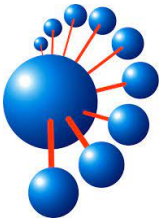
DEPARTAMENTO DE INGENIERÍA
INFORMÁTICA
Y CIENCIAS DE LA COMPUTACIÓN
FACULTAD DE INGENIERÍA
UNIVERSIDAD DE CONCEPCIÓN



Adaptación y Análisis de técnicas para el reconocimiento emocional en tiempo real con Jetson Xavier NX

por

Vicente Alejandro Rodríguez Alfaro

Memoria presentada para la obtención del título de

Ingeniero Civil Informático

Patrocinante: Julio Erasmo Godoy Del Campo

Concepción, Mayo 2026

Agradecimientos

El desarrollo de esta memoria de título y esta etapa universitaria no habrían sido posibles sin el apoyo, la paciencia y el cariño de una gran red de personas que me acompañaron en el proceso.

En primer lugar, quiero expresar mi sincero agradecimiento a mi profesor guía, Julio Godoy, por su orientación, disposición y por compartir su conocimiento y experiencia conmigo. Gracias por guiar el rumbo de este proyecto y por el constante apoyo a lo largo de su desarrollo.

A mis padres, Tomás y Marisol, por ser mi pilar fundamental. Gracias por su apoyo incondicional, por los valores entregados, por creer siempre en mí y por brindarme las herramientas para llegar hasta este momento. Este logro es tan mío como de ustedes.

A mis hermanos, Francisco e Ignacio, por el compañerismo, las risas y por estar siempre presentes cuando los necesito, siendo una parte esencial de mi vida y mi formación.

A mi pareja, Steven, por ser mi refugio y mi mayor fuente de ánimo. Gracias por tu compañía incondicional, por tu paciencia infinita durante las largas jornadas de código y escritura, y por celebrar cada pequeño avance a mi lado.

A mis amigos, Giorgio, Felipe, Sebastián, Vicente, Yerko y Vanessa. Gracias por aligerar la carga académica, por las conversaciones, los momentos compartidos y por hacer que los años universitarios hayan sido una experiencia memorable.

Finalmente, quiero extender un agradecimiento muy especial a todas las personas que participaron voluntariamente en la validación experimental de este trabajo. Gracias por regalarme su tiempo y sus expresiones faciales para que el sistema pudiera cobrar vida. A Yerko, Tomas E., Steven, Nacho, Nico, Ignacio, Tomás, Camilo, Víctor, Carlos, Martina, Vale, Jeshu, Dani, Mare, Vale Serón, Fer, Carla, Pancho, Catalina, Alondra, María José, Montse, y

a todos aquellos que, aunque prefirieron no aparecer en esta sección, fueron piezas fundamentales e invaluables para el éxito de este experimento.

A todos ustedes, muchas gracias.

Índice general

Capítulo 1

Introducción

1.1. Contexto y motivación

El reconocimiento automático de emociones faciales (FER, por sus siglas en inglés: *Facial Expression Recognition*) se ha consolidado como un pilar fundamental dentro de la Computación Afectiva, una disciplina que busca cerrar la brecha de decisión entre el estado emocional humano y los sistemas digitales. En el contexto actual de la Industria 4.0, la capacidad de las máquinas para percibir e interpretar señales no verbales es crítica para aplicaciones que van desde la robótica social hasta los sistemas de monitoreo de seguridad en la conducción [34].

Para llevar a cabo esta tarea de reconocimiento, la literatura se ha sustentado transversalmente en el modelo de Paul Ekman, el cual postula la existencia de siete emociones básicas y universalmente reconocibles: alegría, tristeza, ira, miedo, sorpresa, disgusto y neutralidad [11]. Este enfoque ha guiado el desarrollo de los sistemas de visión computacional aplicados al análisis facial desde sus inicios [37].

En el plano de la implementación, el estado del arte en FER ha estado dominado por las Redes Neuronales Convolucionales (CNNs), tales como ResNet, VGG o MobileNet. Estas arquitecturas operan construyendo jerarquías de características desde bordes simples hasta formas complejas [22] de la imagen. Sin embargo, las emociones humanas a menudo se definen por relaciones semánticas globales y sutiles distorsiones musculares que pueden perderse en las operaciones de *pooling* agresivas típicas de las CNNs.

Recientemente, la visión por computador ha experimentado un cambio de

paradigma hacia los *Vision Transformers* (ViT). A diferencia de las CNNs, los ViT procesan la imagen como una secuencia de parches, utilizando mecanismos de auto-atención (*Self-Attention*) que otorgan al modelo un campo receptivo global desde la primera capa [8]. En particular, el modelo CLIP (*Contrastive Language-Image Pre-training*), desarrollado por OpenAI, ha demostrado que el pre-entrenamiento con pares imagen-texto a gran escala permite aprender representaciones visuales robustas y semánticamente ricas, capaces de generalizar mejor ante variaciones de iluminación y postura [32].

Sin embargo, la implementación de estas arquitecturas basadas en Transformers presenta un desafío debido a su alto costo computacional. Modelos como CLIP ViT-B/32 requieren una capacidad de procesamiento que generalmente restringe su uso a servidores con GPUs de alto rendimiento o a la nube. Esta dependencia de la nube introduce latencia y problemas de privacidad que son importantes para aplicaciones de interacción en tiempo real. Por lo cual, existe una necesidad de trasladar esta tecnología a tiempo real.

El presente trabajo de título aborda esta problemática mediante la implementación de un sistema de reconocimiento de emociones en tiempo real sobre la plataforma NVIDIA Jetson Xavier NX [5]. La motivación principal es demostrar que, mediante técnicas de *Fine-Tuning* y una ingeniería de software eficiente es posible ejecutar modelos de lenguaje-visión en dispositivos de recursos limitados.

El sistema propuesto se entrena y evalúa utilizando una fusión de los conjuntos de datos AffectNet, RAF-DB y FER2013 [25], con el fin de evitar fallos al clasificar expresiones espontáneas, en las cuales la asimetría facial y la baja intensidad son predominantes debido al sesgo inherente de cada conjunto de datos por separado.

Por último, en este trabajo se validará experimentalmente el sistema en tiempo real con 32 sujetos de prueba. Cada evaluación constará de dos fases (una guiada y otra libre), en las cuales se buscará que los participantes representen las siete emociones clásicas, con el fin de aportar evidencia empírica sobre la viabilidad de los Transformers en entornos operativos reales.

1.2. Planteamiento del problema

La implementación de sistemas de reconocimiento de emociones faciales (FER) en entornos no controlados enfrenta una serie de desafíos que van más allá de la precisión algorítmica. Este proyecto aborda una problemática que

se divide en tres aspectos principales: las restricciones computacionales del hardware de borde (Edge), el desbalance en los datos de entrenamiento y la diferencia semántica entre las expresiones faciales posadas y las espontáneas. Cada uno de estos puntos se analizan en profundidad a continuación:

1.2.1. Restricciones de hardware y complejidad computacional

A pesar de que las arquitecturas basadas en Transformers, como CLIP, han superado a las Redes Neuronales Convolucionales (CNNs) en tareas de generalización semántica, entendida como la capacidad del modelo de interpretar el significado de las expresiones faciales más allá de los ejemplos específicos observados durante el entrenamiento, permitiendo reconocer emociones en condiciones nuevas, con variaciones de iluminación, pose o sujeto, su costo computacional es considerablemente alto para dispositivos embebidos.

El mecanismo de *Self-Attention* posee una complejidad cuadrática $O(N^2)$ respecto al número N de parches de la imagen, lo que implica una carga importante de operaciones de multiplicación de matrices [18].

Desplegar un modelo CLIP ViT-B/32 en una plataforma como la NVIDIA Jetson Xavier NX, que opera con un capacidad energética y memoria limitada, genera un cuello de botella no menor. El desafío principal a grandes rasgos consiste en lograr una inferencia en tiempo real (o cercana a ella) sin recurrir a técnicas de compresión que degraden la capacidad del modelo para detectar micro-expresiones sutiles.

1.2.2. Desbalance de los datos de entrenamiento

Los conjuntos de datos utilizados para entrenar modelos capaces de operar en el mundo real, tales como AffectNet y RAF-DB, presentan una distribución de clases de *larga cola* (*long-tailed distribution*). Emociones de alta prevalencia y baja intensidad, como "Neutral" y "Felicidad", están sobrerrepresentadas, constituyendo a menudo más del 50 % de las muestras, mientras que emociones críticas para la seguridad y el análisis psicológico, como "Miedo" y "Disgusto", son escasas [12]. El entrenamiento convencional con la función de pérdida de entropía cruzada (*Cross-Entropy*) en estos datos sesgados induce al modelo a converger hacia una solución trivial: predecir la clase mayoritaria para minimizar el error global, sacrificando la sensibilidad

ante las clases minoritarias. Esto resulta en sistemas que parecen precisos en métricas globales, pero que son funcionalmente inútiles para detectar estados afectivos negativos o de alerta.

1.2.3. Brecha fisiológica: expresiones posadas vs espontáneas

Existe una divergencia fundamental entre las expresiones faciales generadas voluntariamente (posadas) y las involuntarias (espontáneas). Los modelos de *Deep Learning* entrenados predominantemente con datasets de actores (como CK+ o partes de FER2013) aprenden patrones exagerados y estereotipados, lo que limita su capacidad de generalización a expresiones espontáneas en entornos reales [43]. Al desplegar estos modelos en un entorno real con una cámara web, fallan al intentar clasificar reacciones genuinas, las cuales suelen ser más sutiles, rápidas (micro-expresiones) y, a menudo, asimétricas.

Así, la combinación de restricciones de hardware, sesgo en los datos y la brecha entre lo posado y lo espontáneo define el núcleo del problema que este trabajo busca resolver. Abordar estos desafíos de manera efectiva requiere el uso de enfoques capaces de modelar relaciones globales en la imagen sin disparar el costo computacional, sentando la base de la propuesta de objetivos que se detallan a continuación.

Capítulo 2

Objetivos

2.1. Objetivo general

Adaptar y optimizar un modelo de aprendizaje de tipo *Zero-Shot*, para el reconocimiento de emociones faciales en tiempo real mediante tecnicas de transferencia de aprendizaje, evaluando comparativamente su rendimiento, precisión y eficiencia en hardware de borde con recursos limitados (Jetson Xavier NX).

2.2. Objetivos específicos

1. Analizar la arquitectura del modelo CLIP y otros modelos de FER para determinar su viabilidad de implementación en hardware de recursos limitados.

2. Caracterizar el rendimiento base y las capacidades de hardware/software del dispositivo Jetson Xavier NX.

3. Implementar un pipeline de procesamiento de video en tiempo real que integre la captura de imagen, preprocesamiento e inferencia del modelo optimizado en el dispositivo objetivo.

4. Evaluar y comparar cuantitativamente el rendimiento del sistema implementado en el dispositivo, utilizando métricas de precisión (accuracy, precision, Recall y F1-score) y latencia (ms).

5. Interpretar y documentar los resultados obtenidos, discutiendo el *trade-off* entre precisión y eficiencia, y su relevancia para aplicaciones prácticas del reconocimiento emocional en el hardware de borde.

Capítulo 3

Marco teórico

Para dar cumplimiento a los objetivos planteados en este trabajo, centrados en la adaptación, optimización y validación en tiempo real de un sistema de reconocimiento de emociones faciales (FER) en hardware de borde, es fundamental establecer un sustento teórico y metodológico robusto. Para la transición desde los entornos controlados hacia la ejecución en tiempo real en dispositivos con recursos limitados es necesario comprender la naturaleza psicológica y matemática de las expresiones humanas, la evolución algorítmica que fundamenta a los modelos de lenguaje-visión y las estrategias de optimización computacional. Con este propósito, la discusión de este capítulo se articula comenzando por los fundamentos afectivos y las teorías psicológicas que definen y delimitan el espacio de estados emocionales. Posteriormente, se analiza la evolución de las técnicas de visión artificial, transitando desde los descriptores manuales y geométricos hasta el paradigma del aprendizaje profundo, con especial énfasis en las arquitecturas basadas en mecanismos de atención y el modelo multimodal CLIP. Finalmente, se examinan las metodologías de transferencia de aprendizaje, las características de las bases de datos utilizadas, las restricciones físicas asociadas al hardware embebido y las arquitecturas híbridas de optimización, concluyendo con la definición de las métricas multidimensionales necesarias para evaluar el rendimiento global del sistema propuesto.

