[19] B. L. Fredrickson, *Positivity*. New York: Crown, 2009.

[20] R. S. Lazarus, *Emotion and Adaptation*. New York: Oxford University Press, 1991.

[21] R. Plutchik, *Emotion: A Psychoevolutionary Synthesis*. New York: Harper & Row, 1980.

[22] D. Watson and A. Tellegen, "Toward a consensual structure of mood," Psychological Bulletin, vol. 98, no. 2, pp. 219–235, 1985.

[23] H. Selye, *The Stress of Life*, New York: McGraw-Hill, 1956.

[24] American Psychiatric Association, Diagnostic and Statistical Manual of Mental Disorders (DSM-5-TR), 5th ed., text rev. Washington, DC, 2022.

[25] L. A. Clark and D. Watson, "Tripartite model of anxiety and depression: Psychometric evidence and taxonomic implications," Journal of Abnormal Psychology, vol. 100, no. 3, pp. 316–336, 1991

[26] S. K. Khare et al., "Revisión de avances en técnicas de reconocimiento de emociones," IEEE Transactions on Affective Computing, vol. 8, n.º 4, pp. 321-334, 2023

[27] IMSS. “Estrés Laboral”. https://www.imss.gob.mx. Accedido el 16 de mayo de 2024. [En línea]. Disponible: https://www.imss.gob.mx/salud-en-linea/estres-laboral

[28] W. Hu, D. G. Xu, Z. Fan, F. Liu, y Y. He, «VIS-TOP: Visual Transformer Overlay Processor», arXiv (Cornell University), oct. 2021, doi: 10.48550/arxiv.2110.10957

[29] J. P. Domínguez-Morales, Á. Jiménez-Fernández, M. Domínguez-Morales, y G. Jiménez-Moreno, «Deep Neural Networks for the Recognition and Classification of Heart Murmurs Using Neuromorphic Auditory Sensors», IEEE Transactions On Biomedical Circuits And Systems, vol. 12, n.o 1, pp. 24-34, feb. 2018, doi: 10.1109/tbcas.2017.2751545.

[30] C. Zhang, P. Li, G. Sun, Y. Guan, B. Xiao, y J. Cong, «Optimizing FPGA-based Accelerator Design for Deep Convolutional Neural Networks», *Association For Computing Machinery*, feb. 2015, doi: 10.1145/2684746.2689060.

[31] C. Jain, K. Sawant, M. B. Rehman, y R. Kumar, «Emotion detection and characterization using facial features», 3rd International Conference And Workshops On Recent Advances And Innovations In Engineering (ICRAIE), nov. 2018, doi: 10.1109/icraie.2018.8710406.

[32] A. A. Alsheikhy y Y. Fahem, «Design of Embedded Vision System based on FPGA-SoC», *International Journal Of Advanced Computer Science And Applications*, vol. 10, n.º 10, ene. 2019, doi: 10.14569/ijacsa.2019.0101013.

[33] V. Gupta, S. Juyal, G. Singh, C. Killa, y N. Gupta, «Emotion recognition of audio/speech data using deep learning approaches», Journal Of Information And Optimization Sciences, vol. 41, n.o 6, pp. 1309-1317, ago. 2020, doi: 10.1080/02522667.2020.1809089.

[34] B. Phavish, G. Suddul, S. Armoogum, y R. Foogooa, «Identifying Human Emotions from Facial Expressions with Deep Learning», *Zooming Innovation In Consumer Technologies Conference (ZINC)*, may 2020, doi: 10.1109/zinc50678.2020.9161445.

[35] A. Ali Alhussan et al., “Facial Expression Recognition Model Depending on Optimized Support Vector Machine”, Comput., Mater. & Continua, vol. 76, n.º 1, pp. 499–515, 2023. Accedido el 9 de septiembre de 2024. [En línea].

[36] R. Singh, S. Saurav, T. Kumar, R. Saini, A. Vohra y S. Singh, “Facial expression recognition in videos using hybrid CNN & ConvLSTM”, Int. J. Inf. Technol., marzo de 2023. Accedido el 9 de septiembre de 2024. [En línea].

