UNIVERSIDAD NACIONAL DE MOQUEGUA

INGENIERÍA DE SISTEMAS E INFORMÁTICA

TESIS

“RECONOCIMIENTO DE EXPRESIONES FACIALES Y LA POSTURA DE LA CABEZA USANDO MODELOS DE DEEP LEARNING”

PARA OBTENER EL TÍTULO PROFESIONAL DE INGENIERO DE SISTEMAS E INFORMÁTICA

ELABORADO POR:

DIEGO HEDER PERCCA MAMANI

GABRIELA KARINA VELASQUEZ QUISPE

ASESOR:

MTR. VICTOR YANA MAMANI

ILO - PERÚ

2024

UNAM
UNIVERSIDAD NACIONAL DE MOQUEGUA

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

Folio **Nº 012**

ACTA DE SUSTENTACIÓN

I. Modalidad:

Tesis	: [X]
Trabajo de Suficiencia Profesional	: [ ]
Otro:	: [ ]

II. Documento de aprobación para la sustentación:

Resolución de Facultad Nº 218-2024-FISI-UNAM

Fecha: 06 de Noviembre 2024

III. El jurado ha dictaminado la (el) Tesis, TITULADO:

Reconocimiento de Expresiones Faciales y La Postura de la Cabeza usando modelos de Deep learning

Bachiller: Diego Heder Percca Mamani

Asesor Principal: Victor Yana Mamani

Co-Asesor: —

IV. Se dictamina:

Como: Aprobado por Unanimidad la sustentación de manera:

Presencial [X] Virtual [ ]

Con nota: Diecisiete (17) por lo que; Sí [X] No [ ]

Opta el Título Profesional de INGENIERO DE SISTEMAS E INFORMÁTICA.

Siendo las 12:24 horas, del 19 de Nov. del año 2024 se procede con la firma en conformidad de lo dictaminado por el Jurado.

Presidente
Anibal Flores Garcia

Primer Miembro
Ender Tito Chura

Segundo Miembro
Carlos Silva Delgado

Diego Percca Mamani
Bachiller

UNAM

ESCUELA PROFESIONAL DE
INGENIERÍA DE SISTEMAS E INFORMÁTICA

Folio Nº 013

ACTA DE SUSTENTACIÓN

I. Modalidad:

Tesis : [X]
Trabajo de Suficiencia Profesional : [ ]
Otro: ..... : [ ]

II. Documento de aprobación para la sustentación:
Resolución de Facultad Nº 218-2024-FACIA-UNAM

Fecha: 06 de Noviembre 2024

III. El jurado ha dictaminado la (el) Tesis, TITULADO:
Reconocimiento de Expresiones Faciales y la Postura de la Cabeza usando modelos de Deep learning

Bachiller: Gabriela Karina Velazquez Quispe
Asesor Principal: Victor Yana Mamani
Co-Asesor: ——

IV. Se dictamina:
Como: Aprobado por Unanimidad la sustentación de manera:

Presencial [X] Virtual [ ]

Con nota: Diecisiete (17) por lo que; Sí [X] No [ ]

Opta el Título Profesional de INGENIERO DE SISTEMAS E INFORMÁTICA .

Siendo las 12:24 horas, del 19 de Nov del año 2024 se procede con la firma en conformidad de lo dictaminado por el Jurado.

Presidente
Amiel Flores Garcia

Primer Miembro
Ester Fito Chura

Segundo Miembro
Carlos Silva Delgado

Bachiller

AGRADECIMIENTOS

Queremos expresar nuestro más sincero agradecimiento a todas las personas que han contribuido de diversas maneras a la realización de esta tesis.

En primer lugar, queremos expresar nuestro más profundo agradecimiento a nuestras familias, por su incondicional apoyo emocional, su paciencia y comprensión, especialmente en los momentos en que no pudimos dedicarles el tiempo que merecían debido a las exigencias de la realización de esta tesis.

Un agradecimiento especial a nuestro asesor de tesis Mtr. Victor Yana Mamani, quien estuvo siempre dispuesto ante cualquier consulta, siendo nuestro soporte para la culminación de esta tesis.

Agradecemos también al Dr. Anibal Flores por sus clases en Machine Learning y seminario de tesis, que sirvieron de base para seguir investigando sobre esta área, además de formar parte de las primeras ideas para el tema de la tesis. Al Mtr. Hugo Euler Tito Chura, por las bases teóricas en Deep Learning. A Wilbur Naike Chiuyari Veramendi, por su compañía, guía y motivación en la investigación teórica de las arquitecturas CNN.

Finalmente, agradecer a todos nuestros profesores que de alguna manera colaboraron ya sea directa o indirectamente en nuestro aprendizaje, durante estos años de estudio.

A todos ustedes, muchas gracias.

RESUMEN

En la actualidad, la interacción humano computador (IHC) desempeña un papel crucial en diversas aplicaciones, como videoconferencias, e-learning, vigilancia y seguridad, robótica social y monitoreo de la atención de conductores. Dentro de este campo, la detección de la postura de la cabeza y las expresiones faciales es fundamental para que las máquinas comprendan mejor las emociones humanas, infieran el foco de atención y analicen el comportamiento. No obstante, la variabilidad, precisión y velocidad de los sistemas actuales representan desafíos que restringen su implementación en aplicaciones que exigen respuestas en tiempo real.

Este estudio tiene como objetivo principal reconocer las expresiones faciales y la postura de la cabeza mediante el uso de modelos de Deep Learning. Para ello, se utilizaron los conjuntos de datos FER-2013 y Biwi, que fueron divididos adecuadamente para el entrenamiento y evaluación de tres arquitecturas diferentes: AlexNet, DenseNet y ResNet18. En el proceso se aplicaron técnicas de ajuste de hiperparámetros, aprendizaje por transferencia (transfer learning) y ponderación de clases (class weights) para seleccionar el modelo más adecuado. Finalmente, los modelos seleccionados se integraron en una aplicación web de videoconferencia para realizar pruebas en tiempo real.

En cuanto a los resultados, para el conjunto de datos FER-2013 se obtuvieron los siguientes F1-scores: 67.17 % con DenseNet, 64.56 % con ResNet18 y 62.66 % con AlexNet. Para el conjunto de datos Biwi, los accuracy alcanzados fueron del 98.5 % para ResNet18, 98 % para DenseNet y 96.25 % para AlexNet. Aunque DenseNet obtuvo buenos resultados, su mayor complejidad y requisitos de memoria hicieron que ResNet18 fuera la opción más adecuada para la integración en la aplicación web de videoconferencia, donde se lograron resultados en tiempo real satisfactorios.

En conclusión, los modelos implementados demostraron ser efectivos para el reconocimiento de expresiones faciales y la postura de la cabeza. Sin embargo, aunque para el conjunto de datos Biwi se alcanzó una exactitud superior al 90 %, para FER-2013 el F1-score no superó el 70 %. Esto puede deberse a la falta de balance en las clases de FER-2013, la alta similitud de algunas imágenes que pertenecen a diferentes clases y el hecho de que las imágenes son en escala de grises y de baja resolución (48x48 píxeles). Además, aunque el sistema mostró buenos resultados en tiempo real, no fue completamente preciso en ciertos casos, por lo que se recomienda ampliar el conjunto de datos, incorporando mayor variabilidad en términos

de posiciones, fondos, tamaños e iluminación.

Palabras clave: Deep learning, redes neuronales convolucionales, reconocimiento de expresiones faciales, reconocimiento de la postura de la cabeza, interacción humano computador.

ABSTRACT

Currently, human computer interaction (HCI) plays a crucial role in a variety of applications, including video conferencing, e-learning, surveillance and security, social robotics, and driver attention monitoring. Within this field, head pose and facial expression detection are essential for machines to better understand human emotions, infer attention focus, and analyze behavior. However, challenges related to variability, accuracy, and speed in current systems limit their implementation in applications that require real-time responses.

The main objective of this study is to recognize facial expressions and head posture using deep learning models. To achieve this, the FER-2013 and Biwi datasets were utilized, divided appropriately for training and evaluating three different architectures: AlexNet, DenseNet, and ResNet18. Throughout the process, techniques such as hyperparameter tuning, transfer learning, and class weights were applied to identify the most suitable model. Finally, the selected models were integrated into a video conferencing web application for real-time testing.

As for the results, the following F1-scores were obtained for the FER-2013 dataset: 67.17 % with DenseNet, 64.56 % with ResNet18, and 62.66 % with AlexNet. For the Biwi dataset, the accuracy achieved was 98.5 % with ResNet18, 98 % with DenseNet, and 96.25 % with AlexNet. Although DenseNet produced good results, its higher complexity and memory requirements made ResNet18 the most suitable option for integration into a video conferencing web application, where satisfactory real-time performance was achieved.

In conclusion, the implemented models proved effective for facial expression and head posture recognition. However, while accuracy exceeding 90 % was achieved on the Biwi dataset, the F1-score for FER-2013 did not surpass 70 %. This discrepancy may be attributed to the unbalanced classes in FER-2013, the high similarity of some images belonging to different classes, and the fact that the images are grayscale with a low resolution of 48x48 pixels. Additionally, although the system performed well in real-time, it was not entirely accurate in certain cases, so it is recommended to expand the dataset, incorporating greater variability in terms of positions, backgrounds, sizes, and lighting conditions.

Keywords: Deep learning, convolutional neural networks, facial expression recognition, head pose recognition, human computer interaction.

ÍNDICE

LISTA DE TABLAS VII

LISTA DE FIGURAS VIII

LISTA DE SIGLAS IX

INTRODUCCIÓN 1

CAPÍTULO I: PLANTEAMIENTO DEL PROBLEMA 2

1.1 DESCRIPCIÓN Y FORMULACIÓN DEL PROBLEMA 2

1.1.1 Descripción del Problema 2

1.1.2 Formulación del Problema 2

1.2 OBJETIVOS 3

1.2.1 Objetivo General 3

1.2.2 Objetivos Específicos 3

1.3 JUSTIFICACIÓN E IMPORTANCIA 3

1.4 HIPÓTESIS DEL ESTUDIO 4

1.4.1 Hipótesis General 4

1.4.2 Hipótesis Específicas 4

1.5 VARIABLES 4

1.5.1 Variable Independiente 4

1.5.2 Variable Dependiente 4

CAPÍTULO II: MARCO TEÓRICO 5

2.1 ANTECEDENTES 5

2.2 BASES TEÓRICAS 7

2.2.1 Inteligencia artificial 7

2.2.2 Aprendizaje Automático (machine learning) 8

2.2.3 Aprendizaje Profundo (Deep Learning) 10

2.2.4 Redes Neuronales Convolucionales (CNN) 15

2.2.5 Factores que Influyen al Reconocimiento de la Atención . . . 23

2.3 DEFINICIÓN DE TÉRMINOS 25

CAPÍTULO III: METODOLOGÍA 28

3.1 TIPO Y NIVEL DE INVESTIGACIÓN 28

3.2 POBLACIÓN Y MUESTRA 28

3.2.1 Población 28
3.2.2 Muestra 28
3.3 INSTRUMENTOS 28
3.4 PROCEDIMIENTOS 30
3.4.1 Preparación de los Datos 30
3.4.2 División del Conjunto de Datos 32
3.4.3 Implementación de arquitecturas 32
3.4.4 Transfer Learning 32
3.4.5 Entrenamiento de los modelos 32
3.4.6 Evaluación de los modelos 34
3.4.7 Evaluación del funcionamiento de los modelos en tiempo real 34
3.5 ANÁLISIS DE DATOS 35
3.5.1 Reconocimiento de la postura de la cabeza 35
3.5.2 Reconocimiento de expresiones faciales 37

CAPÍTULO IV: RESULTADOS Y DISCUSIÓN 44

4.1 Reconocimiento de la postura de la cabeza 44
4.2 Reconocimiento de expresiones faciales 46
4.3 Prueba en tiempo real 48

CONCLUSIONES 50

RECOMENDACIONES 51

REFERENCIAS 52

ANEXOS 56

ANEXO 1: ESQUEMA DE RED PARA LA CONEXIÓN DE USUARIOS . . 56
ANEXO 2: DIAGRAMA DE FLUJO PARA PROCESAR Y CLASIFICAR LAS IMÁGENES OBTENIDAS DE LA CÁMARA WEB DEL USUARIO 57

LISTA DE TABLAS

Tabla N° 3.1 Arquitectura del modelo Resnet18 33
Tabla N° 3.2 Arquitectura del modelo DenseNet 33
Tabla N° 3.3 Arquitectura del modelo Alexnet 34
Tabla N° 3.4 Distribución de las imágenes del dataset Biwi 36
Tabla N° 3.5 Indicadores de modelos aplicadas a Biwi 38
Tabla N° 3.6 Distribución de las imágenes del dataset FER-2013 38
Tabla N° 3.7 Indicadores de modelos aplicadas a FER-2013 42
Tabla N° 4.1 Indicadores obtenidos por arquitectura, para dataset Biwi . 44
Tabla N° 4.2 Tamaño de los mejores modelos en formato ONNX, para el reconocimiento de la postura de la cabeza 44
Tabla N° 4.3 Indicadores obtenidos por arquitectura, para dataset FER-2013 46
Tabla N° 4.4 Tamaño de los mejores modelos en formato ONNX, para el reconocimiento de expresiones faciales 47

LISTA DE FIGURAS

Figura N° 2.1 Inteligencia Artificial, Machine Learning y Deep Learning . 10
Figura N° 2.2 Arquitectura de un Perceptrón Multicapa 11
Figura N° 2.3 Matriz de confusión binaria 13
Figura N° 2.4 Matriz de confusión multiclase (clase b) 13
Figura N° 2.5 Ejemplo de una convolución 2D 17
Figura N° 2.6 Ejemplo del cálculo de max-pooling y average pooling . . 18
Figura N° 2.7 Arquitectura del modelo LeNet-5 19
Figura N° 2.8 Arquitectura del modelo AlexNet 20
Figura N° 2.9 Arquitectura del modelo VGG16 20
Figura N° 2.10 Arquitectura Googlenet 21
Figura N° 2.11 Arquitectura de ResNet50, Resnet101 y Resnet152 . . . 22
Figura N° 2.12 Arquitectura resumida de DenseNet 22
Figura N° 2.13 Ejemplo de expresiones faciales en las imágenes de la base de datos CK+ 23
Figura N° 2.14 Grados de libertad de la cabeza humana 25
Figura N° 3.1 Distribución de los dataset 35
Figura N° 3.2 Prueba de hiperparámetros para Resnet18 - Biwi 37
Figura N° 3.3 Métricas para las arquitecturas propuestas - Biwi 39
Figura N° 3.4 Prueba de Hiperparámetros - DenseNet 40
Figura N° 3.5 Mejores modelos obtenidos - FER-2013 43
Figura N° 4.1 Matriz de confusión para el mejor modelo Resnet18 . . . 45
Figura N° 4.2 Etiquetas vs predicciones de algunas imágenes aleatorias de dataset Biwi (test) 46
Figura N° 4.3 Matriz de confusión para el mejor modelo DenseNet . . . 47
Figura N° 4.4 Etiquetas vs predicciones de algunas imágenes aleatorias del dataset FER-2013 (test) 48
Figura N° 4.5 Prueba de los modelos en tiempo real 49

LISTA DE SIGLAS

IA	:	Inteligencia Artificial
FER	:	Reconocimiento de Expresiones Faciales (Facial Expressions Recognition)
ML	:	Aprendizaje automático (Machine learning)
DL	:	Aprendizaje profundo (Deep learning)
MLP	:	Perceptrón multicapa (Multilayer Perceptron)
CNN	:	Red neuronal convolucional (Convolutional Neural Network)
RNN	:	Red neuronal recurrente (Recurrent Neural Network)
TLUs	:	Unidades lógicas de umbral (Threshold Logic Units)
SGD	:	Descenso de gradiente estocástico (Stochastic gradient descent)
ADAM	:	Estimación de momento adaptativa (Adaptive Moment Estimation)
ReLU	:	Unidad lineal rectificada (Rectified Linear Unit)
GPU	:	Unidad de procesamiento gráfico (Graphics Processing Unit)

INTRODUCCIÓN

Con el avance de las tecnologías de la información y la comunicación, la interacción humano computador (IHC) ha mejorado significativamente. El reconocimiento de expresiones faciales es crucial para que las computadoras comprendan las emociones humanas, siendo un componente esencial de la IHC (Shi et al., 2020). Además, la estimación de la postura de la cabeza es un área de investigación interesante, ya que permite inferir el foco de atención del sujeto (Wu y Trivedi, 2008). Al igual que el reconocimiento de voz, que se ha incorporado en diversas tecnologías, la estimación de la posición de la cabeza probablemente se convertirá en una herramienta estándar para acortar la brecha entre humanos y computadoras (Murphy-Chutorian y Trivedi, 2008).

Este estudio tiene como objetivo el reconocimiento de expresiones faciales y la postura de la cabeza mediante modelos de Deep Learning. Para ello, se evaluarán los modelos que obtengan el mejor F1-score en el reconocimiento de expresiones faciales y la mayor exactitud en la postura de la cabeza. Además, se implementará una aplicación de videoconferencia para probar en tiempo real los modelos seleccionados.

La metodología incluye la preparación y división de los conjuntos de datos FER-2013 y Biwi, la implementación de tres arquitecturas diferentes (AlexNet, DenseNet, ResNet18), y la evaluación de los modelos, utilizando técnicas como ajuste de hiperparámetros, transferencia de aprendizaje (transfer learning) y class weights para seleccionar el modelo más adecuado. Finalmente, los modelos serán integrados en una aplicación web de videoconferencia para pruebas en tiempo real.

El estudio reconoce limitaciones, como la calidad y el balance de los datos, así como la demanda de recursos computacionales necesarios para entrenar modelos profundos. A pesar de estas limitaciones, se espera que los resultados proporcionen una base sólida para futuras investigaciones en Deep Learning.

La tesis se estructura de la siguiente manera: El capítulo 1 describe y plantea el problema, el capítulo 2 revisa la literatura relevante, el capítulo 3 detalla la metodología utilizada, el capítulo 4 presenta y discute los resultados, el capítulo 5 concluye los hallazgos, y el capítulo 6 ofrece recomendaciones para investigaciones futuras.

CAPÍTULO I: PLANTEAMIENTO DEL PROBLEMA

1.1 DESCRIPCIÓN Y FORMULACIÓN DEL PROBLEMA

1.1.1 Descripción del Problema

En la actualidad, el reconocimiento de la postura de la cabeza y las expresiones faciales se ha vuelto crucial en numerosos sistemas de interacción humano computador, como videoconferencias, educación a distancia, vigilancia y seguridad, robótica social, y monitoreo de la atención de conductores. No obstante, lograr un reconocimiento preciso en entornos dinámicos plantea importantes desafíos. Entre estos, destacan la variabilidad en las posturas de la cabeza, la diversidad de expresiones faciales, los cambios en las condiciones de iluminación, y las limitaciones de los dispositivos con recursos computacionales limitados. A esto se suma la dificultad de equilibrar la precisión de los modelos de Deep Learning con el tiempo de respuesta, un aspecto clave en aplicaciones que requieren resultados en tiempo real.

El problema que esta investigación aborda es reconocer la postura de la cabeza y las expresiones faciales utilizando modelos de Deep Learning. Además, se evalúa el desempeño de estos modelos en una aplicación web de videoconferencia, permitiendo verificar los resultados obtenidos en un entorno interactivo.

1.1.2 Formulación del Problema

1.1.2.1 Problema General

- ¿Será posible reconocer las expresiones faciales y la postura de la cabeza usando modelos de Deep Learning?

1.1.2.2 Problemas Específicos

- ¿Qué modelo de Deep Learning logra el mejor F1-score en el reconocimiento de expresiones faciales?
- ¿Qué modelo de Deep Learning logra el mejor accuracy en el reconocimiento de la postura de la cabeza?
- ¿Es el modelo de Deep Learning seleccionado para el reconocimiento de expresiones faciales y postura de la cabeza efectivo en tiempo real?

1.2 OBJETIVOS

1.2.1 Objetivo General

Reconocer las expresiones faciales y la postura de la cabeza usando modelos de Deep Learning

1.2.2 Objetivos Específicos

- Determinar qué modelo de Deep Learning logra el mejor F1-score en el reconocimiento de expresiones faciales.
- Determinar qué modelo de Deep Learning alcanza el mejor accuracy en el reconocimiento de la postura de la cabeza.
- Probar el modelo de Deep Learning para el reconocimiento de expresiones faciales y la postura de la cabeza en tiempo real.

1.3 JUSTIFICACIÓN E IMPORTANCIA

La interacción entre humanos y máquinas ha experimentado una evolución significativa, demandando cada vez más sistemas capaces de interpretar de manera efectiva el comportamiento humano en tiempo real. La detección de la postura de la cabeza y las expresiones faciales es fundamental en la interacción humano computador, con aplicaciones en diversas áreas como videoconferencias, e-learning, vigilancia y seguridad, robótica social, y monitoreo de la atención en conductores. No obstante, los sistemas actuales suelen enfrentar limitaciones en términos de precisión y velocidad, lo que restringe su uso en aplicaciones críticas que requieren respuestas en tiempo real.

En este contexto, la postura de la cabeza se convierte en un componente clave para inferir la atención del usuario. Determinar si una persona está frente a la cámara o en una postura no frontal es esencial para los sistemas de monitoreo en tiempo real, ya que una postura no frontal puede interpretarse como falta de atención o desinterés.

Este trabajo propone reconocer tanto las expresiones faciales como la postura de la cabeza utilizando modelos de Deep Learning. Para llevar a cabo esta investigación, se implementará una aplicación de videoconferencia que permitirá probar en tiempo real los resultados del modelo seleccionado.

Así mismo, se contribuirá a investigaciones futuras sobre la atención del usuario mediante la implementación de modelos de Deep Learning para el reconocimiento de expresiones faciales y la posición de la cabeza. Además, permitirá evaluar la efectividad de los modelos de Deep Learning para aplicaciones en tiempo real.

1.4 HIPÓTESIS DEL ESTUDIO

1.4.1 Hipótesis General

Se reconoce las expresiones faciales y la postura de la cabeza usando modelos de Deep Learning.