3.1. Computación afectiva y reconocimiento automático de emociones

En primer lugar, la computación afectiva (*Affective Computing*) que corresponde a aquella área que estudia el diseño de sistemas capaces de detectar, interpretar y eventualmente responder a señales afectivas humanas. Desde su formulación clásica, esta área se ha entendido como una intersección entre inteligencia artificial, psicología y procesamiento de señales, con aplicaciones en diversas áreas como: salud, educación, robótica, asistencia, seguridad y analítica del comportamiento entre otras [30, 13].

Dentro de este campo, el reconocimiento de emociones constituye una tarea central. Su objetivo es inferir categorías emocionales a partir de señales observables, tales como expresiones faciales, voz, postura o variables fisiológicas. Entre estas modalidades, el rostro ha sido una de las más estudiadas debido a su disponibilidad práctica, su bajo costo de captura y su relevancia en la comunicación no verbal [30, 38].

No obstante, la inferencia emocional automática presenta una dificultad importante: la emoción interna de una persona no siempre coincide de manera directa con su expresión observable. Por ello, en términos rigurosos, muchos sistemas no reconocen una "emoción interna" sino patrones faciales asociados a categorías emocionales definidas operacionalmente. Esta distinción es importante tanto desde el punto de vista científico como ético [13] por lo cual conoceremos a continuacion parte de los fundamentos psicologicos para el reconocimiento de emociones.

3.2. Fundamentos psicológicos del reconocimiento de emociones faciales

Una parte importante de la literatura en reconocimiento facial de emociones se apoya en la teoría de emociones básicas propuesta por Ekman, quien planteó que ciertas expresiones faciales poseen regularidades interculturales y pueden agruparse en categorías discretas las cuales son: felicidad, tristeza, enojo, miedo, asco, sorpresa y neutralidad [10].

Bajo esta perspectiva, el problema computacional puede formularse como una tarea de clasificación supervisada, donde una imagen o secuencia facial se asigna a una categoría emocional. Sin embargo, la literatura recien-

te ha mostrado que esta simplificación presenta límites importantes: algunas emociones son ambiguas, dependen del contexto, se expresan con distinta intensidad entre individuos y pueden solaparse visualmente entre sí [19, 38].

Estas dificultades son especialmente visibles en emociones como miedo, sorpresa o asco, cuya morfología facial puede compartir rasgos parciales o presentar alta variabilidad inter-sujeto. En consecuencia, el reconocimiento emocional facial no debe entenderse como un problema puramente visual, sino como un problema de representación, contexto y decisión bajo incertidumbre [19, 9].

A pesar de estas limitaciones teóricas, el modelo de Ekman sigue siendo un modelo recurrente en el desarrollo de sistemas computacionales. Esto se debe a su viabilidad operativa, ya que proporciona etiquetas claras y estructuradas para el entrenamiento de modelos de aprendizaje supervisado. De este modo, el desafío no radica en reemplazarlo, sino en diseñar herramientas algorítmicas capaces de capturar con alta precisión las sutiles variaciones geométricas y semánticas del rostro que diferencian a una expresión de otra. A continuación, se analiza cómo ha evolucionado la disciplina encargada de automatizar este proceso de captura y análisis de rasgos faciales.

3.3. Visión por computador aplicada al análisis facial

El reconocimiento de emociones faciales se apoya en técnicas de visión por computador orientadas a extraer información relevante desde imágenes o video. Estas técnicas pueden agruparse principalmente en tres grandes enfoques.

3.3.1. Métodos basados en características manuales

En los inicios del reconocimiento automático de emociones, la literatura se basó en descriptores diseñados de forma manual para procesar y codificar las variaciones en la apariencia del rostro. Herramientas como los patrones binarios locales (LBP) se encargaron de mapear las texturas superficiales de la piel analizando las relaciones de intensidad entre píxeles vecinos—lo que facilitaba la detección de arrugas sutiles y pliegues generados por la contracción muscular. En paralelo, técnicas como el histograma de gradientes orientados (HOG) permitieron capturar la geometría de los componentes del

rostro al registrar la orientación y magnitud de los bordes e intensidades en la imagen, delimitando zonas clave como la apertura de los ojos o la posición de los labios. Estos vectores de características eran posteriormente entregados a clasificadores tradicionales como las máquinas de soporte vectorial (SVM) para categorizar la expresión. Si bien este paradigma destaca por requerir un costo de procesamiento bajo y ofrecer un comportamiento matemático predecible, carece de la flexibilidad necesaria para generalizar en entornos no controlados. Al ser reglas estáticas, factores como las sombras portadas, las rotaciones de la cabeza o la variabilidad anatómica de los usuarios degradan rápidamente su precisión.[16]

3.3.2. Métodos basados en landmarks y geometría facial

Una segunda familia de enfoques en reconocimiento de emociones se basa en la detección de puntos clave del rostro (*facial landmarks*), a partir de los cuales se modelan deformaciones geométricas en regiones como cejas, ojos, nariz y boca.

En términos generales, estos enfoques se pueden clasificar en tres categorías principales:

- **Modelos basados en distancias geométricas:** Utilizan las relaciones espaciales cuantitativas entre los puntos clave del rostro. Estos algoritmos extraen vectores de características calculando distancias euclidianas, ángulos de apertura y proporciones relativas en regiones móviles (como la distancia interocular, el arqueamiento de las cejas o la apertura de la boca), permitiendo mapear la magnitud de la deformación facial de forma estática [29].

- **Modelos basados en Action Units (AUs):** Inspirados directamente en el Sistema de Codificación de Acción Facial (FACS) desarrollado por Ekman y Friesen, estos enfoques traducen las configuraciones de los puntos clave en activaciones de músculos o grupos musculares específicos del rostro (por ejemplo, el levantamiento de la ceja interna o el fruncimiento de los labios). Esto permite dotar al sistema de una capa de interpretación anatómica y semántica estandarizada para describir cualquier expresión [2].

- **Modelos basados en trayectorias temporales:** Extienden el análisis geométrico al dominio del video, registrando y modelando la evo-

lución secuencial de las coordenadas de los *landmarks* a lo largo del tiempo. Al analizar la dinámica y velocidad del movimiento, estos sistemas capturan fases críticas de la expresión como el inicio (*onset*), el punto máximo (*apex*) y la relajación (*offset*), lo que ayuda a diferenciar gestos espontáneos de los ensayados [39].

Uno de los sistemas más utilizados para la detección de landmarks es el propuesto por Kazemi y Sullivan, implementado en la librería Dlib [17].

3.3.3. Métodos basados en aprendizaje profundo

La tercera línea corresponde a los sistemas basados en aprendizaje profundo, donde las representaciones se aprenden directamente desde los datos. Esta transición permitió abandonar la dependencia de descriptores manuales y capturar patrones más complejos, robustos a pose, iluminación, oclusión y variabilidad individual [19, 38]. El mecanismo subyacente de este avance se basa en el aprendizaje jerárquico de representaciones. Las arquitecturas profundas están compuestas por múltiples capas no lineales; las primeras capas actúan a bajo nivel detectando caracteristicas visuales como gradientes o texturas, mientras que las capas intermedias y profundas combinan de manera sucesiva las caracteristicas antes mencionadas para construir conceptos semánticos más complejos, como la disposición tridimensional de los rasgos faciales. Al optimizar los parámetros del modelo en función de una pérdida global, la máquina aprende exactamente qué variaciones espaciales en el rostro corresponden a cada estado afectivo.

3.3.3.1. Redes neuronales convolucionales en reconocimiento facial

Durante varios años, las redes neuronales convolucionales (CNN) fueron el paradigma dominante en reconocimiento facial y FER. Arquitecturas como VGG, ResNet y EfficientNet demostraron un alto rendimiento al capturar patrones locales en la imagen, tales como bordes, texturas, contornos y configuraciones faciales [15, 36], sustentando su principal fortaleza en la extracción jerárquica de características, donde las capas tempranas identifican rasgos simples y las capas profundas integran patrones más abstractos. Sin embargo, estas arquitecturas presentan limitaciones estructurales relevantes para el reconocimiento emocional, debido a que operan inicialmente sobre regiones

locales y requieren de una gran profundidad para lograr integrar información distribuida a gran escala. Asimismo, su desempeño depende fuertemente de la disponibilidad masiva de datos etiquetados y, por la naturaleza de sus operaciones de convolución y agrupamiento (pooling), suelen presentar serias dificultades para modelar relaciones semánticas complejas entre regiones distantes del rostro, como la coordinación simultánea entre el arqueamiento de las cejas y la apertura de la boca. Estas restricciones metodológicas son las que finalmente motivaron la exploración de nuevas arquitecturas capaces de modelar de forma nativa las relaciones espaciales globales y de generalizar con mayor robustez ante escenarios fuera de la distribución de entrenamiento [19, 38].

3.4. Transformers en visión

La arquitectura de red neuronal Transformers, inicialmente desarrollada para el procesamiento de lenguaje natural, introdujo un cambio de paradigma mediante el mecanismo de auto-atención (Self-Attention). A diferencia de las operaciones de convolución de las CNNs, que están limitadas por el tamaño de un filtro local como se mencionó anteriormente, la auto-atención permite modelar relaciones dinámicas entre todos los elementos de una entrada de manera simultánea, sin importar la distancia matemática o espacial que los separe [40].

Para lograr esto, el mecanismo proyecta cada elemento de la secuencia en tres vectores distintos mediante matrices de pesos entrenables: una consulta (Query), una clave (Key) y un valor (Value). El sistema calcula la relevancia o el "peso de atención" entre un elemento y el resto de la secuencia midiendo la similitud matemática (generalmente mediante un producto punto a escala) entre la consulta del elemento en cuestión y las claves de todos los demás componentes. El resultado de esta operación genera una matriz de afinidad que se normaliza con una función softmax, determinando qué porcentaje de atención debe prestar el modelo a cada región de la entrada. Finalmente, este peso pondera los vectores de valor correspondientes, permitiendo que la salida de una sola capa contenga información contextual rica y distribuida globalmente.

Por otro lado, la arquitectura de *Vision Transformer* (ViT) adapta esta idea al dominio visual, dividiendo la imagen en parches (*patches*) y tratándolos como una secuencia. De esta forma, el modelo puede analizar la imagen

de manera global, considerando cómo distintas regiones se relacionan entre sí, incluso si están alejadas espacialmente [7].

Las principales ventajas de los Transformers en el área de visión (imágenes) incluyen:

- capacidad de analizar la imagen como un todo, considerando relaciones entre regiones distantes.
- mayor flexibilidad para aprender representaciones complejas.
- facilidad para integrarse con texto u otras modalidades.
- mejor escalabilidad hacia tareas multimodales.

En reconocimiento emocional, esta propiedad resulta particularmente valiosa, ya que muchas expresiones no dependen de una sola región facial, sino de la combinación entre ojos, cejas, nariz y boca. Aun así, los Vision Transformers también presentan desafíos en dispositivos edge, debido a sus mayores requerimientos de memoria y cómputo frente a CNN livianas [24].

3.5. Modelos multimodales y CLIP

CLIP (*Contrastive Language-Image Pretraining*) es un modelo multimodal que aprende una representación compartida entre imágenes y texto mediante aprendizaje contrastivo [31] , un modelo fundamentado en los transformers. El aprendizaje contrastivo es una técnica de entrenamiento que busca acercar en el espacio de representación aquellas muestras que están relacionadas (pares positivos) y alejar aquellas que no lo están (pares negativos). En el caso de CLIP, un par positivo corresponde a una imagen y su descripción textual asociada, mientras que los pares negativos corresponden a combinaciones incorrectas entre imágenes y textos dentro del mismo batch.