[37] N. Kumar H N, A. S. Kumar, G. Prasad M S y M. A. Shah, “Automatic facial expression recognition combining texture and shape features from prominent facial regions”, IET Image Process., noviembre de 2022. Accedido el 9 de septiembre de 2024. [En línea]. Disponible: https://doi.org/10.1049/ipr2.12700

[38] S. M. Joshi, P. P. Mane, P. Kalavadekar, A. S. Patil, y G. D. Londhe, "Development and Evaluation of a New Algorithm for Real-Time Image Processing," Adv. Nonlinear Var. Inequal., vol. 27, n.º 3, pp. 498–510, julio de 2024. Accedido el 9 de septiembre de 2024. [En línea]. Disponible: https://internationalpubls.com

[39] Litty A., "Real-Time Image Processing in Bioinformatics Using GPU-Accelerated Deep Learning," EasyChair Preprint, no. 13907, pp. 1-15, julio de 2024. Accedido el 9 de septiembre de 2024. [En línea]. Disponible: https://easychair.org

[40] S. K. Khare, V. Blanes-Vidal, E. S. Nadimi, and U. R. Acharya, "Emotion recognition and artificial intelligence: A systematic review (2014–2023) and research recommendations," Information Fusion, vol. 102, p. 102019, 2024. doi: 10.1016/j.inffus.2023.102019

[41] M. J. Lyons, S. Akamatsu, M. Kamachi, y J. Gyoba, "Coding facial expressions with Gabor wavelets," en Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition, Nara, Japón, 1998, pp. 200–205.

[42] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, y I. Matthews, "The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression," en 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops, San Francisco, CA, EE. UU., 2010, pp. 94–101.scite.ai+4ResearchGate+4Papers with Code+4

[43] E. Goeleven, R. De Raedt, L. Leyman, y B. Verschuere, "The Karolinska Directed Emotional Faces: A validation study," Cognition and Emotion, vol. 22, no. 6, pp. 1094–1118, 2008

[44] S. Bobojanov, B. M. Kim, M. Arabboev, y S. Begmatov, “Comparative Analysis of Vision Transformer Models for Facial Emotion Recognition Using Augmented

Balanced Datasets," Applied Sciences, vol. 13, no. 22, Art. no. 12271, Nov. 2023, doi: 10.3390/app132212271.

[45] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li y L. Fei Fei, "ImageNet: A Large Scale Hierarchical Image Database," Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 248–255, Jun. 2009, doi: 10.1109/CVPR.2009.5206848.

[46] I. Loshchilov and F. Hutter, "SGDR: Stochastic Gradient Descent with Warm Restarts," Proc. International Conference on Learning Representations (ICLR), Toulon, France, Apr. 2017.

[47] A. He, C. Luo, X. Tian and W. Zeng, "A Twofold Siamese Network for Real-Time Object Tracking," in Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 2018, pp. 4834–484

[48] U. S. Department of Homeland Security, Science and Technology Directorate, Wearable Camera Systems Focus Group Report (SAVER Program), ManTech International Corp., Tech. Rep., Mar. 2012.

[49] H.A. Shehu, W.N. Browne, H. Eisenbarth, An anti-attack method for emotion categorization from images, Appl. Soft Comput. 128 (2022) 109456, http://dx.doi.org/10.1016/j.asoc.2022.109456, URL https://www.sciencedirect.com/science/article/pii/S1568494622005695.

[50] R. Kumar, S. Muniasamy, N. Arumugam, Facial emotion recognition using subband selective multilevel stationary wavelet gradient transform and fuzzy support vector machine, Vis. Comput. 37 (2021) 1–15, http://dx.doi.org/10.1007/s00371-020-01988-1.

[51] M.A.H. Akhand, S. Roy, N. Siddique, M.A.S. Kamal, T. Shimamura, Facial emotion recognition using transfer learning in the deep CNN, Electronics 10 (9) (2021) http://dx.doi.org/10.3390/electronics10091036, URL https://www.mdpi.com/2079-9292/10/9/1036.

[52] D.K. Jain, P. Shamsolmoali, P. Sehdev, Extended deep neural network for facial emotion recognition, Pattern Recognit. Lett. 120 (2019) 69–74, http://dx.doi.org/10.1016/j.patrec.2019.01.008, URL https://www.sciencedirect.com/ science/article/pii/S016786551930008X.

[53] S.L. Happy, A. Routray, Automatic facial expression recognition using features of salient facial patches, IEEE Trans. Affect. Comput. 6 (1) (2015) 1–12, http://dx.doi.org/10.1109/TAFFC.2014.2386334.

[54] N. Sun, Q. Li, R. Huan, J. Liu, G. Han, Deep spatial-temporal feature fusión for facial expression recognition in static images, Pattern Recognit. Lett. 119 (2019) 49–61, http://dx.doi.org/10.1016/j.patrec.2017.10.022, Deep Learning for Pattern Recognition. URL https://www.sciencedirect.com/science/article/pii/S0167865517303902.