1.4.2 Hipótesis Específicas

- El modelo de Deep Learning seleccionado para el reconocimiento de expresiones faciales logrará un F1-score superior al 70 %.
- El modelo de Deep Learning seleccionado para el reconocimiento de la postura de la cabeza alcanzará un accuracy mayor al 90 %.
- El modelo de Deep Learning para el reconocimiento de expresiones faciales y la postura de la cabeza demostrará ser efectivo en tiempo real.

1.5 VARIABLES

1.5.1 Variable Independiente

La variable independiente para la presente tesis, son los modelos de Deep Learning.

1.5.2 Variable Dependiente

Las variables dependientes para la presente tesis son la expresión facial y la postura de la cabeza.

CAPÍTULO II: MARCO TEÓRICO

2.1 ANTECEDENTES

(Singh y Nasoz, 2020) demostraron que es posible realizar la clasificación de expresiones faciales en imágenes estáticas utilizando Redes Neuronales Convolucionales (CNNs) sin la necesidad de preprocesamiento o extracción de características. Sin embargo, observaron que la incorporación de técnicas de preprocesamiento, como la detección de rostros y la corrección de la iluminación, mejora la exactitud del sistema. A pesar de los avances, la implementación de estrategias como max-pooling y dropout, para prevenir el sobreajuste y aumentar la exactitud, sigue siendo un desafío. En la tarea de clasificación en siete categorías, utilizando el conjunto de datos FER-2013, lograron una exactitud del 61.7 %, lo que subraya la necesidad de continuar investigando métodos más avanzados para alcanzar los resultados de última generación, que alcanzan una exactitud del 75.2 % (Pramerdorfer y Kampel, 2016).

El estudio realizado por (Behera et al., 2019) propone un nuevo enfoque para el reconocimiento de la postura de la cabeza utilizando CNNs que exploran tanto regiones locales como globales de la imagen. A diferencia de los métodos tradicionales que dependen de la localización precisa de puntos de referencia faciales y modelos de cabeza promedio para resolver problemas de correspondencia 2D-3D, este modelo busca mejorar la robustez del reconocimiento capturando cambios topológicos en la imagen derivados de variaciones en el punto de vista. La metodología empleada por los autores se basa en la adaptación de las CNNs existentes, dividiendo la imagen en múltiples regiones consecutivas inspiradas en el cálculo de descriptores HOG (Histogram of Oriented Gradients), lo que permite analizar diferentes partes de la cabeza de manera más precisa. Los resultados experimentales, obtenidos a través de cuatro conjuntos de datos a gran escala, mostraron que el modelo propuesto supera a varios métodos de última generación, incluidos modelos de CNNs tradicionales y la herramienta OpenFace 2.0. Además, los autores contribuyeron con anotaciones sobre la postura de la cabeza en el conjunto de datos VGGFace2, mejorando así los recursos disponibles para futuras investigaciones.

El estudio que realizaron (Ujir et al., 2021) propone un sistema de monitoreo del conductor en tiempo real que utiliza dispositivos móviles para detectar la postura de la cabeza, así como la somnolencia y expresiones de enfado del conductor. La metodología empleada incluye el uso de la cámara del teléfono para recopilar datos de las expresiones faciales, que luego son analizados a través de técnicas de

procesamiento de imágenes en la plataforma Microsoft Azure. Se desarrolló una aplicación móvil que no solo identifica las emociones del conductor, sino que también proporciona alertas a través de mensajes emergentes y retroalimentación de audio. Los resultados obtenidos indican que la aplicación tiene un alto rendimiento, logrando una precisión del 96.66 % en la detección de la emoción de enfado y del 82.2 % para la somnolencia, además de una tasa de éxito del 85.67 % en la detección de la postura de la cabeza en diversos escenarios. Estos hallazgos sugieren que la implementación de esta tecnología podría mejorar significativamente la seguridad vial al ayudar a los conductores a manejar sus emociones y mantener una conducción más segura.

El estudio realizado por (Yolcu et al., 2020) propone un sistema de análisis facial basado en aprendizaje profundo para monitorear el interés del cliente, específicamente detectando su comportamiento hacia productos y anuncios. La metodología utilizada incluye la estimación de la postura de la cabeza del cliente para medir su atención, seguida de un análisis detallado de las expresiones faciales para evaluar su interés. Este proceso comienza con la detección de posturas frontales, donde se segmentan componentes faciales importantes para generar una imagen iconizada. Posteriormente, se analizan las expresiones faciales combinando los valores de confianza de la imagen iconizada con las imágenes faciales originales. Los autores lograron implementar este sistema utilizando dispositivos de imagen comunes, como cámaras web, obteniendo resultados de estado del arte en la clasificación del interés del cliente y permitiendo monitorear la respuesta emocional a diversos estímulos en entornos de marketing

(T S y Guddeti, 2018) proponen un sistema no intrusivo capaz de reconocer a cada estudiante, analizar su nivel de atención y clasificar su estado emocional utilizando cámaras de video vigilancia. La metodología que plantean consiste en obtener fotogramas de las cámaras de video vigilancia, que son la entrada para la arquitectura You Only Look Once (YOLO), el cual se usa para obtener las coordenadas del cuadro delimitador de cada estudiante presente en un fotograma. Posteriormente los fotogramas son recortados de acuerdo con las coordenadas de cada cuadro delimitador y son enviados al modelo Inception V3 para clasificar la atención del estudiante. También estos fotogramas recortados son usados para el reconocimiento de la emoción y la identificación de cada estudiante mediante el uso de una arquitectura CNN. El dataset se obtuvo de 243 estudiantes de diferentes cursos de laboratorio de ciencias de la computación y tecnologías de la información del National Institute of Technology Karnataka Surathkal de India, utilizando 2 cámaras de video vigilancia para cada laboratorio, los datos de entrenamiento se recolectaron por 5 horas y 43 minutos y fueron etiquetados manualmente bajo la supervisión

de un experto. El accuracy que se obtiene de ambos modelos YOLO y CNN son elevados, ya que usan pesos pre entrenados, obteniendo un 0.94 % en la detección y 0.89 % en la clasificación. El análisis de atención muestra que existe una correlación positiva +0.8976 entre el nivel de atención y la calificación obtenida en pequeñas tareas hechas por los estudiantes.

(Monkaresi et al., 2017) Este artículo, tiene como objetivo la detección automática de la atención de los estudiantes mientras realizan actividades de escritura estructurada, basándose en expresiones faciales y la frecuencia cardiaca, utilizando métodos de visión computacional. La metodología utilizada consiste en la extracción de características de cada segmento de video, para lo cual utilizan tres tipos de extractores, frecuencia cardiaca (Heart Rate – HR), unidades de animación (de Kinect face tracker) y patrones binarios locales en tres planos ortogonales (LBP-TOP). Los estudiantes realizaron informes de su atención por cada dos minutos simultáneamente durante la actividad de escritura y retrospectivamente en base a la grabación de videos de sus rostros después de la actividad. Finalmente aplicaron técnicas de aprendizaje automático para la clasificación de la atención de los estudiantes. Para evaluar la precisión del clasificador se utilizó el área bajo la curva ROC (AUC). Logrando un AUC = 0.758 para anotaciones concurrentes y un AUC = 0.733 para anotaciones retrospectivas.

(Keerthana et al., 2020) proponen un sistema de arquitectura híbrida para el análisis dinámico de la participación del estudiante, implementada en tiempo real para generar dos categorías: concentrado o distraído. Esta arquitectura involucra la combinación de 3 módulos: El reconocimiento de emociones faciales mediante un algoritmo de extracción de características aplicando CNN, monitoreo de la mirada con Haar Cascade para la detección de pupilas, e identificación de movimiento de la cabeza mediante patrones binarios locales. La arquitectura fue hecha en Python usando OpenCV para la detección del rostro y convertir las imágenes a escala de grises. El modelo propuesto generó un accuracy del 63,8 %.

2.2 BASES TEÓRICAS

2.2.1 Inteligencia artificial

Existen múltiples definiciones para la IA, ya que, desde sus inicios hasta la actualidad, no se ha establecido una definición clara y única. En (Rouhiainen, 2018) se define la IA como la capacidad de las máquinas para llevar a cabo tareas que imitan la inteligencia humana. Por ejemplo, a través de algoritmos, las máquinas pueden aprender de grandes volúmenes de datos y tomar decisiones basadas en

ese aprendizaje.

La IA se dedica a abordar una gran variedad de problemas, pudiendo aplicarse en casi todas las situaciones. Algunos campos en los que las técnicas de IA han mostrado resultados prometedores incluyen (García, 2012):

- Minería de datos
- Reconocimiento del habla
- Reconocimiento de la escritura
- Visión por computadora
- Diagnóstico médico
- Robótica
- Entornos industriales
- Planificación y control
- Videojuegos
- Seguridad
- Predicciones financieras
- Predicción de desastres
- Marketing especializado

2.2.2 Aprendizaje Automático (machine learning)

Según (Géron, 2022) el aprendizaje automático es la ciencia (y también el arte) de programar computadoras para que puedan aprender a partir de datos.

Desde un punto de vista más general también se puede definir al aprendizaje automático como el campo de estudio que otorga a las computadoras la capacidad de aprender sin ser explícitamente programadas (Samuel, 1959).

El aprendizaje automático es importante para las organizaciones, ya que se ha utilizado en varios sectores, como la medicina, el sector financiero y muchos otros. Los modelos se emplean de diversas maneras; sin embargo, el procedimiento para entrenar y utilizar un modelo de aprendizaje automático varía según la tarea y los datos disponibles, como se muestra a continuación:

2.2.2.1 Aprendizaje supervisado

El aprendizaje supervisado es un enfoque del aprendizaje automático donde el conjunto de entrenamiento que se proporciona al algoritmo incluye las soluciones deseadas, llamadas etiquetas (Géron, 2022).

El aprendizaje supervisado requiere supervisión humana para etiquetar con precisión los datos, debido a que la mayoría de los datos disponibles están sin etiquetar (datos brutos), y la interacción humana es necesaria para crear grandes conjuntos de datos etiquetados, lo cual puede ser muy costoso.

El objetivo del aprendizaje supervisado es aprender patrones en los datos para poder predecir la etiqueta que corresponde a las características de un dato nuevo (Hurwitz y Kirsch, 2018).

2.2.2.2 Aprendizaje no supervisado

A diferencia del aprendizaje supervisado, el aprendizaje no supervisado trabaja con datos que no están etiquetados. Esto significa que el modelo no tiene ejemplos previos con respuestas conocidas que pueda usar para aprender. En el aprendizaje no supervisado, el sistema intenta identificar patrones, estructuras o relaciones en los datos sin guía explícita. No hay un “maestro” que proporcione las respuestas correctas, por lo que el modelo debe descubrir estos patrones por sí mismo (Géron, 2022).

El objetivo del aprendizaje no supervisado es explorar los datos para encontrar patrones relacionados. Los métodos más importantes en este enfoque son la agrupación (clustering), la detección de anomalías y novedades, la visualización y reducción de dimensionalidad, y el aprendizaje de reglas de asociación (Géron, 2022). Estos métodos permiten identificar patrones, segmentar datos en grupos o simplificar datos complejos.

2.2.2.3 Aprendizaje por refuerzo (Reinforcement Learning)

Aprendizaje por refuerzo es un método en el que un sistema de aprendizaje, denominado agente, interactúa con un entorno. El agente observa el entorno, selecciona y realiza acciones, y recibe recompensas (o penalizaciones, que son recompensas negativas) en respuesta a sus acciones (Géron, 2022).

El objetivo en el aprendizaje por refuerzo es encontrar una política que maximice

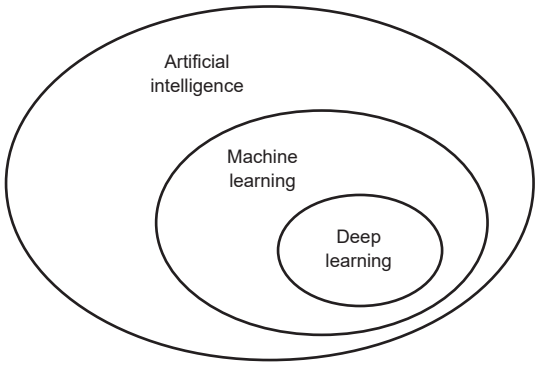
la recompensa acumulativa total. Esto significa que el agente debe aprender qué acciones son las mejores en cada situación para obtener la mayor recompensa posible a lo largo del tiempo. Es utilizado en juegos, robótica y control autónomo.

2.2.3 Aprendizaje Profundo (Deep Learning)

Es un enfoque avanzado de aprendizaje automático que utiliza redes neuronales profundas compuestas por múltiples capas jerárquicas para aprender representaciones de datos cada vez más complejas. Esta técnica, que ha evolucionado significativamente gracias a avances en matemáticas, hardware y grandes conjuntos de datos, se ha convertido en una tecnología fundamental en el campo de la inteligencia artificial (Bhardwaj et al., 2018).

Como se muestra en la Figura N° 2.1 el deep learning es un subcampo específico del aprendizaje automático que se enfoca en aprender representaciones sucesivas de datos mediante el uso de múltiples capas de representaciones. La “profundidad” en deep learning no se refiere a una comprensión más profunda en un sentido literal, sino a la idea de utilizar capas sucesivas para construir representaciones cada vez más significativas. La profundidad del modelo está relacionada con el número de capas que contribuyen a la representación de los datos (Chollet, 2018).

FIGURA N° 2.1: Inteligencia Artificial, Machine Learning y Deep Learning.



Nota: Tomado de (Chollet, 2018)

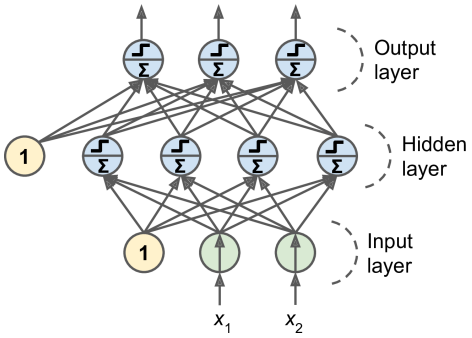
A continuación, se muestran tres técnicas populares de Deep Learning: El Perceptrón Multicapa (MLP), las Redes Neuronales Convolucionales (CNN) y las Redes Neuronales Recurrentes (RNN).

a) Perceptrón Multicapa (MLP)

   Es un tipo de red neuronal artificial que consiste en una capa de entrada, una o más capas ocultas de unidades lógicas de umbral (TLUs), y una capa de

salida. Las capas cercanas a la entrada se conocen como capas inferiores, mientras que las capas cercanas a la salida se denominan capas superiores. Como se observa en Figura N° 2.2 todas las capas, excepto la capa de salida, incluyen una neurona de sesgo y están completamente conectadas a la siguiente capa. Este diseño permite que el MLP procese datos y aprenda patrones complejos mediante el ajuste de los pesos sinápticos durante el entrenamiento (Géron, 2022).

FIGURA N° 2.2: Arquitectura de un Perceptrón Multicapa.



Nota: Tomado de (Géron, 2022)

b) Redes Neuronales Convolucionales (CNN)

Las redes neuronales convolucionales (CNN) se inspiran en la corteza visual del cerebro, especializada en procesar información visual. Las CNN imitan esta estructura al utilizar operaciones de convolución, procesando la entrada de manera uniforme. Aunque las CNN se utilizan principalmente en aplicaciones visuales, también son útiles en aplicaciones lingüísticas y en la resolución de problemas complejos. Algunos ejemplos incluyen aprender a jugar Go, logrando altas tasas de victoria, y la verificación facial en el sistema DeepFace de Facebook (Hearty, 2016).

Los desafíos de reconocimiento de objetos, como el ImageNet de 2014, han sido dominados por modelos que usan CNN. Estos modelos han mejorado gracias al desarrollo de nuevas arquitecturas como ResNet, AlexNet, entre otras.

c) Redes Neuronales Recurrentes (RNN)

Son un tipo de red neuronal que se especializa en procesar secuencias de datos de longitud arbitraria, a diferencia de las redes neuronales tradicionales que manejan entradas de tamaño fijo. Esto las hace especialmente útiles para tareas de predicción de series temporales, procesamiento de lenguaje natural (como traducción automática o conversión de voz a texto) y sistemas de conducción autónoma, entre otros.

A diferencia de las redes neuronales feedforward, donde las activaciones fluyen solo en una dirección (de la capa de entrada a la capa de salida), las RNN tienen conexiones que apuntan hacia atrás. Esto significa que una neurona en una RNN no solo recibe las entradas actuales, sino también su propia salida del paso de tiempo anterior (Géron, 2022).

2.2.3.1 Evaluación de efectividad de modelos de clasificación

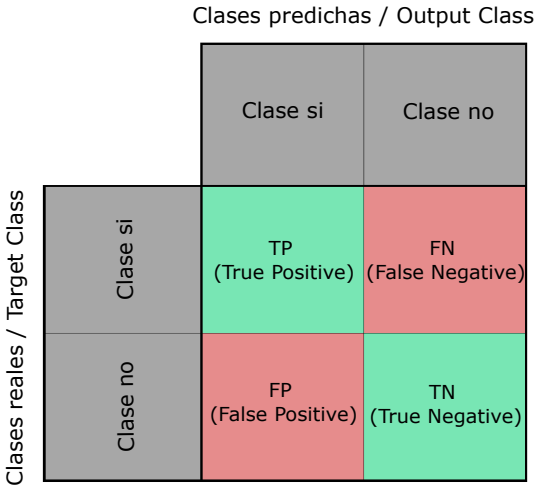
La evaluación de la efectividad de un modelo de Deep Learning es un proceso crítico para determinar su capacidad de realizar predicciones exactas en datos no vistos previamente (conjunto de prueba). Una vez que la red neuronal ha sido entrenada, es esencial evaluarla de manera adecuada para comprender su rendimiento real. Evaluar el modelo únicamente en los datos de entrenamiento no proporciona una indicación confiable de su efectividad, ya que el modelo ha sido ajustado específicamente a estos datos y podría haber memorizado patrones en lugar de aprender a generalizar (Brownlee, 2019).

El proceso de evaluación del modelo incluye el cálculo de varias métricas de rendimiento en el conjunto de prueba; tales como la exactitud (accuracy), que mide la proporción de predicciones correctas; el F1 Score; la sensibilidad o exhaustividad (recall); la precisión o especificidad (precision); el área bajo la curva ROC (AUC-ROC); entre otras.

Estas métricas ofrecen una visión integral del desempeño del modelo, lo cual facilita la identificación de posibles problemas como el sobreajuste (cuando el modelo rinde muy bien en los datos de entrenamiento, pero tiene un desempeño deficiente en los datos de prueba) o el subajuste (cuando el modelo no rinde adecuadamente ni siquiera en los datos de entrenamiento) (Hernández et al., 2022).

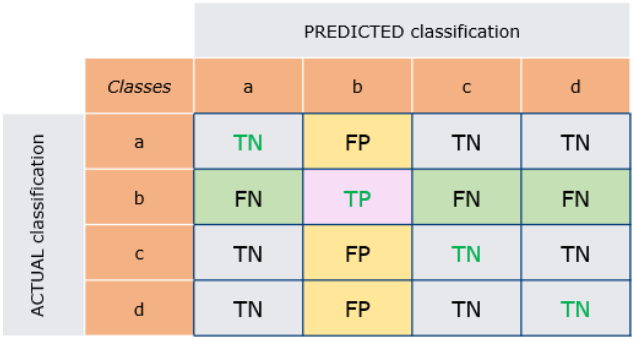
Para comprender mejor el funcionamiento de estas métricas, se utiliza la matriz de confusión, que proporciona información detallada sobre los tipos de errores que el clasificador puede cometer y ayuda a identificar patrones específicos de fallos. Así, la matriz de confusión es una herramienta útil para evaluar la precisión del clasificador (Hernández et al., 2022).

FIGURA N° 2.3: Matriz de confusión binaria.



Nota: Adaptado de (Visa et al., 2011)

FIGURA N° 2.4: Matriz de confusión multiclase (clase b).



Nota: Adaptado de (Grandini et al., 2020)

Donde:

TP : Número de predicciones positivas correctas.

FN : Número de predicciones negativas incorrectas.

FP : Número de predicciones positivas incorrectas.

TN : Número de predicciones negativas correctas.

Utilizando los valores de la matriz de confusión, podemos calcular métricas de rendimiento, como las siguientes:

- Exactitud (accuracy)

  La exactitud es una métrica ampliamente utilizada para evaluar el rendimiento de un modelo de clasificación. Se define como la proporción de predicciones

verdaderas (tanto positivas como negativas) sobre el número total de predicciones (Contreras et al., 2020). En otras palabras, la exactitud mide cuán bien un modelo clasifica correctamente las muestras en comparación con el total de muestras evaluadas. Sin embargo, se debe tener en cuenta que esta métrica puede resultar engañosa para conjuntos de datos con clases desequilibradas (Terven et al., 2023).

La fórmula del accuracy para clasificación binaria es la siguiente:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.1}$$

La fórmula del accuracy para clasificación multiclase es la siguiente:

$$Accuracy = \frac{Número\,de\,predicciones\,correctas}{Total\,de\,muestras} \tag{2.2}$$

- Precisión (precision)

  Evalúa las predicciones positivas realizadas por un modelo. Se define como el número de predicciones positivas verdaderas dividido por el número total de predicciones positivas (verdaderos positivos y falsos positivos). La precisión es crítica donde las consecuencias de los falsos positivos son significativas, ya que una alta precisión indica que el modelo produce pocos falsos positivos (Terven et al., 2023).

  La fórmula de precision para clasificación binaria es la siguiente:

$$Precision = \frac{TP}{TP + FP} \tag{2.3}$$

  La fórmula de precision multiclase, usando la variante macro es la siguiente:

$$Precision_{macro} = \frac{1}{C} \sum_{i=1}^{C} Precision_i \tag{2.4}$$

  Donde C es el número total de clases.

- Exhaustividad (recall)

  El recall mide la capacidad de un modelo para identificar correctamente muestras positivas entre todos los positivos reales en el conjunto de datos. Se define como la relación entre los verdaderos positivos y la suma de los verdaderos positivos y los falsos negativos. Un alto recall indica que el modelo logra capturar una gran proporción de casos positivos, siendo importante en aplicaciones en las que no detectar casos positivos podría tener consecuencias graves (Terven et al., 2023).