Su entrenamiento busca maximizar la similitud entre pares imagen-texto correctos y minimizarla para pares incorrectos, proyectando ambas modalidades en un mismo espacio semántico.

Formalmente, CLIP puede entenderse como una función de representación conjunta:

$$f : (I, T) \rightarrow \mathbb{R}^d \tag{3.1}$$

donde I corresponde a una imagen, T a una descripción textual y d a la dimensión del espacio latente compartido.

En tareas de clasificación, esta arquitectura permite comparar una imagen con distintas descripciones textuales (*prompts*) en lugar de depender exclusivamente de una capa densa entrenada para clases fijas. Esta propiedad es especialmente relevante en contextos donde interesa aprovechar conocimiento semántico previo, generalización fuera de distribución o flexibilidad de etiquetas [31].

En reconocimiento emocional, CLIP introduce una ventaja conceptual importante: no solo analiza patrones visuales, sino que relaciona esos patrones con una semántica textual. Sin embargo, esta misma propiedad puede generar ambigüedades cuando las clases emocionales son visualmente cercanas o semánticamente solapadas. La literatura reciente en FER basada en CLIP ha respondido a este problema enriqueciendo los *prompts*, incorporando descripciones más precisas y agregando componentes temporales o multimodales [44, 3, 23].

3.6. Transfer learning y fine-tuning

El *transfer learning* consiste en una técnica de inteligencia artificial basada en reutilizar un modelo previamente entrenado en una tarea general para adaptarlo a un dominio más específico. Esta estrategia es importante en visión por computador, donde entrenar modelos de gran escala desde cero suele ser costoso y requerir grandes volúmenes de datos [19].

Dentro de este marco, el *fine-tuning* corresponde al ajuste parcial o total de los pesos del modelo preentrenado. En la práctica, existen tres estrategias frecuentes:

- **Congelamiento total:** solo se entrena la capa de salida.
- **Fine-tuning parcial:** se ajustan capas superiores o bloques específicos.
- **Fine-tuning completo:** se optimiza toda la arquitectura.

En tareas especializadas como FER, el *fine-tuning* parcial suele ofrecer un equilibrio adecuado entre adaptación y estabilidad, permitiendo que el modelo incorpore patrones específicos del dominio sin destruir su capacidad de generalización [19, 38].

3.7. Datasets de reconocimiento emocional facial

El desarrollo de modelos FER depende fuertemente de la disponibilidad de datasets etiquetados. Entre los conjuntos más relevantes se encuentran FER2013, RAF-DB y AffectNet, los cuales son considerados como referencias estándar para entrenamiento y evaluación [14, 25, 21].

- **RAF-DB:** 41000 imágenes que corresponden dataset realista con imágenes faciales etiquetadas en condiciones no controladas.
- **AffectNet:** 30000 imágenes correspondiente a uno de los mayores conjuntos *in-the-wild*, con alta variabilidad de pose, iluminación, oclusión y espontaneidad expresiva.
- **FER2013:** 35900 imágenes de base clásica con imágenes en escala de grises, ampliamente utilizada como benchmark introductorio.

Estos datasets presentan problemas conocidos, como desbalance entre clases, ruido en las etiquetas, sesgos demográficos y variabilidad en condiciones de captura. Tales factores influyen directamente en la capacidad de generalización del modelo y ayudan a explicar por qué los resultados obtenidos en benchmarks no siempre se replican en escenarios reales con usuarios [19, 9].

3.8. Estado del arte en reconocimiento emocional facial

El estado del arte reciente muestra una evolución desde modelos CNN centrados en imágenes estáticas hacia enfoques más robustos en condiciones *in-the-wild*, incluyendo arquitecturas multimodales, temporales y adaptadas a despliegue en tiempo real [19, 38, 9].

A grandes rasgos, los avances actuales pueden agruparse en cuatro direcciones:

- modelos más robustos a variaciones de iluminación, pose y oclusión;
- integración de información temporal para secuencias de video;

- incorporación de señales multimodales o semánticas;
- optimización para ejecución en edge y sistemas embebidos.

En particular, la evidencia reciente muestra que la principal brecha entre benchmark y despliegue en la vida real no está únicamente en la arquitectura del clasificador, sino en la combinación entre robustez visual, latencia, calibración, estabilidad temporal y factibilidad de uso con personas reales [33, 24, 4, 20].

3.9. Hardware y restricciones de despliegue

La computación en el borde (*edge computing*) es un área de la computación que se centra en ejecutar procesamiento cerca de la fuente de datos, reduciendo latencia, dependencia de conectividad y transferencia de información sensible. Esta estrategia es especialmente valiosa en sistemas de visión y robótica, donde la respuesta en tiempo real y la privacidad son factores críticos [4, 1].

Dispositivos como NVIDIA Jetson Xavier NX permiten desplegar modelos de deep learning directamente sobre hardware embebido. Sin embargo, estos sistemas están sujetos a restricciones concretas que condicionan su viabilidad operativa, comenzando por una memoria RAM limitada que restringe el tamaño máximo de los modelos y un presupuesto energético acotado que exige un uso eficiente de los ciclos de cómputo. A estas limitaciones se suma una capacidad térmica restringida, la cual puede provocar caídas de rendimiento por sobrecalentamiento y una constante competencia por recursos de hardware entre la CPU, la GPU y los subsistemas de entrada y salida al capturar y procesar video en tiempo real. También es importante evaluar factores como el tiempo de inferencia, la tasa de cuadros por segundo, el uso de memoria y la complejidad de integración del sistema en un entorno de producción real [24, 4].

Desde el punto de vista del despliegue, estas restricciones implican desafíos adicionales. La implementación de modelos en dispositivos embebidos requiere adaptar tanto la arquitectura como el flujo de procesamiento para lograr estabilidad, eficiencia y capacidad de respuesta en tiempo real. Esto incluye la optimización del pipeline de inferencia, la gestión de memoria, la reducción de latencia y la coordinación entre componentes de hardware.

Asimismo, el despliegue en escenarios reales introduce factores adicionales como variabilidad en las condiciones de captura (iluminación, movimiento, oclusiones), disponibilidad limitada de recursos y la necesidad de operación continua sin degradación del rendimiento. En este contexto, el diseño del sistema debe considerar desempeño del modelo y la integración efectiva en un entorno real.

3.10. Arquitecturas híbridas para reconocimiento emocional en tiempo real

Una estrategia frecuente para compatibilizar precisión y eficiencia en edge consiste en diseñar arquitecturas híbridas, donde tareas ligeras de monitoreo se ejecutan en CPU y la inferencia pesada se reserva para GPU únicamente cuando es necesario.

Este principio de activación selectiva de la inferencia se observa en múltiples pipelines modernos de video en tiempo real: primero se detecta o rastrea una región de interés; luego, sólo bajo determinadas condiciones, se activa el modelo profundo. En términos de ingeniería, este enfoque reduce carga innecesaria, mejora la latencia percibida y permite explotar mejor la heterogeneidad del hardware [45, 28, 27].

En el presente trabajo, esta idea se materializa mediante un disparo geométrico basado en landmarks, que actúa como compuerta de activación para la inferencia emocional. Desde el punto de vista teórico, esta arquitectura articula tres niveles: detección estructural del rostro, modelado semántico de la emoción y gestión eficiente del cómputo embebido.

3.11. Métricas y criterios de evaluación

La evaluación de sistemas de reconocimiento de emociones faciales (FER) requiere el uso de métricas y criterios estandarizados que permitan cuantificar de manera objetiva el rendimiento del modelo. Las métricas de evaluación son descriptores matemáticos y estadísticos diseñados para medir la discrepancia entre las predicciones emitidas por un algoritmo y las etiquetas reales o de referencia provistas por los conjuntos de datos. En problemas multiclase y potencialmente desbalanceados, accuracy global es entendida como la proporción de predicciones correctas sobre el total de muestras, esto suele

suele ser insuficiente, ya que puede ocultar fallas severas en las clases minoritarias debido a la influencia de las categorías sobrerepresentadas. Por ello, la literatura científica exige complementar el análisis con métricas de precisión (precision), que mide la proporción de verdaderos positivos entre todas las predicciones asignadas a una clase, y exhaustividad (recall), que cuantifica la capacidad del modelo para identificar la totalidad de los ejemplos pertenecientes a dicha categoría. La media de ambas métricas da origen al F1-score, un indicador robusto que equilibra la especificidad y la sensibilidad del clasificador bajo escenarios de desbalance de datos. Asimismo, la matriz de confusión actúa como una herramienta fundamental de diagnóstico visual, mapeando las tasas de acierto y los patrones de error específicos entre las distintas clases emocionales [19, 38].

En el plano de los sistemas desplegados en tiempo real, la evaluación trasciende el comportamiento puramente estadístico para incorporar métricas de rendimiento en ingeniería de software y hardware. Entre estos indicadores destaca la latencia de respuesta, medida comúnmente en milisegundos, que define el tiempo total requerido por el pipeline desde la captura del cuadro de video hasta el despliegue de la inferencia, impactando directamente en la tasa de cuadros por segundo (FPS) realizables por el dispositivo de borde. En estrecha relación con la fluidez del sistema se evalúa el perfil de consumo energético y el uso porcentual de la CPU y la GPU de la plataforma embebida, factores críticos que determinan la viabilidad operativa y la estabilidad térmica del hardware en ejecuciones continuas. Finalmente, la consistencia o estabilidad temporal de la salida analiza si el clasificador mantiene predicciones coherentes a lo largo de cuadros sucesivos en una misma secuencia de video, mitigando el parpadeo de etiquetas causado por el ruido ambiental o las fluctuaciones menores en la pose del usuario.

Cuando el sistema interactúa con usuarios humanos en entornos operativos reales, el diseño metodológico de la evaluación debe complementarse con métricas centradas en la interacción humano-computador. Estos criterios miden el impacto del sistema en escenarios prácticos mediante variables como el tiempo de reacción del software frente a una expresión espontánea, el número de intentos requeridos para registrar un cambio afectivo válido y la consistencia de reconocimiento bajo condiciones de iluminación y movimiento no controladas. Al integrar la precisión del modelo adaptado, la eficiencia del hardware periférico y la experiencia de uso con personas reales, se construye un marco de evaluación integral que valida de manera empírica la viabilidad del sistema más allá del análisis tradicional de laboratorio [33].

3.12. Limitaciones científicas, éticas y de ingeniería

El reconocimiento emocional automático enfrenta limitaciones que no son solo técnicas, sino también científicas y éticas.

Desde lo científico, existe debate sobre hasta qué punto la emoción puede inferirse de manera confiable a partir del rostro aislado, sin información contextual o temporal. La literatura reciente en revisiones sistemáticas ha enfatizado la necesidad de distinguir entre expresión facial observable y estado emocional interno, así como de reconocer la influencia de cultura, contexto, identidad y situación [13].

Desde la ética y la ingeniería, emergen además consideraciones sobre:

- privacidad y tratamiento de datos faciales;
- riesgo de sesgo demográfico o contextual;
- interpretaciones erróneas de la emoción;
- uso responsable de sistemas que podrían influir en decisiones sobre personas.

Por ello, un diseño de ingeniería robusto no solo debe optimizar desempeño, sino también transparentar limitaciones, definir ámbitos de uso apropiados y considerar las implicancias sociales del sistema.

3.13. Resumen del marco teórico

En síntesis, el reconocimiento de emociones faciales es un problema multidisciplinario que articula fundamentos de computación afectiva, visión por computador, aprendizaje profundo y despliegue de sistemas embebidos.

La evolución desde métodos basados en descriptores manuales hacia CNN, Transformers y modelos multimodales ha ampliado significativamente la capacidad de representación y generalización. Sin embargo, persisten desafíos asociados a la ambigüedad emocional, la dependencia del contexto, la dinámica temporal y las restricciones de hardware cuando se busca operar en tiempo real.