[55] D.K. Jain, P. Shamsolmoali, P. Sehdev, Extended deep neural network for facial emotion recognition, Pattern Recognit. Lett. 120 (2019) 69–74, http://dx.doi.org/10.1016/j.patrec.2019.01.008, URL https://www.sciencedirect.com/science/article/pii/S016786551930008X.

[56] D. Sen, S. Datta, R. Balasubramanian, Facial emotion classification using concatenated geometric and textural features, Multimedia Tools Appl. 78 (2019) 10287–10323, http://dx.doi.org/10.1007/s11042-018-6537-9.

[57] J. Li, K. Jin, D. Zhou, N. Kubota, Z. Ju, Attention mechanism-based CNN for facial expression recognition, Neurocomputing 411 (2020) 340–350, http://dx.doi.org/10.1016/j.neucom.2020.06.014, URL https://www.sciencedirect.com/science/article/pii/S0925231220309838.

[58] I. Haider, H.-J. Yang, G.-S. Lee, S.-H. Kim, Robust human face emotion classification using triplet-loss-based deep CNN features and SVM, Sensors 23 (10) (2023) http://dx.doi.org/10.3390/s23104770, URL https://www.mdpi.com/1424-8220/23/10/4770.

[59] J.-H. Kim, B.-G. Kim, P.P. Roy, D.-M. Jeong, Efficient facial expression recognition algorithm based on hierarchical deep neural network structure, IEEE Access 7 (2019) 41273–41285, http://dx.doi.org/10.1109/ACCESS.2019.2907327.

[60] P. Rodriguez, G. Cucurull, J. Gonzàlez, J.M. Gonfaus, K. Nasrollahi, T.B. Moeslund, F.X. Roca, Deep pain: Exploiting long short-term memory networks for facial expression classification, IEEE Trans. Cybern. 52 (5) (2022) 3314–3324, http://dx.doi.org/10.1109/TCYB.2017.2662199.

[61] S. Minaee, M. Minaei, A. Abdolrashidi, Deep-emotion: Facial expression recognition using attentional convolutional network, Sensors 21 (9) (2021) http: //dx.doi.org/10.3390/s21093046, URL https://www.mdpi.com/1424-8220/21/9/3046.

[62] A. Khattak, M.Z. Asghar, M. Ali, U. Batool, An efficient deep learning technique for facial emotion recognition, Multimedia Tools Appl. 81 (2) (2022)1649–1683, http://dx.doi.org/10.1007/s11042-021-11298-w.

[63] Z. ullah, L. Qi, D. Binu, B.R. Rajakumar, B. Mohammed Ismail, 2-D canonical correlation analysis based image super-resolution scheme for facial emotion recognition, Multimedia Tools Appl. 81 (10) (2022) 13911–13934, http://dx.doi.org/10.1007/s11042-022-11922-3.

[64] H. Li, H. Xu, Deep reinforcement learning for robust emotional classification infacial expression recognition, Knowl.-Based Syst. 204 (2020) 106172, http://dx. doi.org/10.1016/j.knosys.2020.106172, URL https://www.sciencedirect.com/ science/article/pii/S0950705120304081.

[65] K. Chowdary, T. Nguyen, D. Hemanth, Deep learning-based facial emotion recognition for human–computer interaction applications, Neural Comput. Appl. (2021) 1–18, http://dx.doi.org/10.1007/s00521-021-06012-8.

[66] H. Zhang, A. Jolfaei, M. Alazab, A face emotion recognition method using convolutional neural network and image edge computing, IEEE Access 7 (2019) 159081–159089, http://dx.doi.org/10.1109/ACCESS.2019.2949741.

[67] N.D. Mehendale, Facial emotion recognition using convolutional neural networks (FERC), SN Appl. Sci. 2 (2020) 1–8.

[68] R. Kumar, S. Muniasamy, N. Arumugam, Facial emotion recognition using subband selective multilevel stationary wavelet gradient transform and fuzzy support, vector machine, Vis. Comput. 37 (2021) 1–15, http://dx.doi.org/10.1007/s00371-020-01988-1.

[69] K. Sarvakar, R. Senkamalavalli, S. Raghavendra, J. Santosh Kumar, R. Manjunath, S. Jaiswal, Facial emotion recognition using convolutional neural networks, Mater. Today: Proc. 80 (2023) 3560–3564, http://dx.doi.org/10.1016/j.matpr.2021.07.297,

SI:5 NANO 2021. URL https://www.sciencedirect.com/science/article/pii/S2214785321051567.