La fórmula de recall para clasificación binaria es la siguiente:

$$Recall = \frac{TP}{TP + FN} \tag{2.5}$$

La fórmula de recall multiclase, usando la variante macro es la siguiente:

$$Recall_{macro} = \frac{1}{C} \sum_{i=1}^{C} Recall_i \tag{2.6}$$

Donde C es el número total de clases.

- F1 Score

  El F1-score significa la media armónica entre precisión y recall (Dalianis, 2018). Una F1 Score más alto indica un mejor equilibrio entre precisión y recuperación, siendo útil en escenarios con distribuciones de clases desequilibradas, donde una clase puede dominar sobre otras (Terven et al., 2023).

  La fórmula de F1-score para clasificación binaria es la siguiente:

$$F1Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{2.7}$$

  La fórmula de F1-score multiclase, usando la variante macro es la siguiente:

$$F1Score_{macro} = \frac{1}{C} \sum_{i=1}^{C} F1Score_i \tag{2.8}$$

  Donde C es el número total de clases.

En problemas multiclase, existen 2 variantes principales para calcular la precision, recall y f1-score: Macro y micro (Grandini et al., 2020). En el presente trabajo, se utiliza la variante macro para evaluar el desempeño de los modelos para el reconocimiento de expresiones faciales, ya que permite comparar el rendimiento del modelo para cada clase de manera equitativa, dando también importancia a las clases minoritarias.

2.2.4 Redes Neuronales Convolucionales (CNN)

Las CNN son algoritmos que procesan imágenes y detectan varias características dentro de ellas para poder diferenciar unas imágenes de otras, pudiendo así clasificar imágenes, es decir identificar a qué grupo pertenece una imagen. (Moreno, 2020)

Las CNN, son un tipo de red neuronal inspirada en el neocórtex, la parte del cerebro responsable de las funciones más complejas en los humanos, específicamente en la manera en que la corteza visual procesa la información. Esta corteza tiene diferentes capas que procesan la información visual, extrayendo características cada vez más complejas a partir de las capas anteriores. Luego, esta información se envía a otras áreas del cerebro para un procesamiento más avanzado y la toma de decisiones. Este sistema es la base que usan los algoritmos de los CNN. (Gegúndez y Pérez, 2021)

Las CNN se han utilizado en una gran variedad de problemas y aplicaciones y ajustando su arquitectura, pueden aplicarse no solo a la clasificación de imágenes, sino también a la segmentación y detección de objetos en imágenes. Además, también se usan para procesar audio y video, convirtiéndolo en una herramienta clave en el área del deep learning.

2.2.4.1 Topología de las (CNN)

La topología de una CNN se distingue de las redes neuronales tradicionales, el incluir distintos tipos de capas y que la dimensión de los datos adquiere una importancia significativa en el procesamiento. Por ejemplo, cuando los datos tienen dos dimensiones, como una imagen en escala de grises, se representan matricialmente y se transforman en un vector para ser procesados por la red neuronal. Y si fueran datos tridimensionales, como imágenes a color, o con más dimensiones, también se transforman en un vector unidimensional. Sin embargo, en las CNN, las dimensiones de los datos pueden respetarse o modificarse, por lo que es crucial conocer siempre la estructura de los datos para aplicar las operaciones adecuadamente. (Gegúndez y Pérez, 2021)

A continuación se nombran las principales capas de una CNN común:

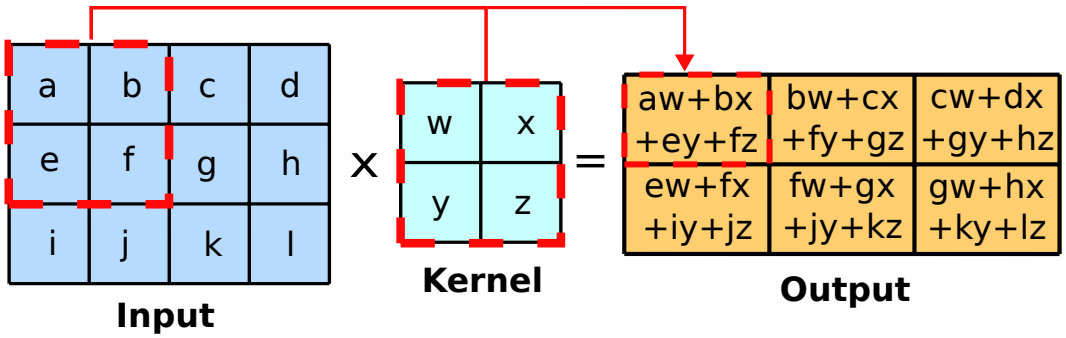
a) Capas de convolución (convolution layers)

   La capa de convolución es una de las más cruciales en una CNN. Esta capa permite extraer características de los datos de entrada. Para lograrlo, se utilizan filtros, también conocidos como kernels, para buscar y detectar características. Estos filtros son matrices que se aplican a un área de igual tamaño en la imagen que se desplazan por toda la imagen, de izquierda a derecha y de arriba a abajo, calculando los valores de salida de la capa convolucionada. Para ello, se calcula el producto escalar y se suman los resultados para formar la nueva capa convolucionada (Ver Figura N° 2.5). Por ejemplo, para una primera convolución, si se escogen 10 filtros, se obtendrían 10 matrices

de salida.

Un aspecto fundamental al trabajar con capas de convolución es el uso de funciones de activación, para potenciar la capacidad de la red en las capas de convolución. Esto se realiza luego de generar el mapa de características de salida, donde cada uno de sus elementos se somete a la función de activación y se actualiza su valor. La función más usada, ya que tiene un buen desempeño para imágenes es la función de no linealidad ReLU, el cual anula los negativos. (Moreno, 2020)

FIGURA N° 2.5: Ejemplo de operación de una convolución 2D.



Nota: Adaptado de (Bosch et al., 2019)

b) Capas de agrupación (pooling layers)

Se usan para reducir la dimensionalidad de las capas convolucionales resultantes, reteniendo la información más relevante. La operación se realiza sobre cada región (según el tamaño elegido) de la imagen que va recorriendo de izquierda a derecha y de arriba hacia abajo. (Gegúndez y Pérez, 2021)

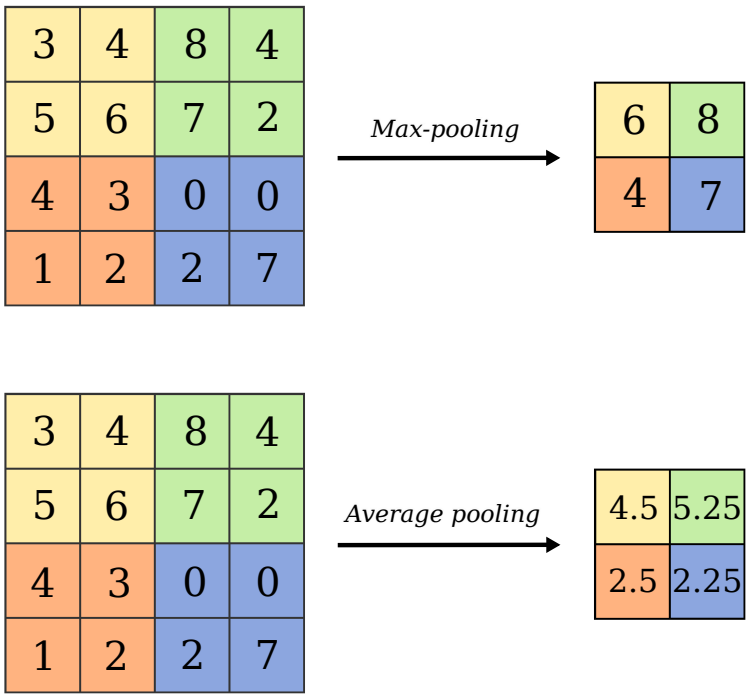
Esta operación se calcula según elección, de las cuales las más conocidas son (Ver Figura N° 2.6):

- Max-pooling: Máximo valor de la región analizada.
- Average pooling: El promedio de la región analizada.

c) Capas de salida (The output layer)

Es la capa de clasificación llamada Fully Connected layer, la cual analiza las características de las capas de convolución finales, para relacionarlas con una clase en particular. Para ello, convierte la salida de las capas convolucionales en un vector con el número de clases a clasificar como dimensión, donde cada elemento de este vector indica la probabilidad de pertenecer a la clase. (Moreno, 2020)

FIGURA N° 2.6: Ejemplo del cálculo de max-pooling y average pooling.



Nota: Adaptado de (Gegúndez y Pérez, 2021)

2.2.4.2 Arquitecturas de Convolutional Neural Networks (CNN)

Algunas arquitecturas estándar fueron diseñadas para competiciones exigentes, habiendo demostrado su eficacia, por lo que se puede ahorrar tiempo al utilizar una de estas arquitecturas pre-diseñadas en lugar de crear una desde cero mediante prueba y error.

El transfer learning por ejemplo, permite usar arquitecturas previamente entrenadas y ajustar solo la capa final para un nuevo problema específico, o utilizarlas como punto de partida para un nuevo entrenamiento.

A continuación, se mencionan algunas arquitecturas conocidas por su impacto histórico, resultados y novedades, las cuales se incluyen en casi todas las librerías de deep learning. (Gegúndez y Pérez, 2021)

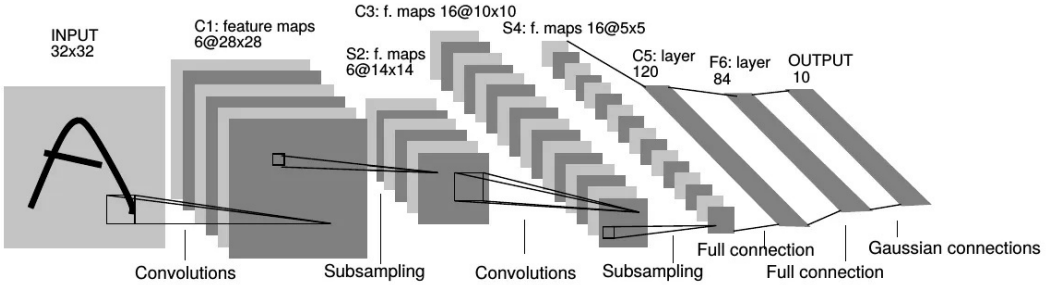
a) LeNet-5

   LeNet-5 es reconocida como la primera red neuronal convolucional de la historia, presentada en 1998 por Yann LeCun y su equipo. Esta red fue desarrollada para reconocer dígitos escritos a mano en el servicio postal estadounidense. (Gegúndez y Pérez, 2021)

Esta arquitectura es ideal para aprender y realizar pruebas iniciales con CNN, ya que es pequeña y puede ejecutarse incluso sin GPU. Como se muestra en la Figura N° 2.7, fue entrenada con imágenes en escala de grises de 32 × 32 × 1 y consta de 2 capas convolucionales con sus respectivos pooling y 3 capas completamente conectadas, donde la última usa la función Softmax. Aunque la función de activación más usada es ReLU, para esta arquitectura se usó tanh, que en su tiempo de creación fueron uno de las más usadas.

Esta arquitectura contiene aproximadamente 60,000 parámetros, comparado con las redes modernas que manejan entre 10 y 100 millones de parámetros. (Bosch et al., 2019)

FIGURA N° 2.7: Arquitectura del modelo LeNet-5.



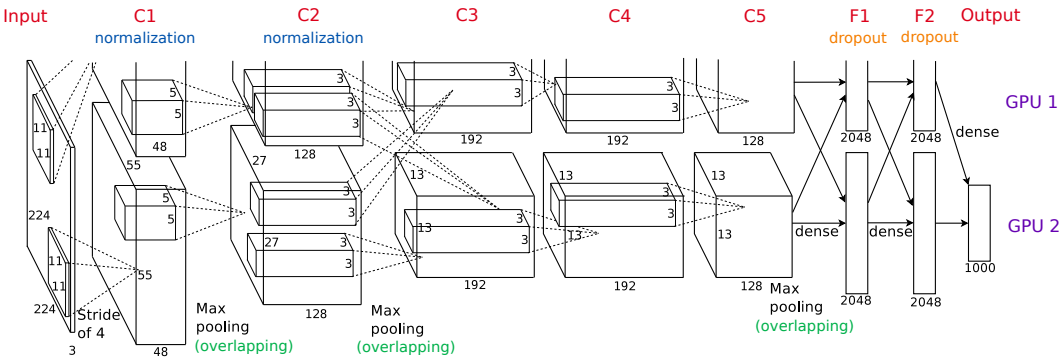
Nota: Tomado de (Lecun et al., 1998)

b) AlexNet

AlexNet fue la primera CNN en ganar la competición ImageNet, marcando el inicio de la popularidad del deep learning para problemas de procesamiento de imágenes. Presentada en 2012 por Alex Krizhevsky bajo la supervisión de Geoffrey Hinton. (Gegúndez y Pérez, 2021)

Esta arquitectura es más grande que la de LeNet-5, compuesta por 5 capas de convolución, 3 capas totalmente conectadas y 60 millones de parámetros. A la primera, segunda y quinta capa convolucional se le aplicaron una capa de max-pooling. En esta arquitectura se usó la función de activación ReLU después de cada capa de convolución y cada capa totalmente conectada. Además se incluyó una función de normalización en la primera y segunda capa de convolución. Adicionalmente, usaron 2 técnicas para evitar el sobre-ajuste: Data augmentation y Dropout. Como en esos tiempos las GPUs eran un poco lentas, su entrenamiento se realizó en 2 GPUs, tal como se muestra en la Figura N° 2.8, donde se ve la arquitectura dividida en 2 partes, con el objetivo de mejorar el tiempo de entrenamiento. (Moreno, 2020)

FIGURA N° 2.8: Arquitectura del modelo AlexNet.

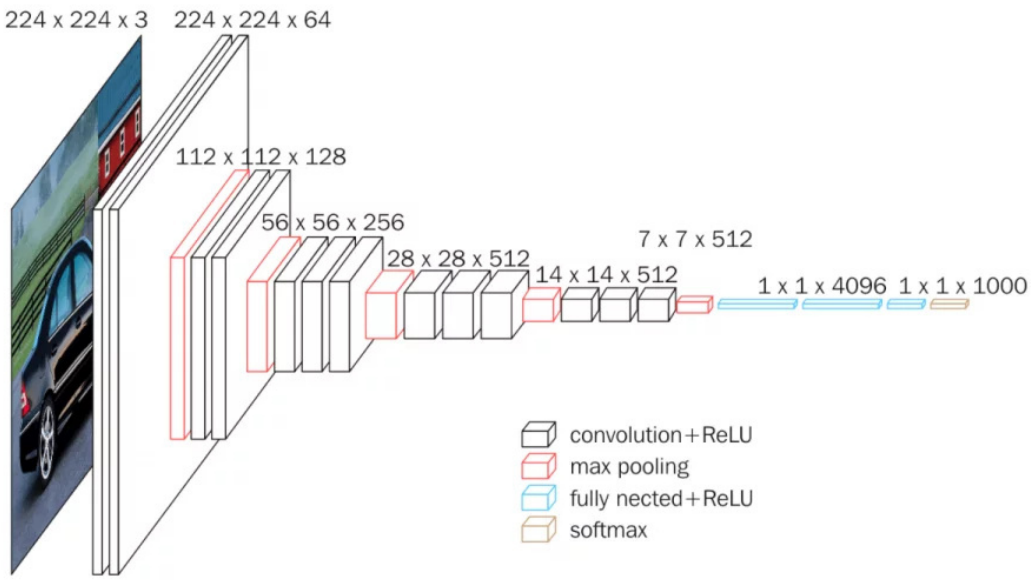


Nota: Adaptado de (Krizhevsky et al., 2012)

c) VGG

Para esta arquitectura se usó una red muy simple y uniforme, pero bastante profunda con un total de 138 millones de parámetros. Se dice que es una arquitectura uniforme porque usan bloques de capas convolucionales de 3x3 y stride $S = 1$ (cada una seguida de la función ReLU), luego se le aplica max-pooling de 2x2 y $S = 2$, y finalmente, terminan con capas completamente conectadas. Además, el número de filtros aplicados son 64, 128, 256 y 512, siendo valores duplicados en cada aumento de filtros. En la Figura N° 2.9 se muestra un ejemplo de una arquitectura vgg16, donde se tiene 13 capas de convolución y 3 capas completamente conectadas. Sin embargo, existe otra versión con 19 capas, llamada vgg19. (Bosch et al., 2019)

FIGURA N° 2.9: Arquitectura del modelo VGG16.

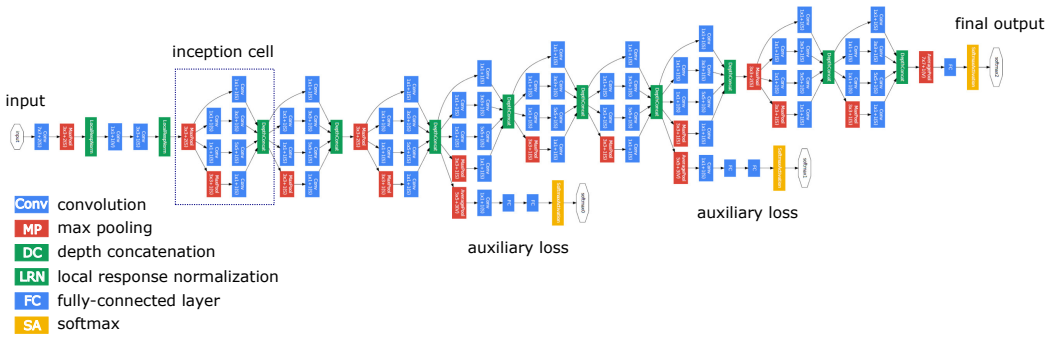


Nota: Tomado de (Krizhevsky et al., 2012)

d) GoogLeNet

Esta arquitectura se diferencia de las anteriores, que además de ser todavía mucho más profunda, incluye la paralelización de procesos mediante módulos Inception. Estos módulos permiten realizar varias operaciones en paralelo, eligiendo distintos tamaños de filtro, diferentes convoluciones y diferentes pooling, logrando extraer diferentes tipos de características de la imagen. Además, elimina las capas totalmente conectadas, permitiendo convertir cada mapa de características de 7x7x1024 a uno de 1x1x1024. En la Figura N° 2.10 se puede ver esta arquitectura que consta de 9 módulos Inception.(Moreno, 2020)

FIGURA N° 2.10: Arquitectura Googlenet.



Nota: Tomado de (Szegedy et al., 2014)

e) ResNet

En 2015, Microsoft presentó esta arquitectura basada en bloques residuales, la cual ganó la competición Imagenet con un error del 3.57 %, siendo la primera red en superar el error humano considerado entre el 5 % y 10 %. (Moreno, 2020)

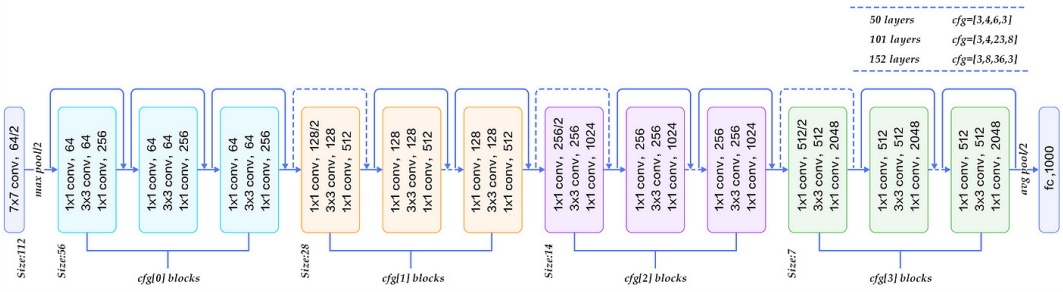
La idea de esta arquitectura surgió debido a que al apilar muchas capas, se llega a un punto donde la red ya no mejora, debido a la desaparición y explosión del gradiente. Es así que el uso de bloques residuales permite a la red saltarse algunas capas, permitiendo además, ignorar el gradiente de algunas capas durante la retropropagación (backpropagation), es decir reenviar el gradiente desde capas superiores a capas inferiores. Por ejemplo, para un caso extremo, la red podría reenviar directamente el gradiente desde la última capa hasta la primera, evitando hacer cálculos en todas las capas.

Para entender este flujo, supongamos que la entrada de un bloque residual es X y el resultado de este bloque residual es $F(x)$, entonces el resultado de la entrada para el siguiente bloque residual será $H(x) = F(x) + X$, y si

igualamos $F(x) = 0$, el resultado sería $H(x) = X$, logrando saltar un bloque residual. (Bosch et al., 2019)

Esta arquitectura presentó diferentes variantes en su profundidad de la red: Resnet18, Resnet50, Resnet101 y Resnet152, donde el número indica la cantidad de bloques incluidas en la red. En la Figura N° 2.11, se muestra la arquitectura de Resnet, donde se indica el número de bloques para Resnet50, Resnet101 y Resnet152.

FIGURA N° 2.11: Arquitectura de ResNet50, Resnet101 y Resnet152.



Nota: Tomado de (Gegúndez y Pérez, 2021)

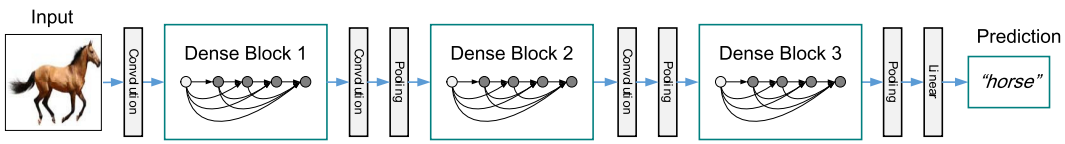
f) DenseNet

En 2016, esta arquitectura ganó la competición Imagenet, basada en bloques densos, los cuales preservan la información de capas anteriores, logrando superar a Resnet.

En la Figura N° 2.12, se muestra los bloques densos dentro de esta red, los cuales consisten en que el mapa de características de cada capa se concatena a la entrada de cada capa posterior dentro de un bloque denso. Esto permite que cada capa aproveche directamente las características de todas las capas anteriores a este. Los autores afirman que "la concatenación de mapas de características aprendidos por diferentes capas aumenta la variación en la entrada de las capas posteriores y mejora la eficiencia" (Huang et al., 2018)

Sin embargo, debemos tener en cuenta que al concatenar el mapa de características de varias capas, requerirá de más capacidad de procesamiento.