Capítulo 4

Metodología y diseño de la solución

Este capítulo detalla el proceso metodológico que conecta la teoría del proyecto con su implementación física. Inicia con la visión general de la arquitectura. Con esa base, se profundiza en la preparación de los datos y las estrategias de entrenamiento, pasos críticos para asegurar la robustez del modelo. El capítulo concluye con el diseño de ingeniería de software para el hardware de borde, donde se integran los mecanismos de disparo geométrico que posibilita la operación del sistema bajo restricciones de tiempo real.

4.1. Visión general de la arquitectura propuesta

Para abordar la problemática del reconocimiento de emociones espontáneas en tiempo real sobre el hardware borde, esta memoria propone una arquitectura híbrida en dos etapas basada en la **inferencia disparada por eventos** (*Event-Triggered Inference*). A diferencia de los enfoques convencionales que procesan cada fotograma de forma continua mediante una Red Neuronal Profunda (DNN) lo cual satura los recursos de el hardware de borde, el sistema propuesto adopta un diseño en cascada que desacopla la detección del movimiento facial del análisis semántico de la emoción [41, 42].

El diseño se estructura en dos fases operativas distintas bajo un enfoque de lo general a lo específico (*coarse-to-fine*) [6]:

1. **Fase de detección ligera (CPU):** Un algoritmo basado en regresión de árboles (Dlib) monitorea continuamente la geometría facial a un costo computacional marginal.

2. **Fase de clasificación profunda (GPU):** Solo cuando el preprocesamiento geométrico detecta una distorsión elástica significativa en el rostro (el evento desencadenante), se activa el modelo CLIP ViT-B/32 para inferir la categoría emocional.

Esta estrategia de mitigación de carga permite mantener una tasa de cuadros (FPS) fluida para la interacción con el usuario, reservando los *Tensor Cores* de la NVIDIA Jetson Xavier NX exclusivamente para los fotogramas de alta relevancia semántica. Para que esta fase de clasificación profunda opere con la precisión requerida en el mundo real, el modelo no puede depender de su entrenamiento original genérico, sino que debe especializarse en el dominio facial. Por ello, el siguiente paso consistió en la recolección y preparación de los datos para este proceso.

4.2. Ingeniería de datos y preprocesamiento

La capacidad de generalización de un modelo *Transformer* depende de la diversidad de los datos de entrenamiento. Para superar el sesgo de los datasets de laboratorio, se construyó uno consolidado.

4.2.1. Consolidación del DataSet

Se unificaron tres conjuntos de datos estándar para cubrir un espectro amplio de variabilidad:

- **RAF-DB (Real-world Affective Faces Database):** Aporta 29,672 imágenes con expresiones *in-the-wild* y anotaciones de alta confiabilidad mediante *crowdsourcing*. Es la fuente principal para aprender micro-expresiones genuinas [22].

- **AffectNet :** Se extrajo 20576 imagenes de este dataset masivo para introducir variaciones de iluminación útiles para la robustez en entornos no controlados.

- **FER2013:** Aportó con 25832 imágenes. A pesar de su baja resolución (48×48), se incluyó para forzar al modelo a aprender características estructurales gruesas.

4.2.2. Estrategia de ponderación de clases

El análisis exploratorio de los datos reveló un desbalance severo: la clase "Felicidad" supera en mas del doble a clases críticas como "Miedo" o "Asco". Entrenar bajo estas condiciones provocaría que el modelo colapse hacia la clase mayoritaria. En la imagen a continuación se puede ver claramente lo expresado

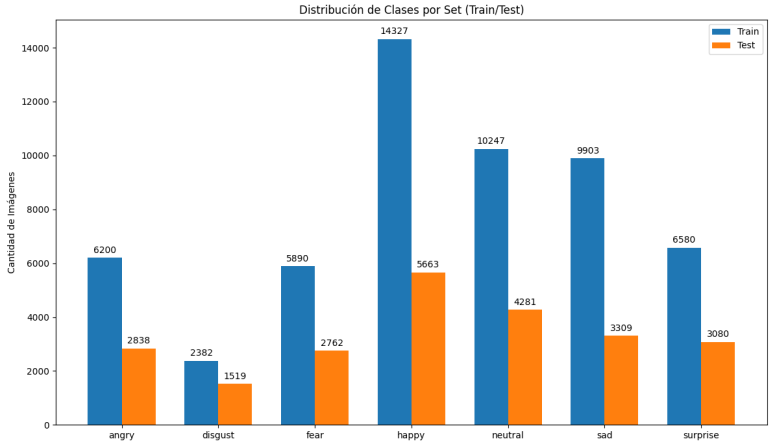


Figura 4.1: Distribución de imágenes por clase en el dataset utilizado para entrenamiento. Se observa un fuerte desbalance hacia la clase *Felicidad*.

Para mitigar este problema, se implementó una función de pérdida de Entropía Cruzada Ponderada (*Weighted Cross-Entropy Loss*). Los pesos w_c para cada clase c se calcularon inversamente a su frecuencia, penalizando más severamente los errores en las emociones minoritarias:

$$w_c = \frac{N_{total}}{C \times N_c} \tag{4.1}$$

Donde $C = 7$ es el número de emociones básicas de Ekman.

Con el Data-Set consolidado y las asimetrías de clase mas balanceadas que al inicio, la siguiente etapa corresponde a la asimilación de características, diseñar la estrategia mediante la cual el modelo CLIP integraría este nuevo conocimiento sin perder su robustez original.

4.3. Estrategia de entrenamiento: Deep Fine-Tuning

El entrenamiento del modelo CLIP se diseñó bajo una estrategia de transferencia de aprendizaje progresiva diseñada para adaptar la representación semántica generalista de CLIP a la tarea específica de FER sin perder su conocimiento previo.

4.3.1. Descongelamiento selectivo de capas

El codificador de imagen de CLIP (ViT-B/32) consta de 12 capas de transformadores (Bloques de Atención). En lugar de un ajuste fino total (que requiere demasiados datos y riesgo de *overfitting*) o un entrenamiento solo del clasificador (que limita la capacidad de aprendizaje), se optó por un enfoque parcial:

- **Capas 0-5 (Congeladas):** Estas capas capturan primitivas visuales básicas (bordes, texturas simples) que son universales. Se mantienen congeladas para reducir el consumo de VRAM y preservar la estabilidad.
- **Capas 6-11 (Descongeladas):** Las capas superiores, responsables de la abstracción semántica y las relaciones globales (ej. la configuración de ojos vs boca), se liberan para el entrenamiento. Esto permite que el mecanismo de auto-atención se reespecialice en las sutilezas de la musculatura facial humana [35].

4.3.2. Hiperparámetros y regularización

El proceso de *Fine-Tuning* se ejecutó con una configuración estricta orientada a preservar el conocimiento preentrenado del modelo y mejorar su capacidad de generalización:

- **Optimizador:** Se utilizó AdamW con un *Weight Decay* de 0,1. Esta variante del optimizador Adam desacopla la regularización del proceso de actualización de los pesos, lo que permite controlar de manera más efectiva el sobreajuste y evitar que las capas profundas memoricen ruido presente en datasets de menor calidad, como FER2013.

- **Learning Rate Diferencial:** Se empleó una tasa de aprendizaje baja (5×10^{-6}) en las capas del transformador preentrenado. Este enfoque permite realizar ajustes graduales sobre representaciones ya aprendidas, evitando modificaciones abruptas que puedan degradar el conocimiento previo del modelo.
- **Label Smoothing (0,15):** Se aplicó suavizado de etiquetas, reemplazando las etiquetas categóricas duras por distribuciones de probabilidad suavizadas. Esto reduce la confianza excesiva del modelo en una única clase y permite capturar la ambigüedad inherente entre ciertas emociones, como "Miedo" y "Sorpresa", mejorando así la calibración de las predicciones [26].

Una vez consolidada la etapa de entrenamiento del clasificador, se procedió al diseño del sistema de inferencia para el hardware. Esta fase se enfoca en resolver de manera práctica los requerimientos de latencia y gestión de recursos mediante una arquitectura de software especifica para la plataforma de borde.

4.4. Diseño del sistema de inferencia en el dispositivo

La implementación en la NVIDIA Jetson Xavier NX requirió una serie de ajustes para obtener un procesamiento estándar para superar las limitaciones de latencia inherentes al hardware.

4.4.1. Motor de preprocesamiento híbrido

Se diseñó un pipeline que combina dos librerías de visión artificial con roles especializados:

- **Dlib (C++ optimizado):** Ejecuta el algoritmo de alineación facial de Kazemi y Sullivan [17] para extraer 68 puntos de referencia (*landmarks*). Estos puntos se utilizan para calcular la geometría del rostro y normalizar la rotación (alineación de ojos) antes de pasar la imagen a la red neuronal. Como se observa en la Figura 4.2, este proceso permite visualizar tanto la detección de landmarks como las variaciones geométricas utilizadas para el disparo del modelo.

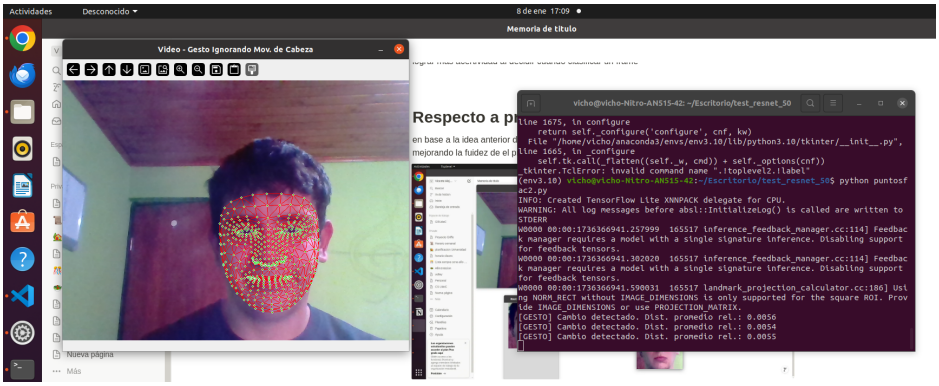


Figura 4.2: Visualización durante la fase de preprocesamiento y registro en consola de las variaciones geométricas (*Geometric Event Triggering*).

- **MediaPipe (Segmentación):** Se utiliza para la sustracción de fondo (*Selfie Segmentation*). Eliminar el contexto visual irrelevante ayuda al Vision Transformer a focalizar su atención exclusivamente en la topología facial, reduciendo alucinaciones causadas por objetos del entorno.

4.4.2. Algoritmo de disparo por elasticidad facial

Para optimizar el uso de la GPU, se implementó un algoritmo de disparo basado en la "Energía de Distorsión" del rostro. En este contexto, un algoritmo de disparo se define como un mecanismo que activa un proceso únicamente cuando se cumple una condición específica, evitando ejecuciones continuas innecesarias.

En este trabajo, dicho mecanismo actúa como una compuerta de activación: la inferencia del modelo profundo solo se ejecuta cuando la variación geométrica facial supera un umbral predefinido, evitando así evaluaciones redundantes en estados de reposo.

El sistema calcula continuamente la sumatoria de las distancias físicas entre puntos faciales de alta movilidad (como las cejas y las comisuras de los labios) respecto a un punto de anclaje anatómico rígido (el puente nasal), lo que se ilustra en la Figura 4.2. Esta métrica se formaliza en la ecuación 4.2:

$$E = \frac{\sum_{i=1}^{n} \sqrt{(x_i - x_{ancla})^2 + (y_i - y_{ancla})^2}}{D_{io}} \tag{4.2}$$

donde D_{io} corresponde a la distancia interpupilar, utilizada como factor de

normalización para hacer la medida invariante a escala. Además n es el número de puntos móviles analizados y D_{io} representa la distancia interocular. Para garantizar que esta métrica sea invariante a la escala, es decir, que funcione correctamente independientemente de la distancia del usuario a la cámara, el valor de esta deformación se normaliza dividiéndolo por la distancia interocular. El sistema mantiene un historial temporal de esta métrica y la inferencia del modelo profundo solo se desencadena si la variación geométrica dentro de una ventana de tiempo de 5 a 10 fotogramas (aproximadamente 160-330 ms) supera un umbral de tolerancia del 5 % de deformación relativa.

Este mecanismo opera efectivamente como un filtro de paso bajo que descarta el ruido visual y micro-movimientos involuntarios, reservando el procesamiento intensivo de la GPU para cambios expresivos genuinos.

4.4.3. Arquitectura del software

El software se estructura utilizando el patrón *Producer-Consumer*, este tipo de arquitectura es común en pipelines de procesamiento de video en tiempo real basados en inteligencia artificial, donde es necesario balancear latencia, throughput y uso de recursos computacionales [45]. El hilo principal (Main Thread) maneja la captura de cámara y el renderizado a 30 FPS. Un hilo de trabajo o hebra (*Worker Thread*) aloja el modelo CLIP en la GPU. La comunicación se realiza mediante una cola (*Queue*) de tamaño 1 tipo LIFO (*Last In, First Out*), garantizando que, si el modelo se retrasa, siempre procese el fotograma más reciente disponible, descartando los obsoletos para minimizar la latencia percibida por el usuario.

Si bien el diseño de inferencia descrito garantiza la ejecución fluida, el estado final del modelo en la Jetson Xavier NX es el resultado de una evolución de una serie de pruebas y errores. El proceso de entrenamiento no fue estático, sino que atravesó distintas fases para superar la fase inicial, lo cual se detalla a continuación.

4.5. Progresión del Fine-Tuning y convergencia

La configuración de los hiperparámetros no se ejecutó de manera lineal y estática, sino que atravesó una secuencia evolutiva programada para estabilizar la dinámica de convergencia del modelo. Esta progresión se encuentra documentada operativamente a través de las iteraciones de código `train_v3.py`,

`train_v4.py` y `train_v5.py`, hasta alcanzar la compilación conclusiva en `train_v7.py` [1].

Durante la etapa inicial, la calibración forzó un congelamiento matricial sobre el bloque inferior de la red (capas de auto-atención de la 0 a la 5). Esta restricción permitió transferir recursos hacia el descongelamiento superior del modelo (capas de la 6 a la 11). Bajo esta parametrización, la trayectoria de descenso hacia el mínimo global se garantizó estipulando una velocidad de aprendizaje de 5×10^{-6}, con un decaimiento residual (*Weight Decay*) de valor 0,1 mitigando así, el sobreajuste frente al ruido intrínseco del conjunto de datos facial.

Por último, el modelo fue sometido a un descongelamiento completo (*Full Fine-Tuning*), activando la actualización de gradientes en la totalidad de sus parámetros. Para asegurar la estabilidad de la convergencia y mitigar las oscilaciones (patrón de "sierra") observadas en las curvas de pérdida previas, se redujo drásticamente la tasa de aprendizaje a un valor ultraconservador de 5×10^{-7}. Dadas las restricciones de memoria del hardware (*Jetson Xavier NX*), se utilizó un tamaño de lote unitario (`batch_size = 1`). Para compensar esta limitación, se aplicó acumulación de gradientes durante 64 iteraciones, actualizando los pesos del modelo solo después de procesar múltiples muestras. Esto permite aproximar el comportamiento de entrenamiento de un lote mayor (equivalente a 64 muestras), manteniendo estabilidad en el proceso de optimización sin exceder los recursos disponibles. Finalmente, la estabilización del modelo frente a las técnicas de aumentación de datos agresivas (como *Random Erasing* y *Color Jittering*) se reforzó mediante la aplicación de *Label Smoothing* con un factor de 0,20, mejorando la calibración de las predicciones.

[1] Repositorio con las iteraciones del código de entrenamiento disponible en `https://github.com/VicenteRodriguezAlfaro/MT.git`

Capítulo 5

Detalles de implementación y optimización

Para validar la viabilidad práctica de la solución propuesta, este capítulo presenta el proceso de desarrollo y calibración del prototipo embebido. El texto se organiza describiendo en primera instancia el entorno de hardware y las configuraciones de energía del hardware de borde Jetson Xavier NX. Posteriormente, se presenta la implementación adaptada para el entrenamiento y las transformaciones aplicadas a los datos. Finalmente, se profundiza en las optimizaciones de memoria y el diseño del pipeline de software, elementos críticos para resolver las limitaciones de latencia en la captura e inferencia en tiempo real.

5.1. Plataforma de hardware y entorno

Como se mencionó anteriormente se desplegó sobre un kit de desarrollo **NVIDIA Jetson Xavier NX** (8GB RAM), un sistema en módulo (SoM) diseñado para la computación acelerada. A diferencia de las GPUs de escritorio, este dispositivo opera bajo restricciones estrictas de energía y temperatura.

5.1.1. Especificaciones técnicas relevantes

Para maximizar el rendimiento del modelo CLIP, se aprovecharon características específicas de la GPU integrada:

- **Tensor Cores:** La Jetson Xavier NX dispone de 48 Tensor Cores.

Estos núcleos están especializados en operaciones de multiplicación y acumulación de matrices (MMA) en precisión mixta, fundamentales para acelerar los bloques de atención del *Vision Transformer* [5].

- **Modo de Energía (Power Mode):** Se configuró el dispositivo en el modo de máximo rendimiento (15W 6-core), fijando las frecuencias de reloj de la CPU y la GPU al máximo mediante el comando `jetson_clocks`. Esto elimina el latencia variable introducida por el escalado dinámico de frecuencia (DVFS).

5.2. Implementación del entrenamiento (Deep Fine-Tuning)

El entrenamiento del modelo se comenzó utilizando el script desarrollado `train_v4.py`. La implementación se basó en extender las clases base para adaptar el proceso a las necesidades del dataset desbalanceado.

5.2.1. Personalización del *Trainer* (clase *WeightedTrainer*)

Dado que el conjunto de datos combinado presenta un desequilibrio de clases significativo, Se implementó una clase personalizada `WeightedTrainer` que sobrescribe el método de cálculo de pérdida:

```
class WeightedTrainer(Trainer):
    def compute_loss(self, model, inputs, return_outputs=False):
        labels = inputs.get("labels")
        outputs = model(**inputs)
        logits = outputs.get("logits")
        # Inyección de pesos calculados previamente para balancear clases
        loss_fct = CrossEntropyLoss(weight=class_weights_tensor)
        loss = loss_fct(logits.view(-1, num_labels), labels.view(-1))
        return (loss, outputs) if return_outputs else loss
```

Esta modificación asegura que los gradientes generados por errores en clases minoritarias (como "Miedo") tengan una magnitud comparable a los de las clases mayoritarias, estabilizando la convergencia.

5.2.2. Pipeline de aumentación de datos

Para combatir el sobreajuste al descongelar las capas profundas del codificador (capas 6-11), se implementó un pipeline de transformaciones utilizando `torchvision`. Se aplicaron rotaciones aleatorias ($\pm 20°$), cambios de iluminación (*Color Jitter*). Estas perturbaciones obligan al modelo a aprender características invariantes a la posición y la calidad de la imagen, simulando las imperfecciones de una cámara web real.

5.3. Ingeniería de software para inferencia en tiempo real

El software de despliegue en la Jetson Xavier NX se estructuró siguiendo el patrón de diseño *Productor-Consumidor* para separar la captura de video del procesamiento neuronal pesado. Esto se implementó mediante el módulo `threading` de Python y colas seguras para Hebras.

5.3.1. Captura y renderizado (hebra principal)

Se encapsuló el acceso a la cámara en la clase `WebcamStream`, la cual lee fotogramas en una hebra dedicada para evitar el bloqueo de E/S. la Hebra principal se encarga de:

1. Obtener el último fotograma disponible.
2. Ejecutar la detección facial ligera con Dlib (CPU).
3. Calcular la métrica de distorsión facial y gestionar el disparador de eventos.
4. Renderizar la interfaz gráfica (HUD) y mostrar el resultado a 30 FPS constantes, independientemente de la velocidad de la red neuronal.

5.3.2. Worker de inferencia

La clase `InferenceWorker` opera en segundo plano. Mantiene una cola de tamaño 1 (`maxsize=1`), lo que garantiza que siempre procese el dato más reciente y descarte los fotogramas antiguos si la GPU está saturada (política *Drop-Oldest*).

```
def run(self):
    while not self.stopped:
        try:
            frame = self.queue.get(timeout=0.1)
            emotion, score = self.ai.classify_emotion(frame)
            self.result = (emotion, score)
        except Empty:
            continue
```

Este diseño es crítico para la percepción de fluidez del usuario, ya que evita que el video se "congele" durante los 30-50ms que tarda la inferencia de CLIP.Para maximizar el rendimiento de este subproceso bajo las restricciones del hardware, se incorporaron estrategias de optimización a nivel de memoria y de ejecución en CUDA. En primer lugar, se utilizó de manera estricta el gestor de contexto `torch.no_grad()` durante la evaluación del modelo CLIP. Esto suprime la creación del grafo de computación dinámico de PyTorch, reduciendo drásticamente los ciclos redundantes de CPU y el consumo de memoria RAM en el dispositivo. En segundo lugar, se implementó una rutina de calentamiento (*warm-up*) durante la inicialización del sistema. Esta lógica ejecuta una inferencia asíncrona inicial empleando tensores vacíos antes de habilitar la interfaz de usuario, forzando la asignación previa de los buffers de memoria en la GPU y eliminando los retardos de latencia que típicamente afectan a la primera predicción en tiempo real.

Capítulo 6

Resultados

Este capítulo presenta la evaluación cuantitativa del sistema desplegado en hardware (NVIDIA Jetson Xavier NX), analizando su exactitud, latencia y robustez durante la interacción en tiempo real con usuarios humanos en un entorno no simulado.

6.1. Procedimiento experimental

El protocolo de evaluación se implementó y automatizó mediante el script `exp10.py`[1]. Para mitigar la latencia computacional y mantener la fluidez del sistema, la inferencia del modelo profundo se delegó a un hilo asíncrono. Este proceso utiliza una cola de capacidad unitaria con una política de descarte de datos obsoletos (*Drop-Oldest*), garantizando que la GPU siempre procese el fotograma más reciente detectado como movimiento suficiente para ser considerado una expresión de emocionalidad.

El acceso a la GPU se activa únicamente mediante un disparador geométrico evaluado en la CPU. Este evento se define por el cálculo de distancias sobre 68 puntos de referencia faciales (*landmarks*) extraídos con Dlib. Para evitar falsos positivos ante micro-movimientos involuntarios, el sistema exige una probabilidad de activación superior al umbral $\alpha = 0{,}5$, combinada con un margen diferencial de confianza respecto a la segunda clase más probable de $\Delta = 0{,}15$.

[1] Código fuente del script experimental disponible en `https://github.com/VicenteRodriguezAlfaro/MT.git`

6.2. Descripción y protocolo del experimento

La validación experimental de la arquitectura se llevó a cabo sobre una muestra demográficamente balanceada de $n = 32$ participantes (16 mujeres y 16 hombres), con edades comprendidas estrictamente entre los 18 y 35 años, obteniendo un total de 448 observaciones consolidadas.

El procedimiento interactivo se automatizó íntegramente mediante el entorno experimental implementado en el script `exp10.py`. En primera instancia, la aplicación proporciona una interfaz visual en tiempo real que permite visualizar simultáneamente el rostro capturado por la cámara, la imagen segmentada utilizada durante el preprocesamiento y los puntos de referencia faciales (*facial landmarks*) ajustados dinámicamente sobre la geometría del rostro. Adicionalmente, el sistema despliega barras de representación probabilística asociadas a las emociones inferidas por el modelo, permitiendo observar la interpretación semántica que la red neuronal realiza de la expresión facial del usuario en cada instante.

Sobre esta infraestructura interactiva se estructuró la evaluación experimental de cada participante, organizando el procedimiento en 14 iteraciones (*trials*) distribuidas en dos etapas metodológicas distintas:

- **Fase 1 (Expresión Inducida por Texto):** A través de la interfaz gráfica superpuesta (*HUD*), es decir, una capa visual que se renderiza directamente sobre la imagen de la cámara en tiempo real, el sistema instruye al participante a gesticular una emoción específica indicada únicamente mediante texto (ej. "Miedo"). El objetivo es evaluar la representación interna y la ejecución biomecánica libre del usuario, como se muestra en la Figura 6.1.