FIGURA N° 2.12: Arquitectura resumida de DenseNet.



Tomado de (Huang et al., 2018)

2.2.5 Factores que Influyen al Reconocimiento de la Atención

2.2.5.1 Reconocimiento de expresiones faciales

El reconocimiento de expresiones faciales es la capacidad de identificar y analizar las emociones que se manifiestan a través de los cambios faciales y gestuales. Este proceso es fundamental para la comunicación y la interacción social entre humanos, ya que permite transmitir estados emocionales a otros individuos (Saracco, 2012).

Debido a la importancia de las expresiones faciales en la comunicación, se han realizado numerosas investigaciones con el objetivo de mejorar la interacción a través medios digitales. Un objetivo importante de esta investigación es el reconocimiento de expresiones faciales mediante técnicas de deep learning. Para comprender mejor este objetivo, es esencial conocer algunos conceptos básicos, como las expresiones faciales, las emociones universales, y las técnicas de reconocimiento automático de expresiones faciales.

- Expresión facial

  Las expresiones faciales son importantes en la interacción con otras personas, ya que revelan estados mentales complejos y aportan información emocional y de comunicación (Darwin, 1872). El rostro humano es el principal sistema de señales para mostrar emociones y es la parte del cuerpo más observada durante cualquier interacción social, la Figura N° 2.13 muestra ejemplos de expresiones faciales de un conjunto de datos conocido como CK+.

FIGURA N° 2.13: Ejemplo de expresiones faciales en las imágenes de la base de datos CK+.



Nota: Adaptado de (Lucey et al., 2010)

- Emociones básicas

  Una emoción es un cambio temporal e intenso en el estado de ánimo, que

puede ser placentero o doloroso, y que se manifiesta con respuestas físicas en el cuerpo (Real Academia Española, 2020).

Según (Ekman, 1994) existen expresiones faciales universales que distinguen cada una de las principales emociones (Ira, asco, miedo, alegría, tristeza y sorpresa). Numerosas investigaciones se basan en estas principales emociones por su estabilidad a través de culturas, edades y otros factores de identidad.

De acuerdo a la (Real Academia Española, 2020) las principales emociones se definen de la siguiente manera:

Ira	:	Sentimiento de indignación que causa enojo.
Asco	:	Impresión desagradable causada por algo que repugna.
Miedo	:	Perturbación angustiosa del ánimo por un riesgo o daño real o imaginario.
Alegría	:	Sentimiento grato y vivo que suele manifestarse con signos exteriores.
Tristeza	:	Sentimiento de dolor anímico producido por un suceso desfavorable que suele manifestarse con un estado de ánimo pesimista, la insatisfacción y la tendencia al llanto.
Sorpresa	:	Alteración emocional causada por algo imprevisto o inesperado.

- Técnicas de reconocimiento automático de expresiones faciales.

  Estas técnicas no son intrusivas y pueden brindar resultados sumamente precisos sin requerir equipos costosos o especializados. Su implementación es computarizada y se basa en algoritmos que detectan rostros de forma instantánea, codifican sus expresiones y reconocen la emoción facial. Para ello, se utilizan cámaras digitales conectadas a una computadora, teléfono inteligente u otro dispositivo móvil, capturando videos del rostro de los sujetos mientras son expuestos a distintos estímulos que desencadenan sus expresiones.

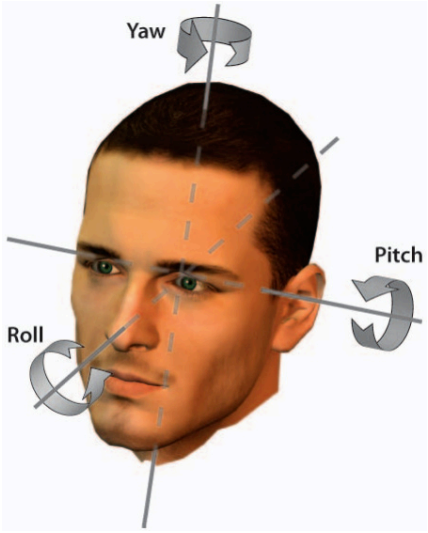
2.2.5.2 Reconocimiento de postura de la cabeza

El reconocimiento de postura de la cabeza ha sido un problema ampliamente estudiado en el área de visión por computadora, debido a que tiene muchas aplicaciones incluyendo la estimación de la atención en personas (Chen et al., 2021).

En visión por computadora, la estimación de la postura de la cabeza se refiere a la capacidad de inferir la orientación de la cabeza de una persona con respecto

a la cámara. La cabeza humana puede modelarse como un objeto rígido con tres grados de libertad, correspondientes a los ángulos Pitch (cabeceo), Roll (balanceo) y Yaw (guiñada), como se ilustra en la Figura N° 2.14 (Murphy-Chutorian y Trivedi, 2008).

FIGURA N° 2.14: Grados de libertad de la cabeza humana.



Nota: Adaptado de (Murphy-Chutorian y Trivedi, 2008)

2.3 DEFINICIÓN DE TÉRMINOS

a) El filtro (kernel)

Es un conjunto de pesos que se aplica a toda la capa convolucional para aprender patrones. Se desliza sobre los datos de entrada, produciendo un mapa de características bidimensional. Los filtros tienen cuatro hiperparámetros: tamaño (área del filtro: altura por ancho), profundidad, zancada y relleno con ceros.

b) El tamaño (size)

Es el área del filtro, que puede variar, por ejemplo, 3×3, 5×5 y 7×7, y no necesariamente es cuadrado. Los filtros más grandes tienden a superponerse más, lo que puede mejorar la precisión de la clasificación, pero también produce salidas más grandes. Por lo tanto, administrar el tamaño de los resultados de las capas convolucionales es crucial para mantener la eficiencia de la red.

c) La profundidad (depth)

La profundidad del filtro se relaciona con la complejidad de una imagen, es decir, si una imagen de entrada tiene 3 canales, entonces un filtro aplicado a

esa imagen debe tener una profundidad de 3.

d) La zancada (stride)

Se refiere a la cantidad de movimiento del filtro a lo largo de la imagen de entrada. Específicamente, la zancada define el cambio en el desplazamiento del filtro tanto en las direcciones horizontal como vertical. Por lo general, la zancada es simétrica en dimensiones de altura y ancho, y se mide en píxeles.

e) El relleno con ceros (zero-padding)

Es el proceso de añadir ceros alrededor de los bordes de una imagen de entrada para ajustar el tamaño de salida de una capa convolucional. Esto permite que el tamaño de la salida sea controlado y a menudo igual al tamaño de la entrada, facilitando la gestión de los valores de zancada y profundidad.

f) Convolución

Es una operación matemática que involucra dos componentes: la imagen de entrada y el filtro o kernel de convolución. La convolución se realiza deslizando el kernel sobre la imagen de entrada, calculando el producto escalar en cada posición, lo que produce un mapa de características.

g) Tasa de aprendizaje (Learning rate)

Es un parámetro importante para los algoritmos de optimización utilizados en el entrenamiento de modelos de aprendizaje automático. Controla el tamaño del ajuste aplicado a los pesos de la red neuronal en cada iteración durante el proceso de entrenamiento. Elegir un learning rate adecuado influye directamente en la velocidad y estabilidad de la convergencia del modelo.

h) Función de activación

Permiten a las neuronas activarse de manera no lineal en respuesta a la entrada recibida, facilitando la captura de patrones y características no lineales en los datos. Así, las funciones de activación permiten obtener representaciones profundas y mejorar el rendimiento de las redes neuronales en tareas de aprendizaje automático y reconocimiento de patrones.

i) Transfer Learning

Es una técnica que permite tomar un modelo preentrenado en grandes conjuntos de datos, tomándolo como punto de partida, y adaptarlo para realizar una clasificación diferente en menor entrenamiento. Esto se logra gracias a que el modelo preentrenado ya sabe reconocer características generales, como bordes, texturas y formas básicas, que son necesarias para cualquier tipo de clasificación de imágenes.

j) Haar Cascade

Es un tipo de clasificador de características utilizado en el método de detección de objetos de Viola-Jones. Propuesto por Paul Viola y Michael Jones en 2001, este método se destacó por su capacidad para detectar objetos con tasas de detección relativamente altas, especialmente efectivo en la detección de rostros en tiempo real.

k) GPU - Unidad de procesamiento gráfico (Graphics Processing Unit)

Es un circuito electrónico capaz de realizar cálculos matemáticos a alta velocidad. Está diseñado para ejecutar operaciones similares en paralelo sobre grandes conjuntos de datos, lo que lo hace ideal para tareas como la renderización de gráficos, el machine learning y la edición de videos. Esta capacidad de procesamiento paralelo aumenta la eficiencia en tareas que requieren gran capacidad de cómputo.

l) Retropropagación (Backpropagation)

Es un método de cálculo utilizado en algoritmos para entrenar redes neuronales artificiales, propagando hacia atrás el error obtenido en la capa de salida hacia la capa de entrada, a través de la gradiente descendente, que permite ajustar los pesos y sesgos de la red.

CAPÍTULO III: METODOLOGÍA

3.1 TIPO Y NIVEL DE INVESTIGACIÓN

La presente investigación se clasifica como aplicada o tecnológica, dado que este tipo de estudios plantea problemas e hipótesis con el objetivo de solucionar cuestiones prácticas de la vida productiva de la sociedad. Se basa en conocimientos teóricos obtenidos de investigaciones básicas o fundamentales, y su resultado es la creación de un producto tecnológico (Paitán et al., 2014). Esto se refleja en el desarrollo de este estudio, que utiliza el conocimiento en Deep Learning para diseñar modelos capaces de reconocer expresiones faciales y la postura de la cabeza.

3.2 POBLACIÓN Y MUESTRA

3.2.1 Población

La población utilizada para el reconocimiento de expresiones faciales está conformada por $35\,887$ imágenes de expresiones faciales de rostros del dataset FER-2013 (Goodfellow et al., 2015).

La población utilizada para el reconocimiento de posición de cabeza está conformada por $15\,678$ imágenes de personas fotografiadas girando la cabeza, pertenecientes al dataset *Biwi Kinect Head Pose* (Fanelli, Dantone, Gall, Fossati, y Van Gool, 2013).

3.2.2 Muestra

Para la muestra se utilizaron $35\,795$ imágenes del dataset FER-2013 y 4000 del dataset Biwi, ya que estas fueron las imágenes finales que se usaron para el entrenamiento y evaluación de los modelos, luego del preprocesamiento de los datos.

3.3 INSTRUMENTOS

a) Google Colaboratory

   Google Colaboratory, o Colab, permite programar y ejecutar Python en el navegador sin necesidad de una configuración previa, con acceso gratuito a recursos informáticos (GPU y TPU limitados) y facilidad para el trabajo colaborativo. Es útil para estudiantes, científicos de datos e investigadores de

IA, ya que combina código ejecutable y cuadernos interactivos almacenados en Google Drive. Colab soporta múltiples bibliotecas de Python y permite importar datos desde diversas fuentes. Se puede integrar con otros frameworks como Pytorch y Tensorflow, siendo ideal para desarrollar redes neuronales. Se debe tener en cuenta que el plan gratuito tiene 15GB de VRAM, por lo que, si se supera este límite, se tendrá que acceder a otros planes de pago, los cuales tienen mayores recursos en cuanto a unidades informáticas, VRAM, etc.

b) Pytorch

PyTorch es un framework de deep learning de código abierto, ampliamente utilizado por científicos de datos e investigadores de deep learning debido a su rendimiento, facilidad de uso y flexibilidad. PyTorch proporciona una amplia gama de operaciones tensoriales, que están altamente optimizadas, permitiendo su fácil uso para ejecutarse en CPUs, GPUs y TPUs. También aprovecha conceptos de Python como clases, estructuras y bucles condicionales, permitiendo un enfoque orientado a objetos para el desarrollo de algoritmos de deep learning. (Subramanian, 2018).

c) ONNX

Pytorch permite exportar los modelos de Deep Learning al formato ONNX (Open Neural Network Exchange), el cual es una representación intermedia ampliamente soportada por diversas librerías y frameworks de deep learning, permitiendo entrenar un modelo en PyTorch y luego exportarlo en el formato onnx para luego importarlo en otros entornos, como TensorFlow. De esta manera ONNX maximiza la flexibilidad, ya que no se necesitaría Pytorch para poner en producción en otros entornos, incluso en navegadores web y IoT. Sin embargo, la cantidad y tipo de operaciones soportadas por ONNX puede ser limitada, aunque esto está en constante mejora.

d) Django

Es un framework web de código abierto y gratuito con una comunidad activa y gran documentación, escrito en Python. Permite crear aplicaciones web de manera fácil, rápida y con menos código, ya que cuenta con un gran ecosistema de componentes listos para ser incorporados en cualquier proyecto.

e) Django Channels

Django Channels o simplemente canales es una extensión de Django que amplía sus capacidades, permitiendo a los proyectos gestionar no solo HTTP, sino también la creación de aplicaciones en tiempo real utilizando WebSockets, los cuales permiten una comunicación bidireccional entre el cliente y el

servidor, lo que posibilita el envío de actualizaciones al cliente sin necesidad de solicitudes continuas.

f) WebRTC

WebRTC permite que las aplicaciones y sitios web intercambien datos entre navegadores. Para ello, establece conexiones entre dos pares mediante la interfaz RTCPeerConnection, los cuales no requieren controladores ni complementos especiales y no necesitan ningún servidor intermediario. Una vez establecida la conexión, se pueden agregar transmisiones multimedia y/o canales de datos a la conexión.

g) Arquitecturas CNN

- Resnet18
- Densenet
- Alexnet

h) Métricas

- Accuracy
- Precision
- Recall
- F1-score

3.4 PROCEDIMIENTOS

3.4.1 Preparación de los Datos

3.4.1.1 Obtención de Datos

- Expresiones Faciales: Se utilizó el dataset FER-2013, que fue creado por Pierre Luc Carrier y Aaron Courville, fue desarrollado utilizando la API de búsqueda de imágenes de Google para recopilar imágenes de rostros utilizando palabras clave referentes a emociones. Se utilizó OpenCV para detectar rostros, seguido de una verificación y corrección manual. Las imágenes resultantes fueron redimensionadas a 48x48 píxeles y convertidas a escala de grises. Mehdi Mirza e Ian Goodfellow ajustaron el conjunto de datos para el concurso de la *International Conference on Machine Learning (ICML 2013)*, clasificando las emociones en siete categorías. El conjunto de datos final contiene $35\,887$ imágenes distribuidas entre las emociones de ira (angry), disgusto

(disgust), miedo (fear), felicidad (happy), tristeza (sad), sorpresa (surprise) y neutral (neutral) (Goodfellow et al., 2015).

- Postura de la Cabeza: Se empleó el dataset *Biwi Kinect Head Pose*, creado por Gabriele Fanelli y su equipo de investigación, diseñado para estimar la postura de la cabeza en tiempo real utilizando el dispositivo Kinect de Microsoft. El conjunto de datos incluye grabaciones de 20 personas, 4 de las cuales fueron filmadas en dos ocasiones, mientras giraban la cabeza para cubrir un amplio rango de ángulos. En total, el dataset contiene 15678 imágenes RGB de 640x480 píxeles, acompañadas de anotaciones sobre la posición y rotación de la cabeza en 3D (Fanelli et al., 2013).

  Las imágenes para este dataset fueron etiquetadas manualmente en dos clases: front y no-front, siguiendo los criterios que se describen a continuación:

  - front: Imágenes donde la cabeza está orientada frontalmente hacia la cámara. Considerando un rango de -20º a 20º en el eje Pitch, de -45º a 45º en los ejes Yaw, y de -50º a 50º en los ejes Roll. Este rango solo se clasifica como frontal si la mirada está orientada hacia la cámara.
  - no_front: Imágenes en las que la cabeza no está orientada frontalmente hacia la cámara, es decir, aquellas que no cumplen los criterios de la clase front.

3.4.1.2 Preprocesamiento de Datos

- Limpieza de datos: En el caso del conjunto de datos FER-2013, se eliminaron 92 imágenes que no correspondían a sus categorías correctas o que no representaban una expresión facial.
- Normalización: Todas las imágenes fueron normalizadas a valores entre 0 y 1 dividiendo los valores de los píxeles por 255. Esto contribuye a un entrenamiento más rápido, y un mejor rendimiento y generalización del modelo.
- Redimensionamiento: En el conjunto de datos Biwi, no fue necesario redimensionar las imágenes, ya que su tamaño era adecuado. No obstante, en el caso del conjunto de datos FER-2013, fueron redimensionadas a 224x224 píxeles para ser compatibles con las arquitecturas de Deep Learning utilizadas.
- Balanceo de clases: En el caso del dataset FER-2013, se calcularon los class weights asignando un mayor peso a las clases minoritarias, lo que aumenta su relevancia en la función de pérdida y reduce el riesgo de que el modelo pase por alto estas clases. Esta estrategia impulsa al modelo a aprender mejor sobre las clases minoritarias, mejorando su sensibilidad para detectarlas.

Para el dataset Biwi, se implementó la técnica de random undersampling, seleccionando aleatoriamente 2000 imágenes por clase con el fin de equilibrar la cantidad de ejemplos entre las clases front y no_front.

3.4.2 División del Conjunto de Datos

Para llevar a cabo el entrenamiento y la evaluación de los modelos, se dividieron los datasets FER-2013 y Biwi en proporciones de 80 % para train, 10 % para validation y 10 % para test.

3.4.3 Implementación de arquitecturas

Para el dataset FER-2013 y Biwi se implementaron 3 arquitecturas de redes neuronales convolucionales: Resnet18, Densenet y Alexnet, las cuales se muestran en la Tabla N° 3.1, la Tabla N° 3.2 y la Tabla N° 3.3, respectivamente.

3.4.4 Transfer Learning

Se descargaron los modelos preentrenados de las arquitecturas ResNet, DenseNet y AlexNet, los cuales habían sido entrenados previamente con el dataset ImageNet. Posteriormente, se utilizó la técnica de Transfer Learning, aprovechando los pesos aprendidos en ImageNet como punto de partida para nuestros modelos. Esto permitió que los modelos mantuvieran conocimientos previos en la detección de características generales de imágenes, facilitando el proceso de adaptación a nuestros datasets específicos, FER-2013 y Biwi.

Además de la opción de congelar algunas capas y actualizar solo las finales, se optó también por actualizar los pesos en todas las capas de cada arquitectura. Esto permitió que el modelo ajustara sus parámetros completamente a las características particulares de los nuevos datasets. De esta manera, el modelo no solo retuvo los patrones útiles aprendidos en ImageNet, sino que también los adaptó progresivamente, optimizando su precisión y sensibilidad en el nuevo contexto de los datasets FER-2013 y Biwi.

3.4.5 Entrenamiento de los modelos

Los modelos se entrenaron utilizando el optimizador Adam y para el caso del dataset FER-2013, se usó class weights en la función de pérdida, y así compensar el desbalance de clases, asignando más peso a las clases minoritarias.

TABLA N° 3.1: Arquitectura del modelo Resnet18

Resnet18	
Capa	**Kernel size / Stride / Padding**
Conv	7x7 / 2x2 / 3x3
Maxpool	3x3 / 2x2 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 2x2 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	1x1 / 2x2
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 2x2 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	1x1 / 2x2
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 2x2 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	1x1 / 2x2
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Fc linear	Biwi: **2** FER-2013: **7**

TABLA N° 3.2: Arquitectura del modelo DenseNet

DenseNet	
Capa	**Kernel size / Stride / Padding**
Conv	7x7 / 2x2 / 3x3
Maxpool	3x3 / 2x2 / 1x1
Dense Block 1	
Conv (1x1)	1x1 / 1x1 / 0x0
Conv (3x3)	3x3 / 1x1 / 1x1
Transition Layer 1	
Conv	1x1 / 1x1 / 0x0
Maxpool	2x2 / 2x2 / 0x0
Dense Block 2	
Conv (1x1)	1x1 / 1x1 / 0x0
Conv (3x3)	3x3 / 1x1 / 1x1
Transition Layer 2	
Conv	1x1 / 1x1 / 0x0
Maxpool	2x2 / 2x2 / 0x0
Dense Block 3	
Conv (1x1)	1x1 / 1x1 / 0x0
Conv (3x3)	3x3 / 1x1 / 1x1
Transition Layer 3	
Conv	1x1 / 1x1 / 0x0
Maxpool	2x2 / 2x2 / 0x0
Dense Block 4	
Conv (1x1)	1x1 / 1x1 / 0x0
Conv (3x3)	3x3 / 1x1 / 1x1
Fc linear	Biwi: **2** FER-2013: **7**

Durante el proceso de entrenamiento, se utilizaron las particiones de entrenamiento (train) y validación del conjunto de datos (val), ajustando hiperparámetros clave como la tasa de aprendizaje (learning rate), el tamaño de los lotes (batch size) y el número de épocas, con el objetivo de obtener el modelo con el mejor rendimiento posible.

Además, se implementaron técnicas de transfer learning para evaluar posibles me-

TABLA N° 3.3: Arquitectura del modelo Alexnet

AlexNet	
Capa	**Kernel size / Stride / Padding**
Conv	11x11 / 4x4 / 2x2
Maxpool	3x3 / 2x2 / 0x0
Conv	5x5 / 1x1 / 2x2
Maxpool	3x3 / 2x2 / 0x0
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Maxpool	3x3 / 2x2 / 0x0
Fc linear	(4096)
Fc linear	(4096)
Fc linear	Biwi: **2** / FER-2013: **7**

joras en los modelos, aprovechando el conocimiento adquirido de modelos preentrenados en grandes conjuntos de datos y adaptándolo a los conjuntos de datos específicos utilizados en esta investigación.

3.4.6 Evaluación de los modelos

Después de completar el entrenamiento de cada modelo, se procedió a su evaluación utilizando la partición de prueba (test), que no fue utilizada en el entrenamiento. Las métricas para evaluar los modelos fueron F1-score macro para el reconocimiento de expresiones faciales y accuracy para la clasificación de la postura de la cabeza.

3.4.7 Evaluación del funcionamiento de los modelos en tiempo real

Los modelos se exportaron al formato ONNX y luego se importaron utilizando la librería ONNX Runtime para Python en la aplicación web.

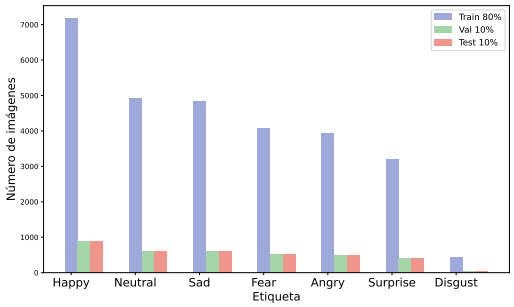
La aplicación web fue desarrollada en Django, empleando Django Channels y WebRTC para manejar la comunicación de audio y video en tiempo real. Se establecieron dos tipos de usuarios: uno común, que puede transmitir y recibir video, y un administrador con capacidades adicionales. Este administrador, además de transmitir y recibir video, puede ver los resultados obtenidos de los modelos importados, correspon-

dientes al reconocimiento de expresiones faciales y la postura de la cabeza a partir de los frames de la cámara web de los usuarios (ver Anexo2).

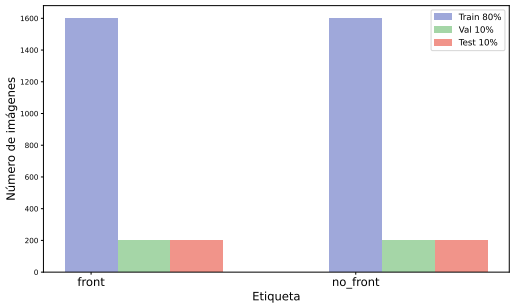
Una vez seleccionado el mejor modelo, se llevó a cabo una prueba en tiempo real con dos usuarios y un administrador (ver Anexo1). Durante esta fase, se evaluó el funcionamiento del modelo en el reconocimiento de expresiones faciales y la postura de la cabeza.

3.5 ANÁLISIS DE DATOS

FIGURA N° 3.1: Distribución de los dataset



(a) Distribución del dataset FER-2013. Adaptado de Goodfellow et al. (2015)



(b) Distribución del dataset Biwi. Elaboración propia

3.5.1 Reconocimiento de la postura de la cabeza

Para el reconocimiento de la postura de la cabeza, se utilizó el dataset Biwi, compuesto por 15678 imágenes etiquetadas en dos clases: front y no_front. Debido a que la mayoría de las imágenes pertenecían a la clase no_front, y para evitar sesgo hacia esta clase mayoritaria, se aplicó la técnica de random undersampling, seleccionando 2000 imágenes por clase. Este balance no solo evitó el sesgo, sino que también redujo el tiempo y los recursos computacionales necesarios para el entrenamiento, al limitar el número total de imágenes procesadas. En la Figura N° 3.1b y la Tabla N° 3.4 se muestran los detalles de esta reducción, así como la división del conjunto de datos en tres particiones: 80 % para entrenamiento, 10 % para validación y 10 % para prueba.

Para el entrenamiento, se utilizó el optimizador Adam, ya que ofrece una convergencia más rápida y estable en comparación con el Stochastic Gradient Descent (SGD), el cual fue evaluado en pruebas preliminares, pero arrojó resultados menos consistentes.

El desempeño de los modelos se evaluó utilizando accuracy, dado que las clases

TABLA N° 3.4: Distribución de las imágenes del dataset Biwi

Etiqueta	Total	Train 80 %	Val 10 %	Test 10 %
frontal	2000	1600	200	200
no_frontal	2000	1600	200	200
Total	**4000**	**3200**	**400**	**400**

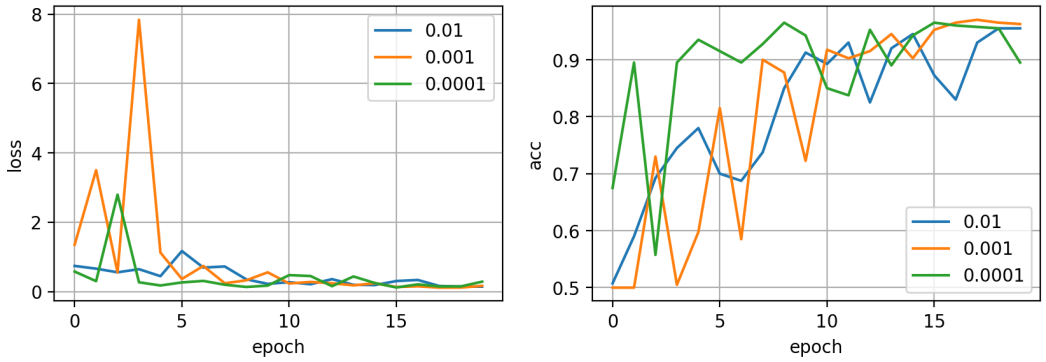
estaban balanceadas, proporcionando así una medida general de las predicciones correctas. Además, se generaron gráficos durante el entrenamiento para visualizar el comportamiento de los modelos a lo largo de las épocas.

Por ejemplo, para la arquitectura Resnet18, la Figura N° 3.2 muestra la evolución de los indicadores de rendimiento loss y accuracy por cada época con diferentes hiperparámetros, como el learning rate y el batch size, sobre el conjunto de validación. En la Figura N° 3.2a, se observa que los valores de learning rate 0.01, 0.001 y 0.0001 mostraron un buen desempeño, lo que justificó su uso en pruebas posteriores. En la Figura N° 3.2b, se evalúan diferentes tamaños de batch size, donde los valores de 16 y 64 ofrecieron un mejor rendimiento que el de 32, por lo que se optó por utilizar un batch size de 16 para optimizar los recursos computacionales.

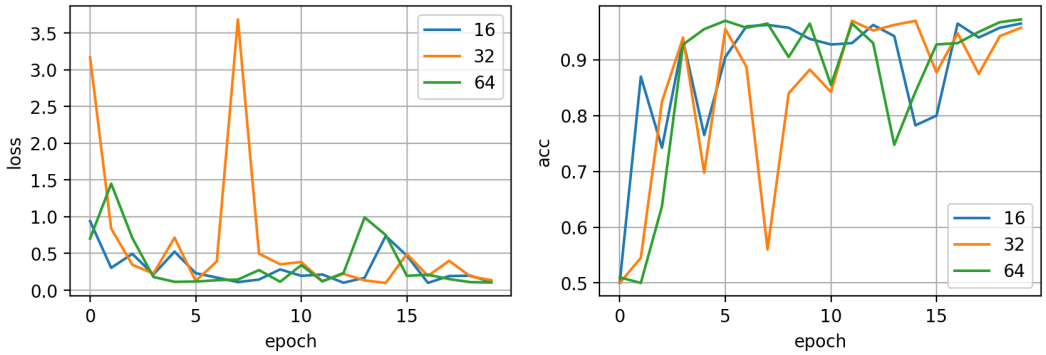
En la Tabla N° 3.5, se resumen los hiperparámetros utilizados para la comparativa entre las arquitecturas CNN y sus indicadores en los datos de prueba (test). Los resultados obtenidos, revelan que el mejor modelo en cada arquitectura se logró mediante el uso de transfer learning con todas las capas descongeladas. Esto se debe a que los modelos preentrenados ya reconocen características básicas como bordes, texturas y formas, lo que facilita su adaptación con menos datos. Este enfoque permitió que los modelos alcancen buenas métricas desde la primera época, optimizando el rendimiento en menos tiempo y con menos recursos computacionales.

Finalmente, en la Figura N° 3.3 se observa la evolución de los indicadores loss y accuracy por cada época para el mejor modelo de cada arquitectura. Todas las arquitecturas mostraron una evolución positiva desde las primeras épocas, sin indicios de sobreajuste en las 20 épocas de entrenamiento. Las tres arquitecturas lograron un accuracy superior al 96 % con el uso de transfer learning. Sin embargo, Resnet18 demostró ser más eficiente en términos de consumo de recursos y tiempo de entrenamiento, lo que lo convierte en la mejor opción para aplicaciones en tiempo real, mientras que Densenet y Alexnet requirieron más recursos y tiempo de procesamiento, aspectos que deben considerarse al seleccionar un modelo adecuado.

FIGURA N° 3.2: Prueba de hiperparámetros para Resnet18 - Biwi



(a) Prueba de learning rates



(b) Prueba de batch size

3.5.2 Reconocimiento de expresiones faciales

Para el reconocimiento de expresiones faciales, se utilizó el conjunto de datos FER-2013, que originalmente contenía $35\,887$ imágenes etiquetadas en 7 clases: 8989 imágenes para happy, 6198 imágenes para neutral, 6077 imágenes para sad, 5121 imágenes para fear, 4953 imágenes para angry, 4002 imágenes para surprise y 547 imágenes para disgust. Durante la limpieza de datos, se descartaron 92 imágenes que no correspondían a sus categorías correctas o que no representaban una expresión facial. Como se muestra en la Tabla N° 3.6, los datos se particionaron en proporciones del 80 % para train, 10 % para validation y 10 % para test.

En la Figura N° 3.1a, se observa un desbalance en la cantidad de imágenes por clase. Este desequilibrio, que podría afectar el rendimiento del clasificador, se gestionó mediante la técnica de class weights, asignando mayores pesos a las clases minoritarias en la función de pérdida, logrando que estas clases tomen mayor relevancia. Los pesos asignados a cada clase fueron: 1.0360 para Angry, 9.3616 para

TABLA N° 3.5: Indicadores de modelos aplicadas a Biwi

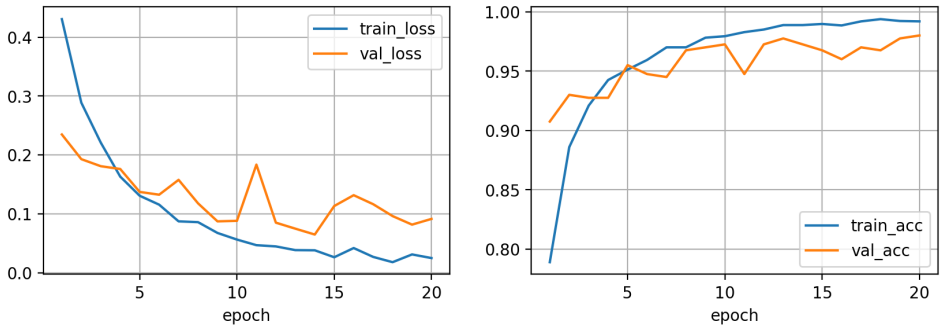
Arquit.	Hiperparámetros						Indicadores	
	Ep.	Lr	Bch.	Optim.	Pesos	Frz.	Loss	Acc.
Alexnet	20	0.1	16	Adam	Aleatorio	No	0.91126	0.5
Alexnet	20	0.01	16	Adam	Aleatorio	No	0.70106	0.5
Alexnet	20	0.001	16	Adam	Aleatorio	No	0.0.6932	0.5
Alexnet	20	0.0001	16	Adam	Aleatorio	No	0.21205	09475
Alexnet	20	0.001	16	Adam	Imagenet	No	0.69337	0.5
Alexnet	20	0.0001	16	Adam	Imagenet	No	**0.10558**	**0.9625**
Alexnet	20	0.00001	16	Adam	Imagenet	No	0.13933	0.96
Densenet	20	0.1	16	Adam	Aleatorio	No	0.222	0.905
Densenet	20	0.01	16	Adam	Aleatorio	No	0.40704	0.84
Densenet	20	0.001	16	Adam	Aleatorio	No	0.21579	0.91
Densenet	20	0.001	16	Adam	Imagenet	No	**0.08385**	**0.98**
Densenet	20	0.0001	16	Adam	Imagenet	No	0.0853	0.975
Densenet	20	0.00001	16	Adam	Imagenet	No	0.08984	0.9775
Resnet	20	0.01	16	Adam	Aleatorio	No	0.25197	0.915
Resnet	20	0.001	16	Adam	Aleatorio	No	0.1744	0.9475
Resnet	20	0.001	16	Adam	Imagenet	Si	0.4686	0.7875
Resnet	20	0.001	16	Adam	Imagenet	No	0.14264	0.9625
Resnet	20	0.0001	16	Adam	Imagenet	No	**0.09009**	**0.985**
Resnet	20	0.00001	16	Adam	Imagenet	No	0.1209	0.9625

TABLA N° 3.6: Distribución de las imágenes del dataset FER-2013

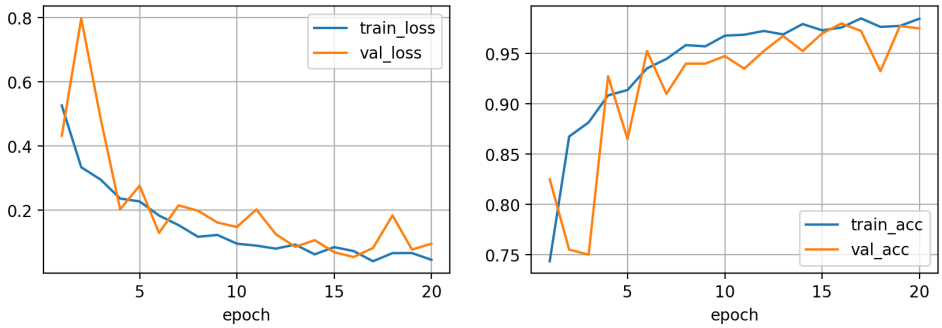
Etiqueta	Total	Train 80%	Val 10%	Test 10%
Happy	8971	7177	897	897
Neutral	6172	4938	617	617
Sad	6067	4853	607	607
Fear	5110	4088	511	511
Angry	4937	3950	493	494
Surprise	3992	3194	399	399
Disgust	546	437	55	54
Total	**35795**	**28637**	**3579**	**3579**

Disgust, 1.0007 para Fear, 0.5700 para Happy, 0.8285 para Neutral, 0.8428 para Sad, y 1.2808 para Surprise. Estos pesos se calcularon de forma inversamente

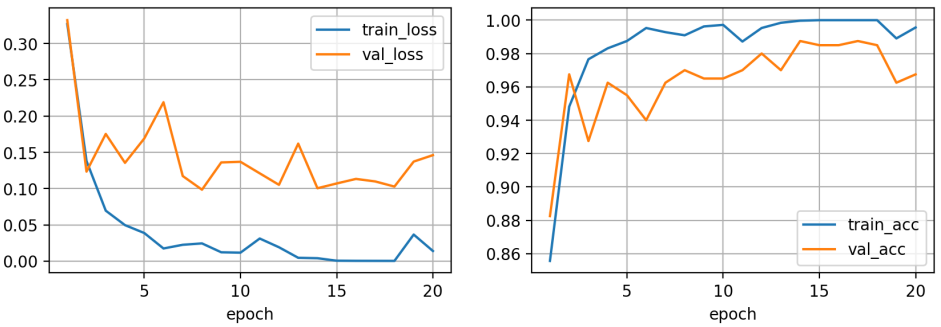
FIGURA N° 3.3: Métricas para las arquitecturas propuestas - Biwi



(a) Modelo Alexnet



(b) Modelo Densenet



(c) Modelo Resnet18

proporcional al número de muestras en cada clase, de manera que las clases con menos muestras recibieran un mayor peso, y así compensar el desbalance en el conjunto de datos durante el entrenamiento.

Fue necesario ajustar las dimensiones de las imágenes del conjunto de datos FER-2013 de 48x48 a 224x224 píxeles, ya que el tamaño original era insuficiente para arquitecturas como ResNet18, DenseNet y AlexNet, ya que estas redes cuentan con capas convolucionales profundas, en las que las dimensiones de las imágenes

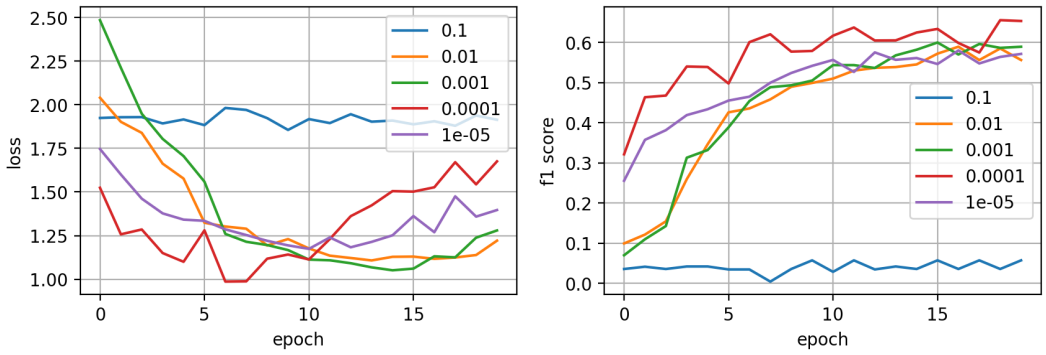
se reducen progresivamente durante los procesos de convolución y pooling.

Al igual que en el conjunto de datos Biwi, en FER-2013 también se utilizó el optimizador Adam, debido a las ventajas mencionadas en la Subsección 3.5.1 en comparación con otros optimizadores como SGD.

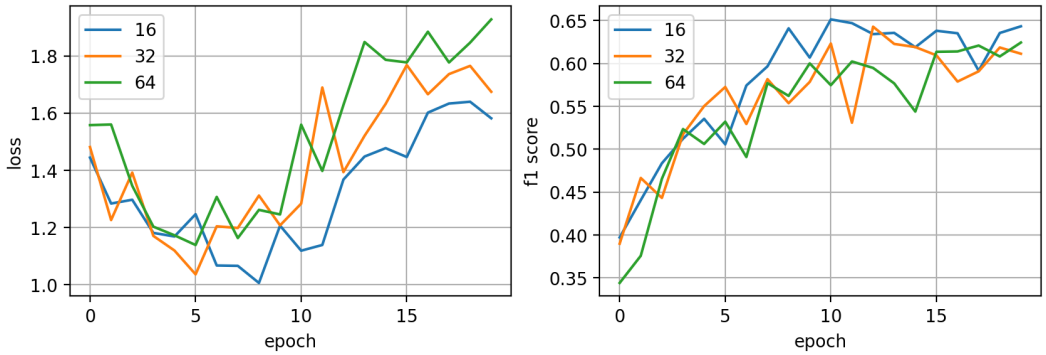
Para el entrenamiento y la evaluación de los modelos con el conjunto de datos FER-2013, se utilizaron los indicadores loss y F1-score debido al desbalance de clases. El F1-score es especialmente útil en escenarios desbalanceados, ya que ofrece una medida equilibrada entre precisión (precision) y exhaustividad (recall), lo que permite una evaluación más justa del desempeño en las clases minoritarias.

Para determinar los mejores hiperparámetros de learning rate y batch size durante el entrenamiento del modelo de reconocimiento de expresiones faciales, se realizaron múltiples experimentos para las arquitecturas: AlexNet, DenseNet y ResNet. A continuación en la Figura N° 3.4 se muestran los resultados obtenidos con DenseNet.

FIGURA N° 3.4: Prueba de Hiperparámetros - DenseNet



(a) Learning rates



(b) Batch sizes

En la Figura N° 3.4a, se observa el impacto de diferentes learning rates (0.1, 0.01, 0.001, 0.0001 y 1e-05) para loss y F1-score, donde se puede notar que con 0.1 no hay mejora para loss ni para F1 Score. Learning rates menores a 0.1 logran resultados muy parecidos, por lo que conviene probarlos en diferentes modelos.

En la Figura N° 3.4b, se observan diferentes batch size: 16, 32 y 64, donde se evidencia que los más pequeños, como 16, logran un menor loss y un mayor F1-score, por lo que se eligió trabajar solo con un batch size de 16.

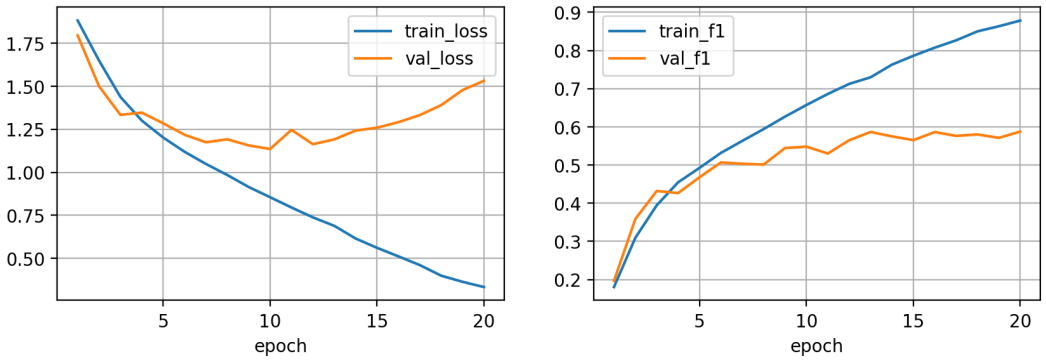
La Tabla N° 3.7 presenta los hiperparámetros y métricas obtenidas de los modelos basados en las arquitecturas CNN (AlexNet, DenseNet y ResNet) utilizando los datos de prueba de FER-2013. Para cada arquitectura, se evalúan diversas configuraciones de hiperparámetros, incluyendo el número de épocas (Ep), la tasa de aprendizaje (Lr), el tamaño del batch (Bch), el optimizador utilizado (Adam), el origen de los pesos (aleatorios o preentrenados con ImageNet), y si las capas fueron congeladas o no (Frz). Además, se presentan los indicadores de rendimiento de cada modelo. Los mejores modelos para cada arquitectura, indicados en negrita, muestran que el uso de un batch size y un learning rate adecuado alcanzaron mejores resultados.

Después de encontrar los hiperparámetros más adecuados para las tres arquitecturas (AlexNet, DenseNet y ResNet), se procedió a aplicar Transfer Learning utilizando pesos preentrenados en ImageNet. El Transfer Learning tuvo un impacto positivo en el rendimiento de los modelos DenseNet y ResNet. Sin embargo, en el caso de AlexNet, los resultados no mostraron ninguna mejora significativa. La Figura N° 3.5 muestra el progreso de las métricas de loss y F1 Score a lo largo de las épocas para los mejores modelos de cada arquitectura.