Figura 6.1: Interfaz experimental utilizada durante la Fase 1, donde la emoción objetivo es presentada únicamente mediante texto sobre el flujo de video en tiempo real.

- **Fase 2 (Imitación de estímulo visual):** El sistema muestra en pantalla una imagen de referencia aleatoria, extraída del conjunto de validación, que exhibe una emoción en condiciones reales. El participante debe interpretar visualmente el estímulo y replicar la configuración facial observada, como se ve en la Figura 6.2.

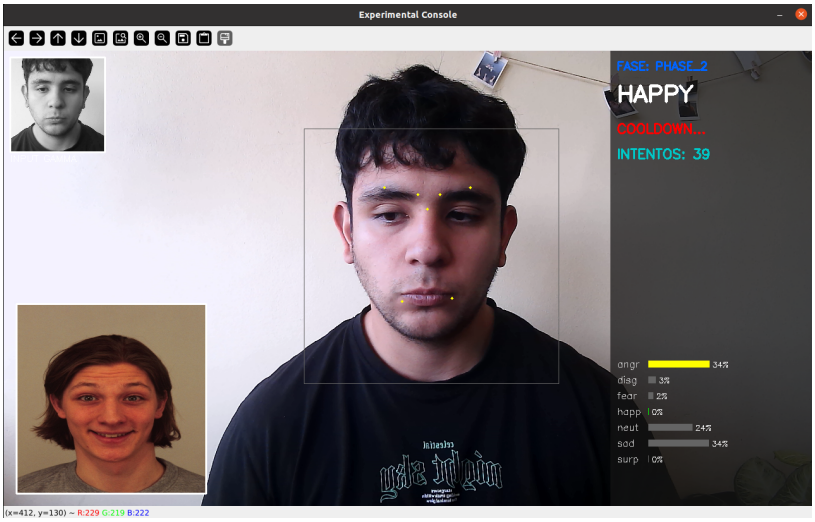


Figura 6.2: Interfaz experimental utilizada durante la Fase 2, donde el usuario debe imitar visualmente una expresión facial de referencia mostrada por el sistema.

Dinámica de interacción y validación: Durante cada *trial*, el participante se sitúa frente a la cámara en un estado de reposo inicial. El usuario debe ejecutar el gesto facial de manera evidente. El sistema no procesa el video de forma continua; en su lugar, espera a que el operador cruce el umbral de deformación elástica establecido (*Energy Distortion*). Al dispararse el evento geométrico, la GPU captura y procesa el fotograma.

Para asegurar la objetividad del experimento, el avance hacia la siguiente emoción está automatizado (*Auto Advance*). El sistema valida la expresión y avanza de forma autónoma únicamente si la red neuronal identifica la emoción objetivo correcta y, además, supera los criterios de seguridad matemática: una confianza absoluta predictiva mayor o igual a 50 % y un margen de certidumbre respecto a la segunda clase más probable mayor o igual a 15 %. En escenarios de estancamiento temporal por falta de reconocimiento, el operador puede efectuar un salto manual (vía teclado) para registrar el ensayo y continuar con el protocolo.

A lo largo de este ciclo continuo, el algoritmo extrae y almacena en tiempo real múltiples métricas asociadas a cada ensayo experimental. Entre ellas se incluyen la emoción objetivo solicitada al participante, la emoción predicha por el modelo en el último intento realizado, el nivel de confianza matemática asociado a dicha predicción (*Score*), así como el resultado final de la clasificación (éxito o fracaso). Adicionalmente, el sistema registra el tiempo de reacción transcurrido entre la exposición del estímulo y la validación exitosa de la emoción, junto con el número de intentos motores requeridos por el usuario antes de alcanzar el umbral de confianza establecido por el modelo.

6.3. Resultado del experimento

El sistema alcanzó una **exactitud (*accuracy*) global de 72,54 %**. La exactitud corresponde a la proporción total de predicciones correctas realizadas por el modelo respecto al número total de muestras evaluadas. En este caso, la métrica se obtuvo dividiendo la cantidad de clasificaciones correctas entre el total de ensayos experimentales ejecutados. Dado que el problema de reconocimiento emocional presenta asimetrías severas entre clases, el análisis se complementó con métricas de evaluación robustas al desbalance.

Métrica	Valor
Accuracy	72,54 %
Precision macro	88,91 %
Recall macro	72,54 %
F1-score macro	67,25 %

Tabla 6.1: Métricas globales del sistema.

La discrepancia observada entre la *precision* y el *recall* evidencia que, si bien el modelo clasifica un subconjunto de emociones con alta fiabilidad, presenta deficiencias para recuperar adecuadamente las clases semánticamente ambiguas.

6.3.1. Resultados por emoción

Como se observa en la Tabla 6.2, el rendimiento del modelo varía significativamente entre emociones, evidenciando una fuerte desigualdad en la capacidad de generalización.

Emoción	Precision (%)	Recall (%)	F1-score (%)	n
Enojo	100	98,44	99,21	64
Asco	100	4,69	8,96	64
Miedo	96,83	95,31	96,06	64
Felicidad	35,96	100	52,89	64
Neutral	94,12	100	96,97	64
Tristeza	95,45	98,44	96,92	64
Sorpresa	100	10,94	19,72	64

Tabla 6.2: Métricas desglosadas por clase emocional.

El desglose por clase revela una fuerte desigualdad en la capacidad de generalización del modelo:

- **Alto desempeño (F1-score >90 %):** Neutral, Enojo, Tristeza y Miedo.
- **Bajo desempeño (F1-score <20 %):** Asco y Sorpresa.

Destaca el comportamiento anómalo de la clase *Felicidad*, la cual registra un *recall* perfecto (100 %) penalizado por una precisión baja (35.96 %). Esto indica que el sistema utiliza iterativamente esta emoción como una clase dominante o absorbente ante la incertidumbre visual.

6.3.2. Matriz de confusión

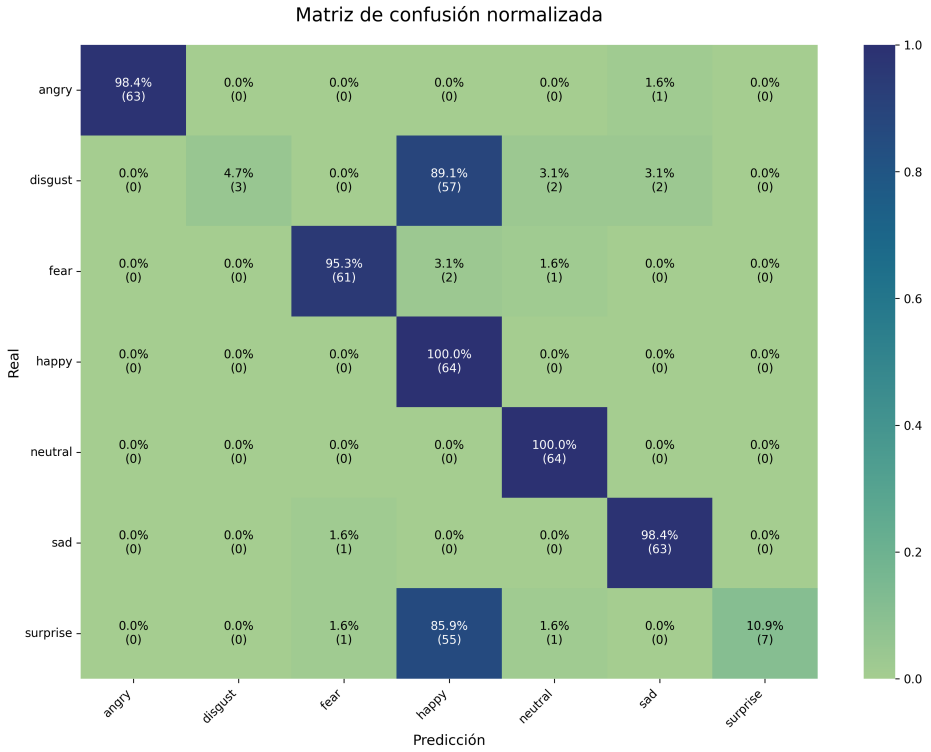


Figura 6.3: Matriz de confusión normalizada.

La matriz de confusión confirma la existencia de un sesgo estructural en el espacio predictivo. Se observa una migración sistemática de errores:

- Instancias de *Asco* se clasifican erróneamente como *Felicidad*.
- Instancias de *Sorpresa* se clasifican erróneamente como *Felicidad*.

Este fenómeno corrobora la tendencia del modelo a favorecer interpretaciones de valencia positiva en escenarios de baja separabilidad semántica.

6.3.3. Resultados por género

Género	**Accuracy (%)**	**Precision Macro (%)**	**F1 Macro (%)**	n
Femenino	73,21	90,68	67,88	224
Masculino	71,88	87,26	66,55	224

Tabla 6.3: Métricas de clasificación desglosadas por género del participante.

A nivel algorítmico, el modelo mantiene tasas de predicción estables con independencia del género del sujeto de prueba, demostrando consistencia frente a la variabilidad facial entre los integrantes de los grupos.

6.3.4. Resultados por fase

Fase	**Accuracy (%)**	**Precision Macro (%)**	**F1 Macro (%)**	n
PHASE_1	73,21	89,91	68,60	224
PHASE_2	71,88	87,99	65,89	224

Tabla 6.4: Métricas de clasificación en relación con la etapa y exposición del usuario.

La similitud de resultados entre fases confirma que el *accuracy* global indica consistencia del modelo y no corresponde a un ajuste del usuario por la repetición de intentos.

6.4. Análisis de Latencia y Métricas de Interacción

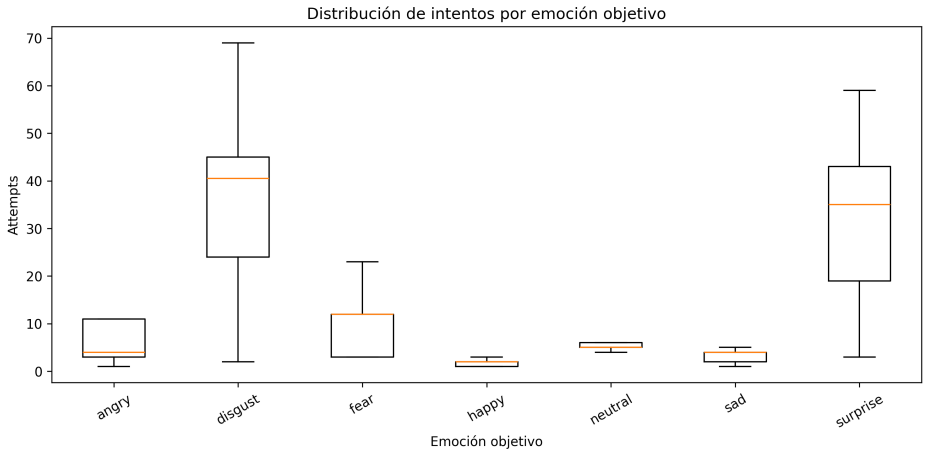


Figura 6.4: Distribución de intentos de interacción por emoción.

Las estadísticas de interacción reflejan que las emociones con métricas de desempeño algorítmico deficientes exigen empíricamente:

- Mayor tiempo de reacción por parte del usuario.
- Mayor número de intentos fallidos antes de lograr una activación exitosa.

Esto es un indicador directo de una alta carga cognitiva y de una mayor dificultad perceptual en la ejecución del gesto.

6.4.1. Análisis Cuantitativo de Latencia

A partir de los datos consolidados, se analizó la variable *Reaction_ Time* como indicador en la interacción humano-computador.

A nivel global, los tiempos de reacción exhiben una alta varianza, demostrando que la complejidad de la inferencia depende críticamente de la emoción subyacente. Las clases con menor tasa de acierto algorítmico (*Asco* y *Sorpresa*) imponen latencias medias que superan en un orden de magnitud

el promedio del sistema. Por el contrario, las categorías *Felicidad* y *Neutral* generan respuestas casi instantáneas.