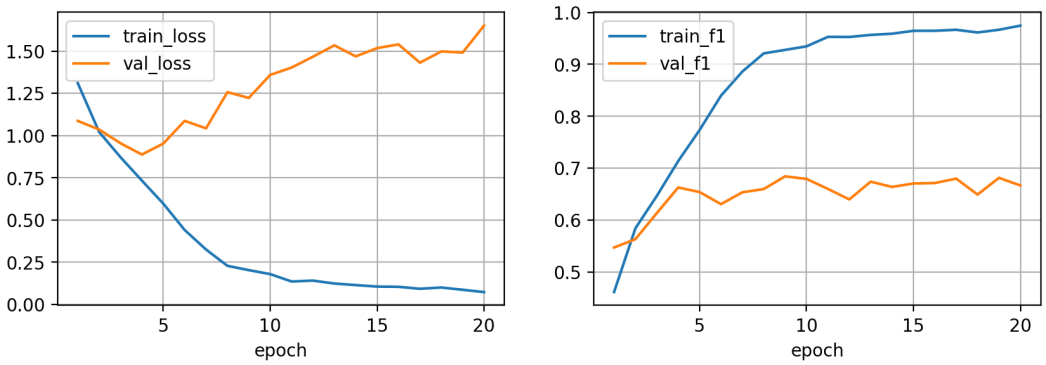
TABLA N° 3.7: Indicadores de modelos aplicadas a FER-2013

Arquit.	Hiperparámetros						Indicadores	
	Ep.	Lr	Bch.	Optim.	Pesos	Frz.	Loss	F1 S.
AlexNet	20	0.1	16	Adam	Aleatorio	No	1.94184	0.05726
AlexNet	20	0.01	16	Adam	Aleatorio	No	1.87909	0.05726
AlexNet	20	0.001	16	Adam	Aleatorio	No	1.89961	0.05726
AlexNet	20	0.0001	16	Adam	Aleatorio	No	**1.41357**	**0.62660**
AlexNet	20	0.00001	16	Adam	Aleatorio	No	1.16368	0.58304
AlexNet	30	0.000001	16	Adam	Aleatorio	No	1.67921	0.29762
AlexNet	20	0.001	16	Adam	Imagenet	No	1.91254	0.05726
AlexNet	20	0.0001	16	Adam	Imagenet	No	1.24802	0.60447
AlexNet	20	0.00001	16	Adam	Imagenet	No	1.23051	0.52739
DenseNet	20	0.1	16	Adam	Aleatorio	No	1.85545	0.05726
DenseNet	20	0.01	16	Adam	Aleatorio	No	1.09759	0.59556
DenseNet	20	0.001	16	Adam	Aleatorio	No	1.10036	0.59825
DenseNet	20	0.0001	16	Adam	Aleatorio	No	1.70392	0.62693
DenseNet	20	0.00001	16	Adam	Aleatorio	No	1.31060	0.57620
DenseNet	20	0.001	16	Adam	Imagenet	No	1.11745	0.60236
DenseNet	20	0.0001	16	Adam	Imagenet	No	**1.30870**	**0.67177**
DenseNet	20	0.00001	16	Adam	Imagenet	No	1.67867	0.65054
ResNet	20	0.1	16	Adam	Aleatorio	No	1.85399	0.05726
ResNet	20	0.01	16	Adam	Aleatorio	No	1.70253	0.57893
ResNet	20	0.001	16	Adam	Aleatorio	No	1.93880	0.58425
ResNet	20	0.0001	16	Adam	Aleatorio	No	1.20364	0.56769
ResNet	20	0.001	16	Adam	Imagenet	No	2.09195	0.61305
ResNet	20	0.0001	16	Adam	Imagenet	No	**1.06724**	**0.64560**
ResNet	20	0.00001	16	Adam	Imagenet	No	1.34648	0.60950

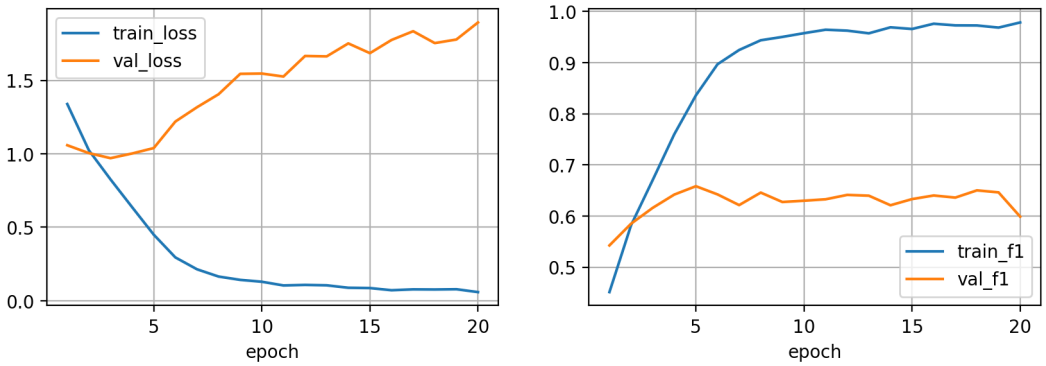
FIGURA N° 3.5: Mejores modelos obtenidos - FER-2013



(a) AlexNet



(b) DenseNet



(c) ResNet

CAPÍTULO IV: RESULTADOS Y DISCUSIÓN

4.1 Reconocimiento de la postura de la cabeza

TABLA N° 4.1: Indicadores obtenidos por arquitectura, para dataset Biwi

Modelo	Loss	Accuracy
Resnet18	0.09009	**0.985**
Densenet	**0.08385**	0.98
Alexnet	0.10558	0.9625

En la Tabla N° 4.1, se presentan los indicadores evaluados en el conjunto de datos de prueba para el mejor modelo seleccionado de cada arquitectura, tras haber ajustado diversos hiperparámetros, como se detalla en la Tabla N° 3.5. Los resultados muestran que ResNet18 alcanzó un accuracy del 98.5 %, seguido de Densenet con un 98 %, y finalmente AlexNet con un 96.25 %. En cuanto al loss, Densenet obtuvo el valor más bajo con 0.08385, seguido de ResNet18 con 0.09009, y AlexNet con 0.10558.

Estos resultados sugieren que, aunque tanto ResNet18 como Densenet lograron un alto rendimiento, la diferencia entre ellos es mínima. Sin embargo, se eligió ResNet18 como el mejor modelo debido a su mayor accuracy y al menor consumo de memoria VRAM en comparación con Densenet, además de que al ser una arquitectura más ligera, resulta ser un modelo de menor tamaño, tal como se evidencia en la Tabla N° 4.2. Esto lo convierte en una opción más eficiente para aplicaciones en tiempo real, donde la rapidez es esencial. Cabe destacar que, para el mejor modelo por cada arquitectura, se utilizaron pesos preentrenados, los cuales fueron ajustados en todas las capas del modelo. Este enfoque demostró ser beneficioso desde las primeras épocas de entrenamiento, reduciendo la cantidad de épocas necesarias y, por ende, el tiempo de entrenamiento.

TABLA N° 4.2: Tamaño de los mejores modelos en formato ONNX, para el reconocimiento de la postura de la cabeza

Modelo	Tamaño
Resnet18	**42.6 MB**
Densenet	101.9 MB

En la Figura N° 4.1, se muestra la matriz de confusión para el mejor modelo, ResNet18, sobre un total de 400 predicciones realizadas con el conjunto de datos de

prueba. El modelo presentó 8 predicciones incorrectas, de las cuales 2 imágenes frontales fueron clasificadas como no frontales y 6 imágenes no frontales fueron clasificadas como frontales.

La matriz de confusión indica que el modelo ofrece un alto rendimiento en la clasificación de la postura de la cabeza. No obstante, se identificaron 6 predicciones incorrectas en la clase no frontal, lo que podría ser problemático en aplicaciones donde la detección precisa de posturas no frontales es crítica. Por otro lado, solo 2 imágenes frontales fueron mal clasificadas como no frontales, lo que destaca la alta capacidad del modelo para capturar la postura frontal con precisión.

En conjunto, estos resultados sugieren que el modelo es confiable para el reconocimiento de la postura de la cabeza, como se ilustra en la Figura N° 4.2, donde se muestran 15 imágenes aleatorias del conjunto de prueba con sus etiquetas y predicciones correspondientes. En este caso, todas las imágenes fueron clasificadas correctamente. No obstante, sería beneficioso realizar ajustes adicionales para reducir las predicciones incorrectas en la clase “no_front”. Esto podría implicar una mayor afinación del modelo o el uso de un conjunto de datos más representativo que capture mejor las variaciones en las posturas no frontales.

FIGURA N° 4.1: Matriz de confusión para el mejor modelo Resnet18

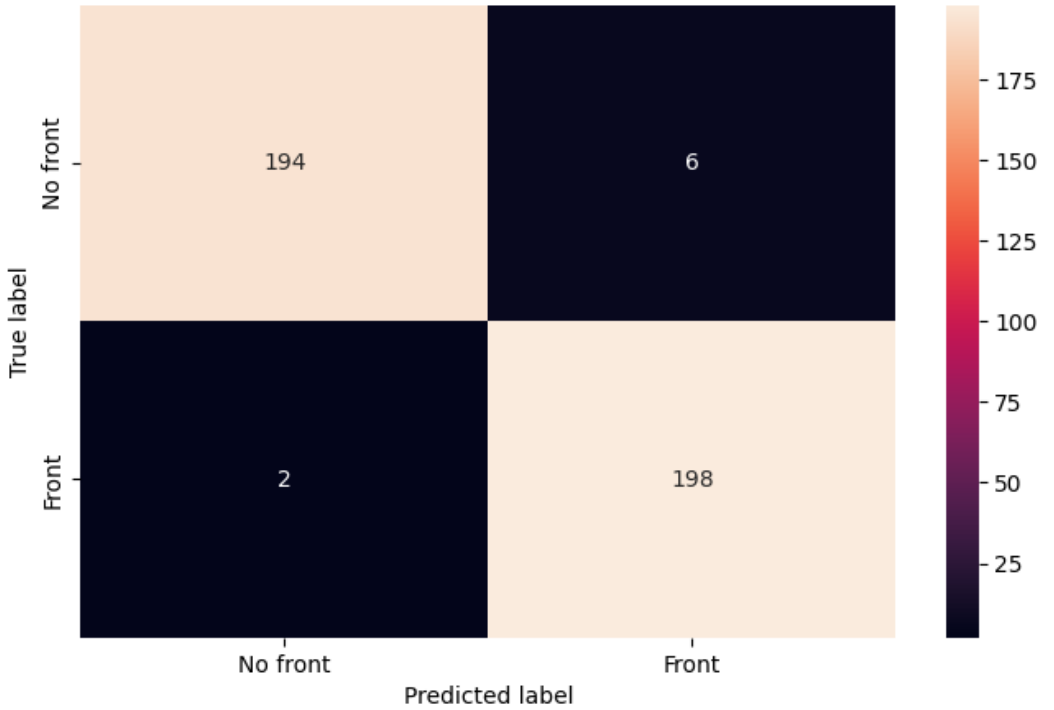
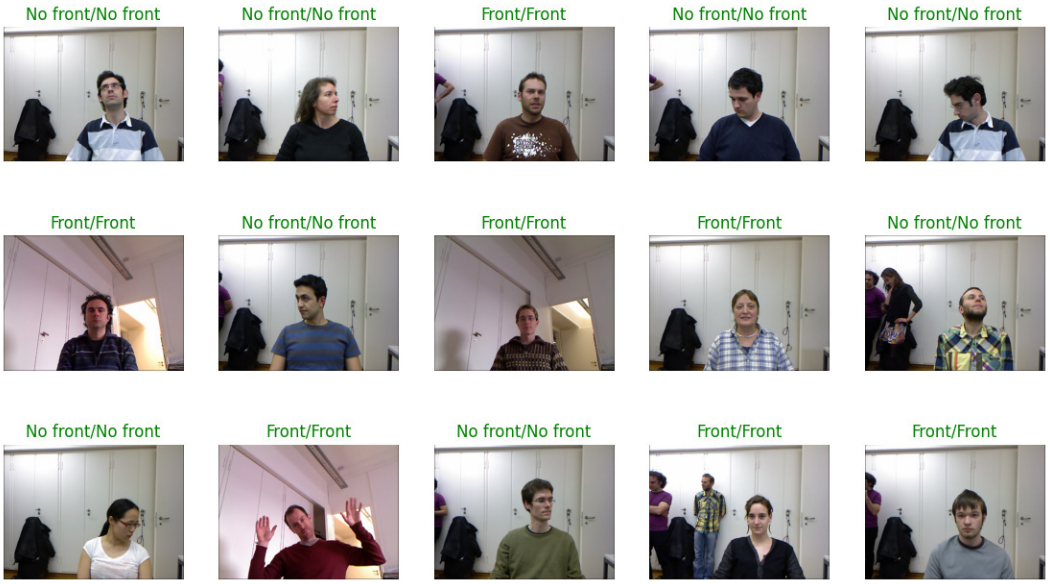


FIGURA N° 4.2: Etiquetas vs predicciones de algunas imágenes aleatorias de dataset Biwi (test)



4.2 Reconocimiento de expresiones faciales

La tabla Tabla N° 4.3 presenta los indicadores de rendimiento obtenidos por las arquitecturas ResNet18, DenseNet y AlexNet, evaluados con los datos de prueba del dataset FER-2013. En los resultados, ResNet18 obtuvo un menor loss, con un valor de 1.06724, no obstante, su F1-score fue de 64.56 % siendo inferior al de DenseNet, que alcanzó un 67.177 %, aunque su loss fue ligeramente superior a la de ResNet18, con un valor de 1.3087. Por lo tanto, el modelo basado en la arquitectura DenseNet resultó el mejor entre los tres.

Sin embargo, para aplicaciones en tiempo real que requieren alta velocidad de procesamiento o que operan en hardware limitado, el modelo ResNet es especialmente adecuado, por su menor tamaño en comparación con Densenet que tiene una arquitectura más compleja, como se muestra en la Figura N° 4.4, permitiendo un uso más eficiente de los recursos del sistema, lo que resulta en tiempos de inferencia más rápidos.

TABLA N° 4.3: Indicadores obtenidos por arquitectura, para dataset FER-2013

Modelo	Loss	F1 Score
Resnet18	**1.06724**	0.6456
Densenet	1.3087	**0.67177**
Alexnet	1.41357	0.6266

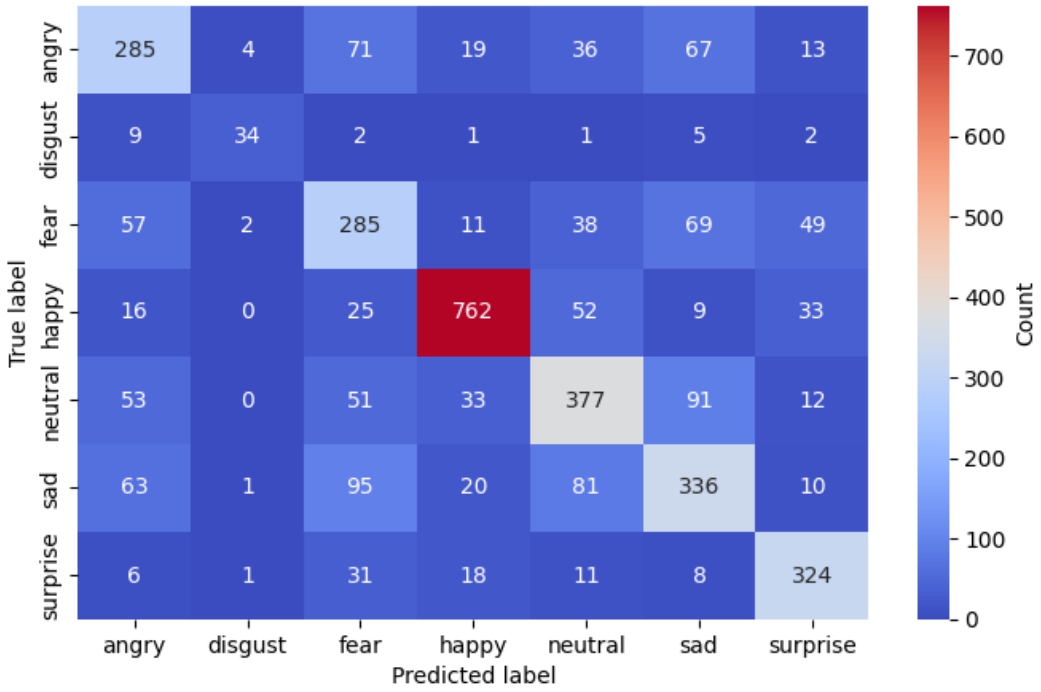
En contraste, AlexNet mostró el peor rendimiento, con un loss de 1.41357 y un F1-score de 62.66 %. Cabe destacar que AlexNet fue el único modelo que no mejoró su rendimiento al aplicar transfer learning, a diferencia de ResNet18 y DenseNet, que lograron mejoras significativas con esta técnica.

TABLA N° 4.4: Tamaño de los mejores modelos en formato ONNX, para el reconocimiento de expresiones faciales

Modelo	Tamaño
Resnet18	**42.6 MB**
Densenet	101 MB

Como se muestra en la Figura N° 4.3, el uso de class weight en el modelo DenseNet mejoró significativamente el rendimiento en la clasificación de expresiones faciales, especialmente en clases menos representadas como disgust y fear. Esta técnica ayudó a reducir la tasa de falsos negativos y falsos positivos, proporcionando un enfoque más equilibrado, lo que se tradujo en un aumento en el número de aciertos para estas clases en comparación con el estudio anterior considerado como antecedente (Singh y Nasoz, 2020). Los resultados indican que el modelo fue más efectivo al prestar atención a las categorías con menos ejemplos, logrando así un rendimiento general más alto.

FIGURA N° 4.3: Matriz de confusión para el mejor modelo DenseNet



En la Figura N° 4.4 se muestran 15 imágenes al azar del dataset FER-2013 (test), donde se visualizan las etiquetas y las predicciones realizadas por el modelo. Evidenciando que al alcanzar un 67.177 % de F1 Score, queda un 32.823 % de error que se refleja en las 3 imágenes clasificadas incorrectamente para las 15 imágenes random.

FIGURA N° 4.4: Etiquetas vs predicciones de algunas imágenes aleatorias del dataset FER-2013 (test)

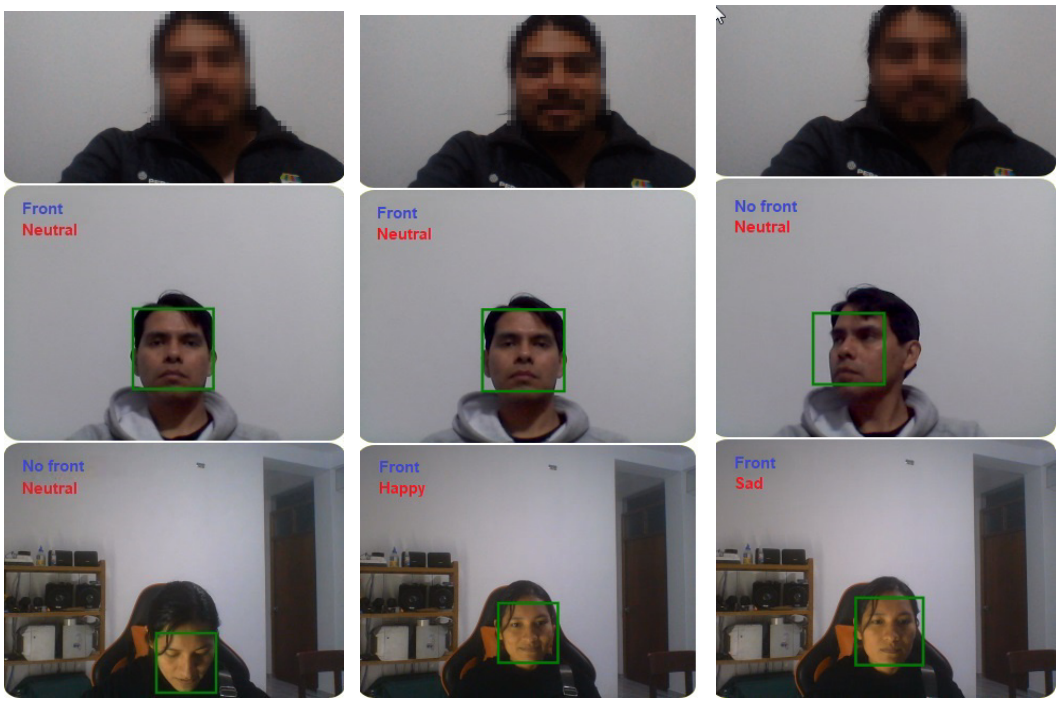


4.3 Prueba en tiempo real

En la Figura N° 4.5 se presentan los resultados obtenidos para el usuario administrador, que se conectó con dos usuarios. En la Figura N° 4.5a, el usuario 1 aparece como “Front” con la expresión facial “Neutral”, mientras que el usuario 2 es clasificado como “No front”, con la expresión facial “Neutral”. En la Figura N° 4.5b, el usuario 1 se mantiene como “Front” con la expresión facial “Neutral”, mientras que el usuario 2 es etiquetado como “Front” con la expresión facial “Happy”. Por último, en la Figura N° 4.5c, el usuario 1 es etiquetado como “No front” con la expresión facial “Neutral”, mientras que el usuario 2 como “Front” y la expresión facial “Sad”.

Como se observa en la Figura N° 4.5, se logró mostrar correctamente las etiquetas correspondientes. Cuando un usuario no está frontal a la cámara, se le asigna la etiqueta “No front”, sin embargo, cuando un usuario está frontalmente a la cámara, se muestra la etiqueta “Front” en color azul. Para las expresiones faciales se muestra la etiqueta en color rojo con la expresión facial detectada.

FIGURA N° 4.5: Prueba de los modelos en tiempo real



(a) Prueba1 (b) Prueba2 (c) Prueba3

La hipótesis planteada se confirmó, ya que los modelos se integraron exitosamente en la aplicación web de videoconferencia y las etiquetas se mostraron en tiempo real, sin retrasos perceptibles. No obstante, se observaron casos en los que el sistema etiquetaba como “no front” a pesar de que la cabeza del usuario1 estaba orientada hacia la cámara. Este comportamiento contradice lo esperado, como se muestra en la Figura N° 4.1, donde se preveía que la etiqueta “front” se asignara con mayor precisión que “no front”. Esta inconsistencia puede deberse a la falta de diversidad en el conjunto de datos, que podría beneficiarse de una mayor variabilidad en cuanto a personas, fondos, ambientes e iluminación. Una alternativa para mejorar la precisión del modelo sería enfocarse en analizar exclusivamente la imagen de la cabeza, recortando los hombros y el fondo, lo que permitiría que el modelo se centre en las características clave de la postura de la cabeza sin distracciones externas, tal como se trabajó con el reconocimiento de expresiones faciales. En el caso del reconocimiento de expresiones faciales, hubo una mayor facilidad para reconocer las expresiones “happy” y “neutral”, mientras que las otras expresiones fueron más difíciles de detectar.

CONCLUSIONES

Para el reconocimiento de expresiones faciales, se aplicaron diversos modelos de DL con el objetivo de determinar cuál ofrece el mejor F1-score y seleccionar el más adecuado para su integración en un sistema de tiempo real. Al comparar las métricas de F1-score de las técnicas empleadas, el modelo DenseNet alcanzó el mejor resultado con un 67.177 %. Este valor no superó el 70 % como se tenía previsto en la hipótesis, posiblemente debido al desbalance del conjunto de datos FER-2013 (a pesar del uso de class weights), al hecho de que las imágenes están en escala de grises y tienen un tamaño reducido (48x48 píxeles), además de la dificultad de clasificar imágenes muy similares que pertenecen a diferentes clases. A pesar de estas limitaciones, el modelo DenseNet y Resnet18 obtuvieron resultados aceptables.

Para el reconocimiento de la postura de la cabeza, también se aplicaron diversos modelos de DL con el objetivo de determinar cuál ofrece el mejor accuracy. Al comparar los indicadores accuracy y loss de las técnicas empleadas, el modelo Resnet18 alcanzó el mayor accuracy con un 98.5 %, superando el 90 % como se tenía previsto en la hipótesis.

El objetivo de integrar los modelos para la clasificación de la postura de la cabeza y las expresiones faciales en tiempo real se logró exitosamente a través de una aplicación web de videoconferencia, permitiendo la conexión de un administrador y dos usuarios simultáneamente. La elección de los modelos resultó adecuada, destacando la selección de ResNet18 sobre DenseNet, ya que, aunque ambas arquitecturas mostraron un rendimiento similar, ResNet18 demostró ser más conveniente para la prueba en tiempo real por su ligereza y eficiencia. Esta decisión facilitó la implementación en un entorno de videoconferencia, optimizando los recursos y la experiencia de uso en tiempo real.

RECOMENDACIONES

Debido al bajo F1-Score obtenido al clasificar expresiones faciales en el conjunto de datos FER-2013, se recomienda experimentar con otros conjuntos de datos que estén balanceados, sean de mayor tamaño y contengan imágenes a color.

Aunque para el dataset Biwi, Resnet18 fue suficiente para alcanzar un buen resultado, existen muchas otras arquitecturas para probar según el dataset (calidad y cantidad) y caso aplicado. Por ejemplo, puede buscarse minimizar los recursos computacionales y se deba usar una arquitectura con una red más pequeña pudiendo ser ejecutada en la memoria disponible y/o se ejecute más rápido.

Se recomienda evaluar los recursos informáticos disponibles, ya que estos influyen en la etapa de entrenamiento según las arquitecturas que se vayan a evaluar, por ejemplo, en este estudio Densenet y Alexnet requirieron más recursos y tiempo de ejecución que Resnet18.

Finalmente, así como se realizó en este estudio, se recomienda probar los resultados en una aplicación en tiempo real para validar que se obtengan los resultados esperados en un entorno práctico. Aunque el modelo pueda mostrar métricas muy favorables, es importante recordar que estas métricas se calculan en función del conjunto de datos utilizado en las pruebas. En un escenario real, pueden surgir diversas complicaciones que no se manifiestan al trabajar únicamente con los datos de entrenamiento y evaluación.

REFERENCIAS

Behera, A., Gidney, A. G., Wharton, Z., Robinson, D., y Quinn, K. (2019). A cnn model for head pose recognition using wholes and regions. En *2019 14th ieee international conference on automatic face & gesture recognition (fg 2019)* (pp. 1–2).

Bhardwaj, A., Di, W., y Wei, J. (2018). *Deep learning essentials: Your hands-on guide to the fundamentals of deep learning and neural network modeling*. Packt Publishing Ltd.

Bosch, A., Casas, J., y Lozano, T. (2019). *Deep learning: principios y fundamentos*. Editorial UOC.

Brownlee, J. (2019). *Deep learning for computer vision: image classification, object detection, and face recognition in python*. Machine Learning Mastery.

Chen, S., Zhang, Y., Yin, B., y Wang, B. (2021). Trfh: towards real-time face detection and head pose estimation. *Pattern Analysis and Applications*, *24*, 1745–1755. Descargado de https://doi.org/10.1007/s10044-021-01026-3

Chollet, F. (2018). *Deep learning with python*.

Contreras, L. E., Fuentes, H. J., y Rodríguez, J. I. (2020). Predicción del rendimiento académico como indicador de éxito/fracaso de los estudiantes de ingeniería, mediante aprendizaje automático. *Formación universitaria*, *13*(5), 233–246.

Dalianis, H. (2018). Evaluation metrics and evaluation. *Clinical Text Mining: secondary use of electronic patient records*, 45–53. Descargado de https://doi.org/10.1007/978-3-319-78503-5_6 doi: 10.1007/978-3-319-78503-5_6

Darwin, C. (1872). *The expression of the emotions in man and animals*.

Ekman, P. (1994). Strong evidence for universals in facial expressions: a reply to russell's mistaken critique. *Psychological Bulletin*. doi: doi:10.1037/0033-2909.115.2.268

Fanelli, G., Dantone, M., Gall, J., Fossati, A., y Van Gool, L. (2013, February). Random forests for real time 3d face analysis. *Int. J. Comput. Vision*, *101*(3), 437–458.

García, A. (2012). *Inteligencia artificial. fundamentos, práctica y aplicaciones*. Rc Libros.

Gegúndez, M., y Pérez, I. (2021). *Deep learning: fundamentos, teoría y aplicación*. Universidad de Huelva.

Géron, A. (2022). *Hands-on machine learning with scikit-learn, keras, and tensorflow*. O'Reilly Media.

Goodfellow, I. J., Erhan, D., Luc Carrier, P., Courville, A., Mirza, M., Hamner, B., … Bengio, Y. (2015). Challenges in representation learning: A report on three machine learning contests. *Neural Networks*, *64*, 59-63. Descargado de https://www.sciencedirect.com/science/article/pii/S0893608014002159 (Special Issue on "Deep Learning of Representations") doi: https://doi.org/10.1016/j.neunet.2014.09.005

Grandini, M., Bagli, E., y Visani, G. (2020). Metrics for multi-class classification: an overview. *arXiv preprint arXiv:2008.05756*.

Hearty, J. (2016). *Advanced machine learning with python: Solve challenging data science problems by mastering cutting-edge machine learning techniques in phyton. packt publishing.* Packt Publishing Ltd.

Hernández, A. Z., Rosales, G. A. G., Santiago, H. J. J., y Lee, M. M. (2022). Métricas de rendimiento para evaluar el aprendizaje automático en la clasificación de imágenes petroleras utilizando redes neuronales convolucionales. *Ciencia Latina Revista Científica Multidisciplinar*, *6*(5), 4624–4637.

Huang, G., Liu, Z., van der Maaten, L., y Weinberger, K. Q. (2018). *Densely connected convolutional networks.* Descargado de http://arxiv.org/abs/1608.06993

Hurwitz, J., y Kirsch, D. (2018). *Machine learning for dummies®, ibm limited edition published by john wiley & sons, inc. 111 river st*.

Keerthana, K., D.Pradeep, y Vanathi, D. B. (2020). Learner's engagement analysis for e-learning platform.. Descargado de https://api.semanticscholar.org/CorpusID:214587777

Krizhevsky, A., Sutskever, I., y Hinton, G. E. (2012, may). Imagenet classification with deep convolutional neural networks. *Commun. ACM*, *60*(6), 84–90. Descargado de https://doi.org/10.1145/3065386 doi: 10.1145/3065386

Lecun, Y., Bottou, L., Bengio, Y., y Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278-2324. doi: 10.1109/5.726791

Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., y Matthews, I. (2010). 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* (p. 94-101). Descargado de https://doi.org/10.1109/CVPRW.2010.5543262 doi: 10.1109/CVPRW.2010.5543262

Monkaresi, H., Bosch, N., Calvo, R. A., y D'Mello, S. K. (2017). Automated detection of engagement using video-based estimation of facial expressions and heart rate. *IEEE Transactions on Affective Computing*, *8*(1), 15-28. Descargado de https://doi.org/10.1109/TAFFC.2016.2515084 doi: 10.1109/TAFFC.2016.2515084

Moreno, L. (2020). *Análisis comparativo de arquitecturas de redes neuronales para la clasificación de imágenes* (Tesis de Master no publicada). Universidad Internacional de la Rioja.

Murphy-Chutorian, E., y Trivedi, M. M. (2008). Head pose estimation in computer vision: A survey. *IEEE transactions on pattern analysis and machine intelligence*, *31*(4), 607–626.

Paitán, H. Ñ., Mejía, E. M., Ramírez, E. N., y Paucar, A. V. (2014). *Metodología de la investigación cuantitativa-cualitativa y redacción de la tesis*. Ediciones de la U.

Pramerdorfer, C., y Kampel, M. (2016). *Facial expression recognition using convolutional neural networks: State of the art.* Descargado de https://arxiv.org/abs/1612.02903

Real Academia Española. (2020). *Diccionario de la lengua española* (23.ª ed.). Madrid: Espasa. (Disponible en línea en https://dle.rae.es/)

Rouhiainen, L. (2018). Inteligencia artificial. *Madrid: Alienta Editorial*, 20–21.

Samuel, A. L. (1959). Machine learning. *The Technology Review*, *62*(1), 42–45.

Saracco, R. A. (2012). Reconocimiento facial de las emociones básicas en pacientes con esquizofrenia, hermanos y controles. *UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO*. Descargado de https://hdl.handle.net/20.500.14330/TES01000682241 doi: 20.500.14330/TES01000682241

Shi, Y., Zhang, Z., Huang, K., Ma, W., y Tu, S. (2020). Human-computer interaction based on face feature localization. *Journal of Visual Communication and Image Representation*, *70*, 102740.

Singh, S., y Nasoz, F. (2020). Facial expression recognition with convolutional neural networks. En *2020 10th annual computing and communication workshop and conference (ccwc)* (pp. 0324–0328). doi: 10.1109/CCWC47524.2020.9031283

Subramanian, V. (2018). *Deep learning with pytorch: A practical approach to building neural network models using pytorch*. Packt Publishing Ltd.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., … Rabinovich, A. (2014). *Going deeper with convolutions.*

Terven, J., Cordova-Esparza, D. M., Ramirez-Pedraza, A., y Chavez-Urbiola, E. A. (2023). Loss functions and metrics in deep learning. a review. *arXiv preprint arXiv:2307.02694*.

T S, A., y Guddeti, R. M. R. (2018). Unobtrusive students' engagement analysis in computer science laboratory using deep learning techniques. En *2018 ieee 18th international conference on advanced learning technologies (icalt)* (p. 436-440). Descargado de https://doi.org/10.1109/ICALT.2018.00110 doi: 10.1109/ICALT.2018.00110

Ujir, H., Jee, E. M., Iqbal, M. F., Mun, Q. K., y Hipiny, I. (2021). Real-time driver's monitoring mobile application through head pose, drowsiness and angry detection. En *2021 8th international conference on computer and communication engineering (iccce)* (pp. 1–6).

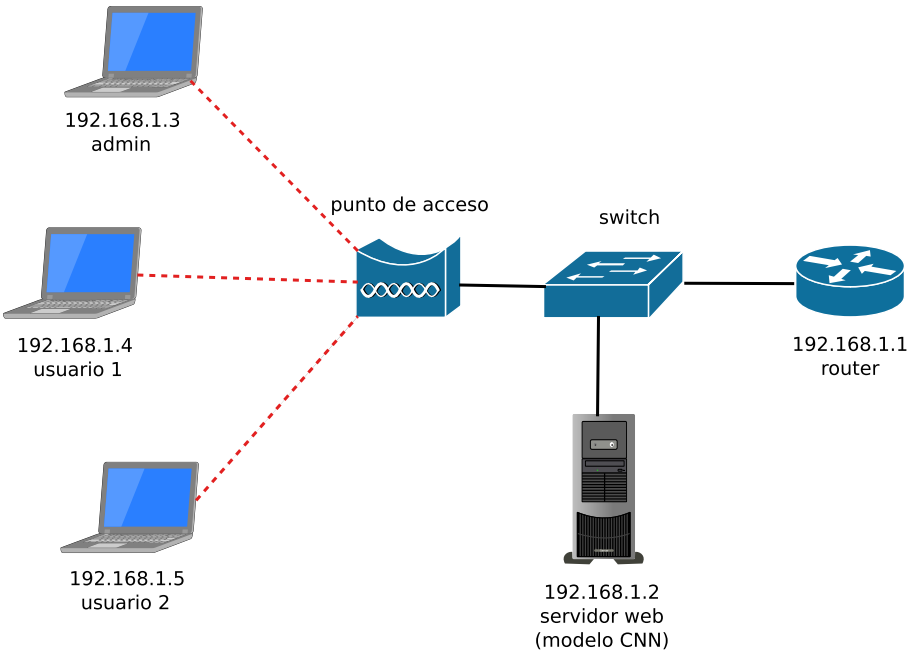
Visa, S., Ramsay, B., Ralescu, A. L., y van der Knaap, E. (2011). Confusion matrix-based feature selection. En *Midwest artificial intelligence and cognitive science conference.* Descargado de https://api.semanticscholar.org/CorpusID:3026044

Wu, J., y Trivedi, M. M. (2008). A two-stage head pose estimation framework and evaluation. *Pattern Recognition*, *41*(3), 1138–1158.

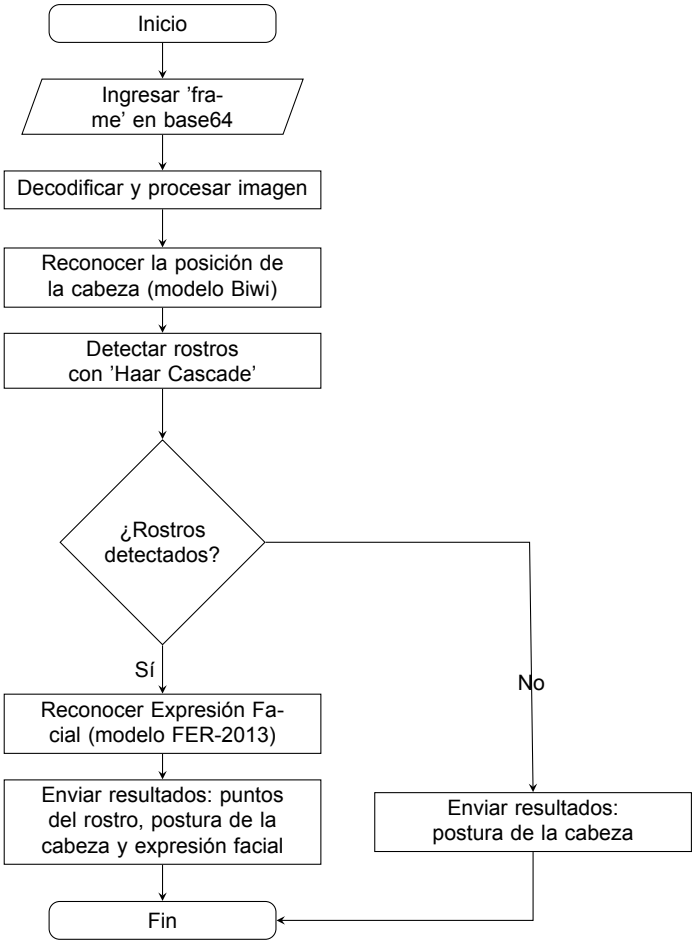
Yolcu, G., Oztel, I., Kazan, S., Oz, C., y Bunyak, F. (2020). Deep learning-based face analysis system for monitoring customer interest. *Journal of ambient intelligence and humanized computing*, *11*, 237–248.

ANEXOS

ANEXO 1: ESQUEMA DE RED PARA LA CONEXIÓN DE USUARIOS



ANEXO 2: DIAGRAMA DE FLUJO PARA PROCESAR Y CLASIFICAR LAS IMÁGENES OBTENIDAS DE LA CÁMARA WEB DEL USUARIO



"UNIVERSIDAD NACIONAL DE MOQUEGUA"

VICERRECTORADO DE INVESTIGACIÓN

FORMATO DE AUTORIZACIÓN DE PUBLICACIÓN EN EL REPOSITORIO INSTITUCIONAL UNAM

Formulario de autorización para la publicación electrónica en la página web del Repositorio Institucional de la Universidad Nacional de Moquegua, de conformidad con la Ley N° 30035 Ley que regula el Repositorio Nacional Digital de Ciencia, Tecnología e Innovación de Acceso Abierto y Resolución de Comisión Organizadora N° 041-2022-UNAM que aprueba el Reglamento que regula el Repositorio Institucional Digital de la UNAM.

1. DATOS DEL AUTOR

Nombres y Apellidos	: Diego Heder Percca Mamani
DNI/Carnet de Extranjería/Pasaporte N°	: DNI: 44752386
Correo electrónico	: dhpercca@gmail.com
Teléfono/Celular	: 969724155

2. GRADO, TITULO O ESPECIALIZACIÓN (Marque con una X)

Bachiller	Título Profesional	Magister	Doctor	Segunda Especialidad
	x			

3. TIPO DE PUBLICACIÓN (Marque con una X)

a) Articulo ( )
b) Tesis de pregrado (x)
c) Tesis de Maestría ( )
d) Tesis de Doctorado ( )
e) Monografía ( )
f) Reporte de caso científico o hallazgos ( )
g) Libro ( )
h) Parte de un libro ( )
i) Revisión ( )
j) Conferencia ( )
k) Presentación ( )
l) Artículo preliminar ( )
m) Articulo antes de ser publicado ( )
n) Artículo de prensa ( )
o) Documentos internos ( )
p) Imagen ( )
q) Documentos técnicos ( )
r) Ponencias presentadas en eventos científicos (Jornada, Congreso, Simposio Conferencias, etc.) ( )
s) Informes de suficiencia profesional ( )
t) Patentes producidas por académicos/as ( )

4. TIPO DE TRABAJO DE INVESTIGACIÓN (Marque con una X)

a) Tesis **(x)**
b) Trabajo de Investigación **( )**
c) Trabajo de suficiencia profesional **( )**
d) Trabajo académico **( )**

5. DATOS DE LA PUBLICACION

a) Título : Reconocimiento de Expresiones Faciales y la Postura de la Cabeza Usando Modelos de Deep Learning.
b) Campo del conocimiento OCDE (Indicar el código de clasificación OCDE de la obra : https://catalogos.concytec.gob.pe/vocabulario/ocde_ford.html) : 2.02.04
c) Fecha de publicación : ...2024

6. DATOS DEL ASESOR:

a) Nombres y Apellidos : Victor Yana Mamani
b) DNI/Pasaporte/Cédula : DNI: 02437887
c) ORCID : 0000-0003-0982-2353

FIRMA : 

DOCUMENTO DE IDENTIDAD : 44752386
NOMBRES Y APELLIDOS : Diego Heder Percca Mamani

ANEXO VIII

DECLARACIÓN JURADA DE ORIGINALIDAD DE TRABAJO DE INVESTIGACIÓN / TESIS / TRABAJO DE SUFICIENCIA PROFESIONAL

Yo, Diego Heder Percca Mamani ,
bachiller de la Escuela Profesional de:
Ingeniería de Sistemas e Informática ,
de la Universidad Nacional de Moquegua, identificado(a) con DNI Nº... 44752386 , autor(a) del Trabajo de Investigación / Tesis / Trabajo de suficiencia profesional denominado:
Reconocimiento de Expresiones Faciales y la Postura de la Cabeza Usando Modelos de Deep Learning.

Declaro bajo juramento que:

1. El trabajo de investigación es auténtico, siendo resultado de mi trabajo personal, que no se ha copiado, que no se ha utilizado ideas, formulaciones, citas integrales e ilustraciones diversas, sacadas de cualquier tesis, obra, artículo, memoria, etc., (en versión digital o impresa), sin mencionar de forma clara y exacta su origen o autor, tanto en el cuerpo del texto, figuras, cuadros, tablas u otros que tengan derechos de autor. Por tanto, el trabajo de investigación no ha sido plagiado total ni parcialmente.
2. El trabajo de investigación no ha sido autoplagiada, es decir, no ha sido publicada ni presentada anteriormente para obtener algún grado académico previo o título profesional.
3. Los datos presentados en los resultados son reales, no han sido falseados, ni duplicados, ni copiados y por tanto los resultados que se presentan en el trabajo de investigación constituirán aporte a la realidad investigada.

Soy consciente de que, el hecho de no respetar los derechos de autor y hacer plagio, es objeto de sanciones universitarias y/o legales, por lo que asumo cualquier responsabilidad que pudiera derivarse de irregularidades en el trabajo de investigación, así como de los derechos sobre la obra presentada.

Moquegua, ...27 de noviembre de 2024.....



Huella digital

Nombres y Apellidos: Diego Heder Percca Mamani
DNI: 44752386

"UNIVERSIDAD NACIONAL DE MOQUEGUA"

VICERRECTORADO DE INVESTIGACIÓN

LICENCIAS

Licencia Estándar

Bajo los siguientes términos, autorizo el depósito de mi trabajo de investigación en el Repositorio Institucional Digital de la UNAM.