Emoción	n	**RT Medio (ms)**	**RT Mediano (ms)**
Enojo	64	7531	6959
Asco	64	25 240	26 530
Miedo	64	6776	6958
Felicidad	64	1300	1409
Neutral	64	3711	3394
Tristeza	64	3023	2826
Sorpresa	64	25 877	22 411

Tabla 6.5: Latencia promedio y mediana por emoción objetivo (ms).

Los resultados demuestran la siguiente relacion entre desempeño del modelo predictivo y la carga operativa del usuario:

- **Interacción rápida:** Felicidad, Neutral.
- **Interacción moderada:** Enojo, Miedo, Tristeza.
- **Interacción Lenta:** Asco, Sorpresa.

Esta relación demuestra que las limitaciones de reconocimiento no provienen de un cuello de botella necesariamente computacional, sino de la variabilidad fisiológica y la ambigüedad expresiva del usuario.

6.4.2. Análisis demográfico: latencia por género

La división del rendimiento por género arroja diferencias en los tiempos medios de reacción. Las variaciones no superan los umbrales de significancia estadística, esto indica que no hay sesgos operativos demográficos en la usabilidad del sistema.

Género	n	RT Medio (ms)
Femenino	224	9081
Masculino	224	11 897

Tabla 6.6: Latencia promedio comparada por género.

6.4.3. Curva de aprendizaje: latencia por fase

El contraste temporal entre la primera y segunda mitad del experimento (`PHASE_1` frente a `PHASE_2`) documenta una leve contracción en los tiempos de respuesta. Sin embargo, la magnitud de esta mejora sugiere un nivel de habituación moderado, descartando una curva de aprendizaje pronunciada a corto plazo.

Fase	n	RT Medio (ms)
PHASE_1	224	12 245
PHASE_2	224	8733

Tabla 6.7: Evolución de la latencia promedio por fase experimental.

6.4.4. Interpretación general

La evidencia recopilada demuestra que:

- Las emociones penalizadas por un bajo *F1-score* originan invariablemente retardos sistémicos severos y obligan a la repetición de gestos.
- Las categorías sólidamente asimiladas por el modelo desencadenan validaciones rápidas y exigen menor sobrecarga física.
- La tasa de error del sistema desplegado es un fenómeno acoplado entre la certidumbre matemática de la red neuronal y la claridad cinética del operador.

6.5. Resumen de Resultados

El análisis experimental en el hardware de borde arroja las siguientes conclusiones estructurales:

- El sistema exhibe alta fiabilidad al clasificar emociones facialmente distintivas.
- Persiste un déficit de representación latente al evaluar categorías de alta ambigüedad semántica.
- Se detectó un desvío estadístico persistente hacia la predicción de *Felicidad* frente a señales inciertas.
- La dificultad técnica que el modelo enfrenta en el límite de decisión se manifiesta físicamente como fricción cognitiva en el usuario.
- El marco de inferencia resultó ser invariable y robusto a factores como el género y el nivel de habituación.

Capítulo 7

Discusión

7.1. Diferenciación conceptual de las emociones

Los resultados obtenidos durante el protocolo experimental Exp10 muestran que el sistema mantiene un desempeño relativamente consistente entre la validación *offline* y el despliegue en condiciones operativas «in-the-wild». La proximidad entre la exactitud obtenida durante entrenamiento-validación (74,41 %) y la alcanzada durante interacción real con usuarios (72,54 %) sugiere que la arquitectura basada en CLIP preserva parcialmente su capacidad de generalización fuera del dominio estricto de entrenamiento.

Sin embargo, el análisis demuestra que el comportamiento del modelo no es homogéneo entre las distintas categorías emocionales. Emociones como *Felicidad*, *Neutral*, *Enojo*, *Miedo* y *Tristeza* muestran métricas elevadas de recuperación y clasificación, mientras que *Asco* y *Sorpresa* exhiben un deterioro significativo en todas las métricas de desempeño.

Este fenómeno puede interpretarse desde la diferencia de la semántica de las expresiones faciales. Las emociones con configuraciones musculares más universales y distintivas poseen fronteras de decisión más claramente definidas en el espacio latente del modelo. *Asco* y *Sorpresa* suelen mostrarse mediante deformaciones parciales o de baja intensidad, dificultando la discriminación precisa en condiciones reales de operación.

Adicionalmente, CLIP fue originalmente entrenado para tareas generales de alineamiento texto-imagen y no específicamente para reconocimiento afectivo fino. Como consecuencia, aunque el modelo posee una capacidad repre-

sentacional elevada, no tiene intrínsecamente de especialización para resolver emociones facialmente ambiguas cuando las diferencias visuales entre clases son sutiles.

7.2. Sesgo predictivo hacia la emoción «felicidad»

La matriz de confusión revela un patrón de errores hacia la clase *Felicidad*. Una proporción significativa de las instancias correspondientes a *Asco* y *Sorpresa* fueron absorbidas por la clase *Felicidad*, evidenciando la existencia de un sesgo estructural en el espacio de representación del modelo.

Este comportamiento puede deberse parcialmente al desbalance presente en los datasets utilizados durante el entrenamiento. Tanto AffectNet como RAF-DB y FER2013 contienen una mayor cantidad de muestras asociadas a emociones positivas o neutras, lo que indica una priorización estadística de esas categorías.

Asimismo, desde el punto de vista semántico, las expresiones ambiguas o incompletas tienden a compartir regiones latentes próximas a configuraciones positivas, especialmente cuando el sujeto no logra ejecutar la expresión objetivo con suficiente intensidad muscular. En consecuencia, el modelo adopta una estrategia conservadora, proyectando entradas inciertas hacia clases con mayor dominancia estadística.

Esto evidencia que la incertidumbre predictiva no se distribuye aleatoriamente, sino que sigue direcciones específicas dentro del espacio de *embeddings*. Por lo tanto, el análisis del comportamiento del modelo requiere considerar no solo métricas globales de exactitud, sino también la dirección y naturaleza de los errores generados.

7.3. Relación entre desempeño algorítmico y rendimiento operacional

Se observa una correlación directa entre el desempeño del modelo y la carga operativa impuesta al usuario. Las emociones con menores tasas de reconocimiento presentaron simultáneamente mayores tiempos de reacción y un incremento considerable en la cantidad de intentos requeridos para alcanzar

una validación exitosa.

Este comportamiento sugiere que las dificultades del sistema no provienen exclusivamente de limitaciones computacionales, sino también de la complejidad fisiológica asociada a la ejecución voluntaria de determinadas expresiones faciales. Mientras emociones como *Felicidad* o *Neutral* generan respuestas rápidas y estables, expresiones como *Asco* y *Sorpresa* requieren una mayor modulación muscular y presentan una mayor variabilidad entre individuos.

Esto quiere decir que la experiencia de uso depende simultáneamente de dos factores: la capacidad discriminativa del modelo y la claridad biomecánica de la expresión ejecutada por el operador humano. En consecuencia, el rendimiento observado en escenarios reales constituye un fenómeno mezclado entre percepción computacional y variabilidad fisiológica.

7.4. Impacto del diseño basado en disparo geométrico

Esta estrategia de clasificar solo aquellos frames que fueron detectados por las condiciones por expresion facial al ver en la geometría de los landmarks superpuestos un cambio aceptable permitió separar la detección de actividad facial del procesamiento intensivo ejecutado en GPU, evitando realizar inferencia continua sobre cada fotograma capturado. Este diseño demostró funcionar para preservar recursos computacionales en la NVIDIA Jetson Xavier NX, manteniendo simultáneamente una interacción fluida en tiempo real. La reducción de procesamiento redundante permitió disminuir la carga sobre la GPU y mantener estabilidad operacional bajo restricciones severas de memoria y consumo energético.

Este enfoque introduce compensaciones importantes. El uso de umbrales geométricos fijos implica que micro-expresiones o deformaciones faciales de baja intensidad pueden no activar el disparador, quedando excluidas del análisis semántico. Por lo tanto, el sistema actual prioriza estabilidad y eficiencia computacional por sobre sensibilidad expresiva extrema.

Este compromiso entre precisión, sensibilidad y viabilidad operacional constituye una característica importante del despliegue de modelos complejos en hardware de borde con recursos limitados.

7.5. Limitaciones del análisis estático

El sistema analiza fotogramas individuales sin incorporar explícitamente la evolución temporal de la expresión facial, no se tiene información relacionada con el inicio, transición y relajación del gesto emocional.

Esta característica afecta particularmente a la discriminación entre emociones fisiológicamente próximas, como *Miedo* y *Sorpresa*, cuya diferenciación suele depender de patrones dinámicos más que de configuraciones faciales instantáneas.

La ausencia de modelamiento temporal limita la capacidad del sistema para capturar información relevante como también la robustez del reconocimiento en expresiones espontáneas o parcialmente desarrolladas. En este contexto, los resultados obtenidos respaldan la tendencia reciente del estado del arte hacia arquitecturas espacio-temporales capaces de incorporar secuencias de video y dinámica facial dentro del proceso inferencial.

7.6. Implicancias para la computación afectiva

Los resultados del presente trabajo permiten extraer implicancias importantes para el desarrollo de sistemas de computación afectiva desplegados en escenarios reales. En primer lugar, se valida la factibilidad de adaptar modelos multimodales de gran escala, originalmente diseñados para tareas generales de lenguaje-visión, al dominio específico del reconocimiento emocional facial.

Asimismo, la investigación demuestra que el rendimiento observado en laboratorio no representa necesariamente el comportamiento operativo final del sistema. Variables asociadas al entorno, calidad de captura, capacidad expresiva del usuario y restricciones de hardware modifican significativamente la dinámica de interacción entre el modelo y el operador humano.

Finalmente, la evidencia experimental sugiere que los principales desafíos actuales del reconocimiento emocional ya no se limitan únicamente a la disponibilidad de capacidad computacional, sino a la necesidad de construir representaciones semánticas y temporales ricas capaces de modelar adecuadamente la complejidad y ambigüedad de las expresiones humanas.

Capítulo 8

Conclusiones y trabajo futuro

8.1. Conclusiones generales

El presente trabajo de título logró diseñar, implementar y validar un sistema de reconocimiento de emociones faciales en tiempo real sobre hardware de borde, demostrando la posibilidad de desplegar modelos basados en *Vision Transformers* (CLIP) en plataformas de recursos limitados como NVIDIA Jetson Xavier NX.

A nivel de modelamiento, se comprobó que el modelo CLIP ViT-B/32, sometido a un proceso de *Deep Fine-Tuning*, es capaz de aprender representaciones semánticas de expresiones faciales, alcanzando una exactitud de validación de **74,41 %** en datasets conocidos (*AffectNet, RAF-DB, FER2013*). Este resultado muestra la capacidad de los modelos de atención global para capturar relaciones espaciales complejas en el rostro humano.

A nivel experimental, el protocolo Exp10 permitió validar el sistema en condiciones reales con usuarios, alcanzando una **accuracy global de 72,54 %**. Sin embargo, el análisis mediante métricas más completas evidenció que el desempeño no es homogéneo: mientras la *precision macro* alcanzó 88,91 %, el *F1-score macro* fue de 67,25 %, revelando una degradación significativa en clases específicas.

El principal aporte del trabajo reside en el procedimiento propuesto. El uso de un mecanismo de *disparo por evento geométrico* permitió separar la inferencia del modelo del flujo de captura, evitando la saturación de la GPU y logrando una experiencia en tiempo real fluida. Este enfoque representa una solución práctica para el despliegue de modelos en hardware de borde.

En términos globales, este trabajo demuestra que es viable trasladar modelos de inteligencia artificial multimodal hacia ecosistemas embebidos de recursos limitados, logrando un equilibrio efectivo entre capacidad y rendimiento en tiempo real. considerando las restricciones computacionales, los resultados dan a conocer que uno de los principales desafíos actuales reside en la complejidad del modelado emocional humano y en la ambigüedad fisiológica de las expresiones faciales.