Con la autorización de depósito da mi trabajo de investigación, otorgo a la Universidad Nacional de Moquegua, una Licencia No Exclusiva para reproducir. distribuir, comunicar al público, transformar (únicamente mediante su traducción a otros idiomas) y poner a disposición del público mi trabajo de investigación (incluido el resumen), en formato físico o digital, en cualquier medio, conocido o por conocerse, a través de los diversos servicios provistos por la Universidad, creados o por crearse, tales como el Repositorio Institucional Digital de la UNAM, Colección de Tesis, entre otros, en el Perú y en el extranjero, por el tiempo y veces que considere necesarias, y libre de remuneraciones.

En virtud de dicha Licencia, la UNIVERSIDAD NACIONAL DE MOQUEGUA, podrá reproducir mi trabajo de investigación en cualquier tipo de soporte y en más de un ejemplar, sin modificar su contenido, solo con los propósitos de seguridad, respaldo y preservación.

Declaro que el trabajo de investigación es una creación de mi autoría y exclusiva titularidad, o coautoría con titularidad compartida, y me encuentro facultado a conceder la presente licencia y, asimismo, garantizo que dicho trabajo de investigación no infringe derechos de autor de terceras personas.

La UNIVERSIDAD NACIONAL DE MOQUEGUA consignara el nombre del/los autor/es del trabajo de investigación, y no le hará ninguna modificación más de la permitida en la presente licencia.

Autorizo el depósito (marque con una X).

(X) **SI autorizo que se deposite inmediatamente.**

FIRMA : .....

DOCUMENTO DE IDENTIDAD : 44752386

NOMBRES Y APELLIDOS : Diego Heder Percca Mamani

FECHA : 27 / 11 / 2024



"UNIVERSIDAD NACIONAL DE MOQUEGUA"

VICERRECTORADO DE INVESTIGACIÓN

FORMATO DE AUTORIZACIÓN DE PUBLICACIÓN EN EL REPOSITORIO INSTITUCIONAL UNAM

Formulario de autorización para la publicación electrónica en la página web del Repositorio Institucional de la Universidad Nacional de Moquegua, de conformidad con la Ley N° 30035 Ley que regula el Repositorio Nacional Digital de Ciencia, Tecnología e Innovación de Acceso Abierto y Resolución de Comisión Organizadora N° 041-2022-UNAM que aprueba el Reglamento que regula el Repositorio Institucional Digital de la UNAM.

1. DATOS DEL AUTOR

Nombres y Apellidos : Gabriela Karina Velasquez Quispe
DNI/Carnet de Extranjería/Pasaporte N° : 77068244
Correo electrónico : gaby.kv13@hotmail.com
Teléfono/Celular : 971204324

2. GRADO, TITULO O ESPECIALIZACIÓN (Marque con una X)

Bachiller	Título Profesional	Magister	Doctor	Segunda Especialidad
	X			

3. TIPO DE PUBLICACIÓN (Marque con una X)

a) Articulo ( )
b) Tesis de pregrado (X)
c) Tesis de Maestría ( )
d) Tesis de Doctorado ( )
e) Monografía ( )
f) Reporte de caso científico o hallazgos ( )
g) Libro ( )
h) Parte de un libro ( )
i) Revisión ( )
j) Conferencia ( )
k) Presentación ( )
l) Artículo preliminar ( )
m) Articulo antes de ser publicado ( )
n) Artículo de prensa ( )
o) Documentos internos ( )
p) Imagen ( )
q) Documentos técnicos ( )
r) Ponencias presentadas en eventos científicos (Jornada, Congreso, Simposio Conferencias, etc.) ( )
s) Informes de suficiencia profesional ( )
t) Patentes producidas por académicos/as ( )

4. TIPO DE TRABAJO DE INVESTIGACIÓN (Marque con una X)

a) Tesis (X)
b) Trabajo de Investigación ( )
c) Trabajo de suficiencia profesional ( )
d) Trabajo académico ( )

5. DATOS DE LA PUBLICACION

a) Título : Reconocimiento de expresiones faciales y la postura de la cabeza, usando modelos de Deep Learning
b) Campo del conocimiento OCDE (Indicar el código de clasificación OCDE de la obra : https://catalogos.concytec.gob.pe/vocabulario/ocde_ford.html) : .....
c) Fecha de publicación : 2024

6. DATOS DEL ASESOR:

a) Nombres y Apellidos : Victor Yana Mamani
b) DNI/Pasaporte/Cédula : 02437887
c) ORCID : 0000-0003-0982-2353

FIRMA : .....

DOCUMENTO DE IDENTIDAD : 77068244
NOMBRES Y APELLIDOS : Gabriela Karina Velasquez Quispe

3

ANEXO VIII

DECLARACIÓN JURADA DE ORIGINALIDAD DE TRABAJO DE INVESTIGACIÓN / TESIS / TRABAJO DE SUFICIENCIA PROFESIONAL

Yo, Gabriela Karina Velasquez Quispe ,
bachiller de la Escuela Profesional de:
Ingeniería de Sistemas e Informática ,
de la Universidad Nacional de Moquegua, identificado(a) con DNI N°... 77068244 , autor(a) del Trabajo de Investigación / Tesis / Trabajo de suficiencia profesional denominado:
"Reconocimiento de expresiones faciales y la postura de la cabeza usando modelos de Deep Learning."

Declaro bajo juramento que:

1. El trabajo de investigación es auténtico, siendo resultado de mi trabajo personal, que no se ha copiado, que no se ha utilizado ideas, formulaciones, citas integrales e ilustraciones diversas, sacadas de cualquier tesis, obra, artículo, memoria, etc., (en versión digital o impresa), sin mencionar de forma clara y exacta su origen o autor, tanto en el cuerpo del texto, figuras, cuadros, tablas u otros que tengan derechos de autor. Por tanto, el trabajo de investigación no ha sido plagiado total ni parcialmente.
2. El trabajo de investigación no ha sido autoplagiada, es decir, no ha sido publicada ni presentada anteriormente para obtener algún grado académico previo o título profesional.
3. Los datos presentados en los resultados son reales, no han sido falseados, ni duplicados, ni copiados y por tanto los resultados que se presentan en el trabajo de investigación constituirán aporte a la realidad investigada.

Soy consciente de que, el hecho de no respetar los derechos de autor y hacer plagio, es objeto de sanciones universitarias y/o legales, por lo que asumo cualquier responsabilidad que pudiera derivarse de irregularidades en el trabajo de investigación, así como de los derechos sobre la obra presentada.

Moquegua, 29 de Noviembre del 2024


Huella digital

Nombres y Apellidos: Gabriela Karina Velasquez Quispe
DNI: 77068244

3

"UNIVERSIDAD NACIONAL DE MOQUEGUA"

VICERRECTORADO DE INVESTIGACIÓN

LICENCIAS

Licencia Estándar

Bajo los siguientes términos, autorizo el depósito de mi trabajo de investigación en el Repositorio Institucional Digital de la UNAM.

Con la autorización de depósito da mi trabajo de investigación, otorgo a la Universidad Nacional de Moquegua, una Licencia No Exclusiva para reproducir. distribuir, comunicar al público, transformar (únicamente mediante su traducción a otros idiomas) y poner a disposición del público mi trabajo de investigación (incluido el resumen), en formato físico o digital, en cualquier medio, conocido o por conocerse, a través de los diversos servicios provistos por la Universidad, creados o por crearse, tales como el Repositorio Institucional Digital de la UNAM, Colección de Tesis, entre otros, en el Perú y en el extranjero, por el tiempo y veces que considere necesarias, y libre de remuneraciones.

En virtud de dicha Licencia, la UNIVERSIDAD NACIONAL DE MOQUEGUA, podrá reproducir mi trabajo de investigación en cualquier tipo de soporte y en más de un ejemplar, sin modificar su contenido, solo con los propósitos de seguridad, respaldo y preservación.

Declaro que el trabajo de investigación es una creación de mi autoría y exclusiva titularidad, o coautoría con titularidad compartida, y me encuentro facultado a conceder la presente licencia y, asimismo, garantizo que dicho trabajo de investigación no infringe derechos de autor de terceras personas.

La UNIVERSIDAD NACIONAL DE MOQUEGUA consignara el nombre del/los autor/es del trabajo de investigación, y no le hará ninguna modificación más de la permitida en la presente licencia.

Autorizo el depósito (marque con una X).

(X) **SI autorizo que se deposite inmediatamente.**

FIRMA : .....

DOCUMENTO DE IDENTIDAD : 77068244

NOMBRES Y APELLIDOS : Gabriela Karina Velasquez Quispe

FECHA : 29/11/2024



CERTIFICADO DE ANÁLISIS
magister

Tesis_UNAM (2)

7% Textos sospechosos

- < 1% Similitudes
- 0% similitudes entre comillas
- 0% entre las fuentes mencionadas
- 6% Idiomas no reconocidos

Nombre del documento: Tesis_UNAM (2).pdf
ID del documento: ac27b63a892612fc03432d0287e664323f04be74
Tamaño del documento original: 5,18 MB
Autores: []

Depositante: VICTOR YANA MAMANI
Fecha de depósito: 28/10/2024
Tipo de carga: interface
fecha de fin de análisis: 28/10/2024

Número de palabras: 19.111
Número de caracteres: 119.802

Ubicación de las similitudes en el documento:



Fuentes principales detectadas

N°	Descripciones	Similitudes	Ubicaciones	Datos adicionales
1	**123dok.net** | Analyse critique des résultats et comparaison avec la littérature https://123dok.net/article/analyse-critique-résultats-comparaison-littérature.1y991mvy	< 1%		Palabras idénticas: < 1% (46 palabras)
2	**pistaseducativas.celaya.tecnm.mx** https://pistaseducativas.celaya.tecnm.mx/index.php/pistas/article/viewFile/2091/1624	< 1%		Palabras idénticas: < 1% (43 palabras)

Fuentes con similitudes fortuitas

N°	Descripciones	Similitudes	Ubicaciones	Datos adicionales
1	**psisemadrid.org** | Reconocimiento de expresiones faciales en niños - PSISE https://psisemadrid.org/reconocimiento-de-expresiones-faciales-en-ninos/	< 1%		Palabras idénticas: < 1% (25 palabras)
2	**sedici.unlp.edu.ar** https://sedici.unlp.edu.ar/bitstream/handle/10915/157870/Documento_completo.pdf-PDFA.pdf?s...	< 1%		Palabras idénticas: < 1% (25 palabras)
3	**rua.ua.es** http://rua.ua.es/dspace/bitstream/10045/72503/1/tesis_jose_carlos_rangel_ortiz.pdf	< 1%		Palabras idénticas: < 1% (20 palabras)
4	**paperswithcode.com** | CK+ Dataset | Papers With Code https://paperswithcode.com/dataset/ck	< 1%		Palabras idénticas: < 1% (13 palabras)
5	**www.academia.edu** | (PDF) A two-stage head pose estimation framework and evalu... https://www.academia.edu/49722579/A_two_stage_head_pose_estimation_framework_and_evalu...	< 1%		Palabras idénticas: < 1% (15 palabras)

Fuentes mencionadas (sin similitudes detectadas)

Estas fuentes han sido citadas en el documento sin encontrar similitudes.

1. https://doi.org/10.1007/s10044-021-01026-3
2. https://doi.org/10.1007/978-3-319-78503-5_6
3. https://www.sciencedirect.com/science/article/pii/
4. https://doi.org/10.1016/j.neunet.2014.09.005
5. http://arxiv.org/abs/

DECENIO DE LA IGUALDAD DE OPORTUNIDADES PARA MUJERES Y HOMBRES
"AÑO DE LA RECUPERACIÓN Y CONSOLIDACIÓN DE LA ECONOMÍA PERUANA"

RESOLUCIÓN DE CONSEJO UNIVERSITARIO Nº 029-2025-CU-UNAM

Moquegua, 30 de enero de 2025

VISTOS:

El Acuerdo de Sesión Ordinaria de Consejo Universitario de fecha 30 de enero de 2025; Proveído N° 286-2025 de fecha 28 de enero de 2025 emitido por el Rectorado; Informe N° 011-2025-UNAM-R/SG de fecha de recepción 27 de enero de 2025 emitido por la Secretaría General; Carta N° 002-2025-UNAM/SEGE/GYT/SVR de fecha de recepción 23 de enero de 2025 emitido por la Especialista de Grados y Títulos; Sobre el otorgamiento del Título Profesional de Ingeniero de Sistemas e Informática, y;

CONSIDERANDO:

Que, la Constitución Política del Perú en su artículo 18º prevé que cada Universidad es autónoma en su régimen normativo, de gobierno, académico, administrativo y económico, y el artículo 8º de la Ley Nº 30220, Ley Universitaria, establece que la referida autonomía inherente a las Universidades, se ejerce de conformidad con la Constitución y las Leyes de la República e implica las organizaciones de su sistema académico, económico y administrativo, así como la administración de sus bienes y rentas, elaborar su presupuesto y aplicar sus fondos con la responsabilidad que impone la Ley;

Que, de conformidad con la Ley N° 28520 de fecha 23/05/2005, se creó la Universidad Nacional de Moquegua, cuyo artículo 1º dispone: "créase la Universidad Nacional de Moquegua como persona jurídica de derecho público interno, con sede en la ciudad de Moquegua, Provincia de Mariscal Nieto, Departamento de Moquegua". Asimismo, a través de la Resolución del Consejo Directivo N° 103-2017-SUNEDU/CD, el Consejo Directivo de la SUNEDU, le otorgó la Licencia Institucional a la Universidad Nacional de Moquegua; la misma que fue modificada a través de la Resolución del Consejo Directivo N° 135-2022-SUNEDU/CD;

Que, la Universidad Nacional de Moquegua, es una institución que brinda educación superior integral de calidad mediante una gestión académica y administrativa, centrada en los derechos humanos, con pertinencia social de las escuelas profesionales y posgrado, investigación científica y el ejercicio responsable del liderazgo universitario, vinculado al desarrollo de la región, el país y el mundo;

Que, en virtud a ello, mediante Resolución de Consejo Universitario Nº 001, 002 y 003-2025-CU-UNAM, de fecha 03 de enero de 2025, se oficializa la elección del Dr. Hugo Ruben Marca Maquera, como Rector, al Dr. Alejandro Manuel Ecos Espino como Vicerrector Académico y al Dr. Jhony Mayta Hancco como Vicerrector de Investigación de la Universidad Nacional de Moquegua;

Que, el artículo 58° de la Ley Universitaria, Ley Nº 30220, concordado con el artículo 14º del Estatuto de la Universidad Nacional de Moquegua, aprobado con Resolución de Comisión Organizadora Nº 578-2021-UNAM, establecen que el **Consejo Universitario es el máximo órgano de gestión, dirección y ejecución académica y administrativa de la universidad;**

Que, el artículo 44° de la Ley Nº 30220, Ley Universitaria, señala que las universidades otorgan los grados académicos de Bachiller, Maestro, Doctor y los Títulos Profesionales que correspondan, a nombre de la Nación (...);

Domicilio Fiscal: Prolongación Av., Ancash S/N - Moquegua
RUC: 20449347448 - Teléfono: (053) 670168
Página Web: www.unam.edu.pe

Que, el numeral 45.2 del artículo 45° de la Ley Nº 30220, Ley Universitaria, precisa **los requisitos mínimos para la obtención del título profesional, como son: el grado de Bachiller y la aprobación de una tesis** o trabajo de suficiencia profesional. Asimismo, establece que el título profesional solo se puede obtener en la universidad en la cual se haya obtenido el grado de bachiller;

Que, mediante Resolución de Comisión Organizadora Nº 1153-2021-UNAM, se aprobó el Reglamento de Grados y Títulos de la Universidad Nacional de Moquegua, el mismo que fue modificado a través de las Resoluciones de Comisión Organizadora N° 410; 453 y 890-2022, N° 880-2023-UNAM y N° 081, 484-2024-UNAM el cual establece los requisitos y norma los procedimientos para la obtención de los grados académicos y títulos profesionales conferidos por esta Casa Superior de Estudios;

Que, mediante la Carta N° 002-2025-UNAM/SEGE/GYT/SVR, el Especialista del Área de Grados y Títulos, remite a Secretaria General, la relación de tres (3) bachilleres de la Escuela Profesional de Ingeniería de Sistemas e Informática, quienes se encuentran aptos para la expedición del Título Profesional de Ingeniero de Sistemas e Informática, al haber cumplido satisfactoriamente con los requisitos legales y reglamentarios[1], exigidos en el Reglamento de Grados y Títulos de la Universidad Nacional de Moquegua, conforme lo señala la Dirección de Actividades y Servicios Académicos y la Dirección de la Escuela Profesional de Ingeniería de Sistemas e Informática, razón por la cual la Secretaría General incluye el expediente como punto de agenda para ser tratado en Sesión de Consejo Universitario de fecha 30 de enero de 2025;

Que, se ha establecido como atribución del Consejo Universitario de la UNAM, la facultad de conferir los grados académicos y los títulos profesionales (...), por lo que en mérito al pedido del Secretario General y debatido el caso en Sesión Ordinaria de fecha 30 de enero de 2025, se acordó por unanimidad **conferir a nombre de la Nación el Título Profesional de Ingeniero de Sistemas e Informática a tres (3) bachilleres de la Escuela Profesional de Ingeniero de Sistemas e Informática de la Facultad de Ingeniería y Arquitectura de la Universidad Nacional de Moquegua**;

[1] Artículo 7°. - Procedimiento y Trámite

Para obtener el Grado Académico, el egresado deberá iniciar el trámite presentando en su Escuela Profesional:

a) Solicitud dirigida al Director de la Escuela Profesional o quien haga sus veces, indicando la fecha y numero de comprobante de pago.
b) Dos (02) fotografías de frente, tamaño pasaporte, a color, con traje formal y fondo blanco: garantizando de esta forma la seguridad ciudadana.

La Dirección de Actividades y Servicios Académicos es la responsable de completar el expediente del egresado junto con las Escuela Profesional, en un plazo máximo de 7 días hábiles para su trámite correspondiente, con los siguientes documentos:

a) Constancia de Prácticas Preprofesionales expedido por el Director de la Escuela Profesional respectiva.
b) Certificado de estudios del idioma inglés o lengua nativa, nivel intermedio, expedido por el Centro de Idiomas de la Universidad Nacional de Moquegua o rendir el examen de suficiencia correspondiente en el Centro de Idiomas de la Universidad Nacional de Moquegua en caso de haber estudiado en otra institución.
c) Certificado de Estudios originales, debiendo acreditar el número de créditos exigidos por su Escuela Profesional.
d) Constancia de haber realizado Actividades Cocurriculares, expedida por la Dirección de Bienestar Universitario y la Dirección Proyección Social y Extensión Universitaria.
e) Constancia de "no adeudo" de Bienes a la Universidad, bienes a los Laboratorios Básicos y de Especialidad, material Bibliográfico; dichos documentos deberán ser refrendados por los responsables a cargo.
f) Copia simple de la Constancia de Egresado.
g) Copia simple de la Constancia de Inicio de Matricula.

La Dirección de Actividades y Servicios Académicos, previa verificación, deriva el expediente a la Oficina de Secretaría General para su trámite correspondiente.

Domicilio Fiscal: Prolongación Av., Ancash S/N - Moquegua
RUC: 20449347448 - Teléfono: (053) 670168
Página Web: www.unam.edu.pe

De conformidad con el artículo 59° de la Ley Universitaria, Ley Nº 30220, concordado con el artículo 16° del Estatuto de la Universidad Nacional de Moquegua, y estando a lo acordado en la I Sesión Ordinaria de fecha 30 de enero de 2025 de Consejo Universitario;

SE RESUELVE:

ARTÍCULO PRIMERO: CONFERIR, a nombre de la Nación el **TÍTULO PROFESIONAL DE INGENIERO DE SISTEMAS E INFORMÁTICA,** por la modalidad de Sustentación de Tesis, a los Bachilleres de la Escuela Profesional de Ingeniería de Sistemas de la Facultad de Ingeniería y Arquitecta de la Universidad Nacional de Moquegua, que se consigna en el listado siguiente:

1. **DIEGO HEDER PERCCA MAMANI**

   TESIS : "RECONOCIMIENTO DE EXPRESIONES FACIALES Y LA POSTURA DE LA CABEZA USANDO MODELOS DE DEEP LEARNING", cuyo Asesor fue el Dr. Víctor Yana Mamani.

2. **AMPARO LISBETH NUÑEZ GALINDO**

   TESIS : "EVALUACIÓN DE MODELOS PREDICTIVOS EMPLEANDO TÉCNICAS DE MACHINE LEARNIG PARA EL PRONÓSTICO DE LA VIOLENCIA FÍSICA CONTRA LA MUJER EN LA PROVINCIA DE ILO, AÑO 2021", cuyo Asesor fue el Dra. Vaneza Flores Gutierrez.

3. **GABRIELA KARINA VELASQUEZ QUISPE**

   TESIS : "RECONOCIMIENTO DE EXPRESIONES FACIALES Y LA POSTURA DE LA CABEZA USANDO MODELOS DE DEEP LEARNING", cuyo Asesor fue el Dr. Víctor Yana Mamani.

ARTÍCULO SEGUNDO: AUTORIZAR, a la Secretaría General de la Universidad Nacional de Moquegua, expedir y registrar los Diplomas correspondientes para su otorgamiento.

ARTÍCULO TERCERO: ENCARGAR al Área de Grados y Títulos de la Universidad Nacional de Moquegua, para que realice las acciones pertinentes conducentes, relacionadas con la inscripción de los grados y títulos en el Registro de Grados y Títulos de la Superintendencia Nacional de Educación Superior Universitaria - SUNEDU.

ARTÍCULO CUARTO: NOTIFICAR, de la presente Resolución a todas las instancias y dependencias administrativas y académicas de la Universidad Nacional de Moquegua, para su conocimiento, cumplimiento y demás fines pertinentes.

ARTÍCULO QUINTO: ENCARGAR, a la Oficina de Tecnologías de la Información, la publicación de la presente Resolución en el portal de transparencia de la Universidad Nacional de Moquegua.

REGÍSTRESE, COMUNÍQUESE, CÚMPLASE Y ARCHÍVESE.

DR. HUGO RUBEN MARCA MAQUERA
Rector
Universidad Nacional de Moquegua

ABG. GUILLERMO S. KUONG CORNEJO
Secretario General
Universidad Nacional de Moquegua

Domicilio Fiscal: Prolongación Av., Ancash S/N - Moquegua
RUC: 20449347448 - Teléfono: (053) 670168
Página Web: www.unam.edu.pe