8.2. Verificación de objetivos y validación

La evaluación global del sistema confirma el cumplimiento de los objetivos metodológicos y tecnológicos planteados. En primer lugar, la construcción y consolidación del conjunto de datos superó el desbalance de los entornos *in-the-wild* mediante la implementación de *class weighting*, mitigando exitosamente el riesgo de sesgo hacia las clases mayoritarias durante el entrenamiento. También, la estrategia de *Fine-Tuning* basada en el descongelamiento selectivo de capas demostró ser efectiva para especializar el modelo CLIP en el dominio facial, logrando la adaptación semántica sin incurrir en una pérdida de su capacidad de generalización. A nivel de despliegue, la integración de asincronía y la arquitectura de procesamiento condicionado por eventos validaron operativamente a la NVIDIA Jetson Xavier NX como una plataforma apta para ejecutar modelos complejos. En consecuencia, la validación experimental corroboró empíricamente que estas redes de alta capacidad pueden operar en tiempo real en escenarios reales; no obstante, el desempeño del sistema no solo se debe a las capacidades algorítmicas, sino también a la claridad expresiva y biomecánica del usuario.

8.3. Limitaciones del sistema

A partir de la evaluación integral, se identificaron restricciones tanto algorítmicas como físicas que limitan el alcance de la solución propuesta. El sistema exhibe una marcada dificultad para discriminar expresiones facialmente ambiguas, agravada por la tendencia del modelo a favorecer clases positivas. Esta limitación se ve marcada por la falta de modelamiento temporal; al operar sobre fotogramas aislados, el modelo descarta por completo la dinámica cinemática y la evolución natural de la expresión facial. Desde

el punto de vista de la usabilidad, la precisión en entornos reales depende de la capacidad de expresión del usuario, introduciendo una varianza poco predecible.

8.4. Trabajo futuro

Las limitaciones vistas trazan una ruta posible para la optimización del presente proyecto. Como prioridad de modelado, resulta imperativa la transición hacia un análisis secuencial que incorpore información temporal, permitiendo al sistema capturar las fases de inicio, clímax y relajación de los gestos faciales. En paralelo, es necesario refinar el espacio semántico de la red mediante ajustes en la ingeniería de *prompts* y la proyección de *embeddings*, con el fin de maximizar la separabilidad matemática entre clases conflictivas.

A un nivel superior, la evolución natural del sistema apunta hacia la fusión multimodal, integrando variables acústicas o señales fisiológicas suplementarias para dotar a la inferencia de mayor resiliencia ante oclusiones o ruido visual. Por último, en el ámbito de la optimización para, los esfuerzos de ingeniería de software deben orientarse a migrar los algoritmos de detección facial inicial directamente hacia la GPU, unificando el flujo de tensores y mitigando los retardos de transferencia entre la memoria del sistema y la memoria de video.

Nota del autor: Durante la etapa de revisión de este manuscrito, se emplearon herramientas de asistencia basadas en inteligencia artificial exclusivamente con el fin de mejorar la fluidez gramatical, el estilo y la ortografía del texto.

Bibliografía

[1] Shahanur Alam, Chris Yakopcic, and Qing Wu. Survey of deep learning accelerators for edge and emerging computing. *Electronics*, 13(15):2988, 2024. doi: 10.3390/electronics13152988.

[2] Marian Stewart Bartlett, Gwen Littlewort, Mark Frank, Claudia Lainscsek, Ian Fasel, and Javier Movellan. Fully automatic facial action recognition in spontaneous behavior. *IEEE Transactions on Systems, Man, and Cybernetics*, 2006.

[3] Haodong Chen, Haojian Huang, Junhao Dong, Mingzhe Zheng, and Dian Shao. Finecliper: Multi-modal fine-grained clip for dynamic facial expression recognition with adapters. *arXiv preprint arXiv:2407.02157*, 2024. URL `https://arxiv.org/abs/2407.02157`.

[4] Ruth Cordova-Cardenas, Daniel Amor, and Álvaro Gutiérrez. Edge ai in practice: A survey and deployment framework for neural networks on embedded systems. *Electronics*, 14(24):4877, 2025. doi: 10.3390/electronics14244877.

[5] NVIDIA Corporation. *NVIDIA Jetson Xavier NX Module Data Sheet*, 2020. URL `https://developer.nvidia.com/embedded/jetson-xavier-nx`. DS-09876-001_v1.0.

[6] David Cristinacce and Timothy F Cootes. A multi-stage approach to facial feature detection. In *British Machine Vision Conference (BMVC)*, 2004. Referencia clásica sobre el enfoque coarse-to-fine (de lo general a lo específico) en análisis facial, justificando la separación entre la búsqueda geométrica ligera y el análisis profundo posterior.

[7] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for

image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021.

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, and Sylvain Gelly. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[9] Chengjun Duan. A survey of facial expression recognition in the wild. *Applied and Computational Engineering*, 6(1):173–181, 2023. doi: 10.54254/2755-2721/6/20230760.

[10] Paul Ekman. An argument for basic emotions. *Cognition & emotion*, 6 (3-4):169–200, 1992.

[11] Paul Ekman and Wallace V Friesen. Constants across cultures in the face and emotion. *Journal of personality and social psychology*, 17(2): 124–129, 1971.

[12] Ali Pourramezan Fard, Mohammad Mehdi Hosseini, and Mohammad H Mahoor. Soft-labels for noise-robust and imbalanced facial expression recognition. *arXiv preprint arXiv:2410.22506*, 2024.

[13] Rosa A. García-Hernández, Huizilopoztli Luna-García, and José M. Celaya-Padilla. A systematic literature review of modalities, trends, and limitations in emotion recognition, affective computing, and sentiment analysis. *Applied Sciences*, 14(16):7165, 2024. doi: 10.3390/app14167165.

[14] Ian J. Goodfellow, Dumitru Erhan, Pierre-Luc Carrier, Aaron Courville, Mehdi Mirza, Ben Hamner, William Cukierski, Yichuan Tang, David Thaler, Dong-Hyun Lee, Yingbo Zhou, Chetan Ramaiah, Fangxiang Feng, Ruifan Li, Xiaojie Wang, Dimitris Athanasakis, John Shawe-Taylor, Maxim Milakov, John Park, Radu Ionescu, Marius Popescu, Cristian Grozea, James Bergstra, Jingjing Xie, Lukasz Romaszko, Bing Xu, Zhang Chuang, and Yoshua Bengio. Challenges in representation learning: A report on three machine learning contests. *Neural Information Processing Systems (NeurIPS) Workshop*, 2013.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[16] Yunxin Huang, Fei Chen, Shaohe Lv, and Xiaodong Wang. Facial expression recognition: A survey. *Symmetry*, 11(10):1189, 2019. doi: 10.3390/sym11101189.

[17] Vahid Kazemi and Josephine Sullivan. One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1867–1874, 2014.

[18] Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. Transformers in vision: A survey. *ACM computing surveys (CSUR)*, 54(10s):1–41, 2022.

[19] Thomas Kopalidis, Vassilios Solachidis, Nicholas Vretos, and Petros Daras. Advances in facial expression recognition: A survey of methods, benchmarks, models, and datasets. *Information*, 15(3):135, 2024. doi: 10.3390/info15030135.

[20] Mohammad Rami Koujan, Luma Alharbawee, Anastasios Roussos, and Stefanos Zafeiriou. Real-time facial expression recognition "in the wild" by disentangling 3d expression from identity. In *IEEE International Conference on Automatic Face and Gesture Recognition*, 2020.

[21] Shan Li and Weihong Deng. Reliability-aware distribution learning for facial expression recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[22] Shan Li, Weihong Deng, and JunPing Du. Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2852–2861, 2017.

[23] Fuyan Ma, Yiran He, Bin Sun, and Shutao Li. Multimodal prompt alignment for facial expression recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12581–12591, October 2025. URL `https://openaccess.thecvf.`

com/content/ICCV2025/html/Ma_Multimodal_Prompt_Alignment_for_Facial_Expression_Recognition_ICCV_2025_paper.html.

[24] Ignacio Martin-Salinas, Jose M. Badia, and Valls. Evaluating and accelerating vision transformers on gpu-based embedded edge ai systems. *The Journal of Supercomputing*, 81:349, 2025. doi: 10.1007/s11227-024-06807-1.

[25] Ali Mollahosseini, Behzad Hasani, and Mohammad H Mahoor. Affectnet: A database for facial expression, valence, and arousal computing in the wild. *IEEE Transactions on Affective Computing*, 10(1):18–31, 2017.

[26] Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. When does label smoothing help? *Advances in neural information processing systems*, 32, 2019.

[27] NVIDIA. Deploying to deepstream for multitask classification, 2025. TAO Toolkit / DeepStream documentation.

[28] OpenVINO Toolkit. Interactive face detection c++ demo, 2023. Open Model Zoo documentation.

[29] Maja Pantic and Leon J. M. Rothkrantz. Automatic analysis of facial expressions: The state of the art. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1424–1445, 2000.

[30] Rosalind W. Picard. *Affective Computing*. MIT Press, Cambridge, MA, 1997.

[31] Alec Radford, Jong Wook Kim, and Chris Hallacy. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021.

[32] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, and Jack Clark. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[33] Sergi Ramis and Joan M. Buades. Using a social robot to evaluate facial expressions in the wild. *Sensors*, 20(23):6878, 2020. doi: 10.3390/s20236878.

[34] Muhammad Sajjad, Fath U Min Ullah, Mohib Ullah, Georgia Christodoulou, Faouzi Alaya Cheikh, Mohammad Hijji, Khan Muhammad, and Joel JPC Rodrigues. A comprehensive survey on deep facial expression recognition: Challenges, applications, and future guidelines. *Alexandria Engineering Journal*, 68:817–840, 2023.

[35] Tim Sennett. Unfreezing the layers you want to fine-tune using transfer learning. Medium, 2023. URL `https://url-shortener.me/K0CO`. Accessed: 2024-05-20.

[36] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *International Conference on Machine Learning (ICML)*, 2019.

[37] Ying-li Tian, Takeo Kanade, and Jeffrey F. Cohn. Recognizing action units for facial expression analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(2):97–115, 2001.

[38] Sana Ullah, Jie Ou, Yuanlun Xie, and Wenhong Tian. Facial expression recognition (fer) survey: a vision, architectural elements, and future directions. *PeerJ Computer Science*, 10:e2024, 2024. doi: 10.7717/peerj-cs.2024.

[39] Michel Valstar, Bihan Jiang, Marc Mehu, Maja Pantic, and Klaus Scherer. Meta-analysis of the first facial expression recognition challenge. *IEEE Transactions on Systems, Man, and Cybernetics*, 2012.

[40] Ashish Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[41] Wavestore. Why edge-based video surveillance is a game-changer for real-time video, 2023. URL `https://url-shortener.me/K0D6`. Contextualiza y justifica la inferencia disparada por eventos (event-triggered) en dispositivos Edge para evitar el procesamiento redundante de fotogramas sin actividad relevante.

[42] Shice Zhang. Data-fusion-based two-stage cascade framework for multi-modality face anti-spoofing. *IEEE Transactions on Information Forensics and Security*, 2021. Documenta el uso de arquitecturas en cascada de dos etapas (Two-stage cascade) para procesar rostros, filtrando información en la primera etapa para optimizar la inferencia en redes neuronales profundas en la segunda etapa.

[43] Zhi Zhang, Jeffrey M. Girard, Ying Wu, Xiaoming Zhang, Peng Liu, Umur Ciftci, Shaun Canavan, Michael Reale, Alexander Horowitz, Hui Yang, and Lijun Yin. Spontaneous facial expression recognition: A survey. *Image and Vision Computing*, 32(10):701–711, 2014.

[44] Zengqun Zhao and Ioannis Patras. Prompting visual-language models for dynamic facial expression recognition. In *34th British Machine Vision Conference (BMVC)*, 2023. URL `https://papers.bmvc2023.org/0098.pdf`.

[45] Zhi-Qiang Zhao, Peng Zheng, Shoutao Xu, and Xindong Wu. Real-time video processing with deep learning: A survey. *IEEE Access*, 7:71696–71726, 2019.