UNIVERSIDAD DE ANTIOQUIA

Análisis de arquitecturas de aprendizaje profundo para el modelamiento de rostros y expresiones faciales

Autor

Luis Felipe Gómez Gómez

Universidad de Antioquia

Facultad de Ingeniería, Departamento de Ingeniería Electrónica y Telecomunicaciones

Medellín, Colombia

2021

Análisis de arquitecturas de aprendizaje profundo para el modelamiento de rostros y expresiones faciales

Luis Felipe Gómez Gómez

Trabajo de investigación presentado como requisito para optar al título de:
Magíster en Ingeniería

Asesores (a):
Prof. Dr.-Ing. Juan Rafael Orozco Arroyave

Línea de Investigación:
Análisis de patrones
Grupo de Investigación:
Grupo de Investigación en Telecomunicaciones Aplicadas (GITA)

Universidad de Antioquia
Facultad de Ingeniería, Departamento de Ingeniería Electrónica y Telecomunicaciones
Medellín, Colombia
2021.

Agradecimientos

Principalmente agradezco a mi madre Arcenia Gómez y a mi padre Rodrigo de Jesús Gómez Monsalve, quienes, con un gran apoyo, consejos y motivaciones, brindaron todo lo necesario para obtener este logro y los próximos que se avecinan. A mi hermana Erica y a mis hermanos Rodrigo y Juan, que siempre me han apoyado. A toda mi familia, porque en los momentos que he necesitado apoyo, ellos siempre han estado ahí.

Agradezco a mis amigos y compañeros Jhon Lopera, Paula Perez, Cristian Rios, Daniel Escobar, Felipe Orlando y Felipe Parra, por toda la ayuda, consejos y discusiones que tuvimos durante estos años y decirles que los considero muy buenos amigos. Igualmente agradecerle a todos los miembros del grupo de investigación GITA que hicieron que todas las horas de trabajo se volvieran mas amenas.

Agradecerle a mi asesor Prof. Dr.-Ing. Juan Rafael Orozco Arroyave, quien me dio la oportunidad de profundizar en mis estudios académicos, y que con sus consejos y conocimientos, me ayudaron a entender que la trayectoria académica que deseo puede lograrse con mucho esfuerzo.

Igualmente agradecerle al profesor Aythami Morales y a todos los integrantes grupo de investigación Biometrics and Data Pattern Analytics - BiDA Lab en Madrid, España, que me acogieron en sus instalaciones y me brindaron sus conocimientos para hacer realidad este trabajo. Al Ministerio de Ciencia Tecnología e Innovación (Minciencias), por permitirme realizar una pasantía académica financiada por el Fortalecimiento de programas y proyectos en ciencias médicas y de la salud con talento joven e impacto regional del Ministerio de Ciencia, Tecnología e Innovación (Minciencias) – RC 752 - 2018 en el grupo GITA.

Finalmente agradecerle a todos los pacientes y voluntarios de Fundalianza Parkinson Colombia. Sin su ayuda y voluntad de colaborar en este trabajo nada de esto hubiera sido posible.

Índice general

Capítulo 1

Introducción

1.1. Motivación

La enfermedad de Parkinson es el segundo desorden degenerativo más prevalente a nivel mundial después del Alzheimer [1]. El Parkinson afecta entre el 1 % y 2 % de las personas mayores de 65 años [2]. Según un estudio realizado, el número de personas con la enfermedad de Parkinson en los 10 países mas poblados del mundo, llegará a entre 8.7 - 9.3 millones en 2030 [3]. En Colombia, la prevalencia de la enfermedad es cerca de 176.4 casos por cada 100,000 habitantes y la región con mas alta prevalencia es Antioquia con 30.7 casos por cada 100,000 habitantes [4].

La enfermedad de Parkinson se caracteriza por la pérdida progresiva de dopamina en la sustancia negra del cerebro. Los problemas que induce la enfermedad de Parkinson incluyen déficit motores como la bradicinesia, rigidez, inestabilidad en la postura y movimiento involuntario en reposo. De acuerdo con la literatura, gran parte de los pacientes con EP muestran anomalías en el rostro como lentitud en el parpadeo, menos amplitud en los movimientos de los músculos faciales, ojos más abiertos y en algunos casos boca entreabierta, comúnmente diagnosticado como hipomimia [5]. Estos síntomas hacen que el rostro de los pacientes se torne rígido, el habla se vuelve más lenta, se pierde movilidad en los labios, cejas y cabeza, generando dificultades para la comunicación oral, esta inexpresividad facial puede causar aislamiento social y frustración en los pacientes con EP.

La habilidad de generar expresiones faciales y el estado neurológico de los pacientes con enfermedad de Parkinson es tomado por neurólogos expertos que se basan en la historia médica, en exámenes físicos y en exámenes neu-

rológicos para evaluar a los pacientes. Sin embargo, las habilidades motoras de los pacientes con enfermedad de Parkinson se ven afectadas, por lo que visitar un hospital para realizar exámenes o evaluaciones médicas no es una tarea sencilla para ellos [6]. Además, el diagnóstico y el seguimiento de los síntomas de la enfermedad de Parkinson son largos, costosos y son lejanos del área de residencia de los pacientes. Por estas razones hay un interés creciente de la comunidad científica para desarrollar sistemas asistidos por computador para un seguimiento de la enfermedad de Parkinson en pacientes. Adicionalmente, el monitoreo continuo de los pacientes con enfermedad de Parkinson podría ayudar a tomar decisiones oportunas con respecto a su medicación y su terapia.

La severidad de la enfermedad es evaluada por neurólogos expertos utilizando muchas escalas y diferentes pruebas. Una de esas escalas es la Escala de Calificación de la Enfermedad de Parkinson de la Sociedad de Trastornos de Movimiento (Movement Disorder Society-Unified Parkinson's Disease Rating Scale, MDS-UPDRS-III) [7]. Esta escala considera tanto los síntomas motores como los no motores, dando una vista a la progresión de la enfermedad en los pacientes. La escala MDS-UPDRS-III se divide en 4 secciones y la evaluación de la habilidad de expresiones faciales es solo un ítem de la sección 3. En esta sección nos enfocaremos en los trabajos realizados en la detección de expresiones faciales y en como algunos de ellos han sido de utilidad en la detección y seguimiento de la hipomimia en los pacientes con EP.

1.2. Estado del arte

1.2.1. Análisis de expresiones faciales

El uso de sistemas automáticos para el análisis de expresiones faciales ha tenido una alta demanda en las últimas décadas gracias al uso de la inteligencia computacional, principalmente enfocado en las técnicas que se basan en el aprendizaje profundo (Deep Learning, DL). Estas arquitecturas de DL necesitan una gran cantidad de datos para crear una representación fiable de las características, pero ese no es el caso en algunos problemas donde las bases de datos disponibles son pequeñas, lo cual limita la posibilidad de desarrollar una arquitectura propia desde cero y puede causar sobre-entrenamiento. Para mitigar estos problemas se utilizan métodos de transferencia de conocimiento que ayudan a disminuir el sobreentrenamiento y además, ayudan a aumentar

el rendimiento de los sistemas.

La transferencia de conocimiento son técnicas utilizadas para mejorar el rendimiento de los modelos utilizando como pesos iniciales modelos desarrollados para la clasificación de un problema A y reentrenarlos para la clasificación de un problema B. Estas técnicas son utilizadas en redes neuronales convolucionales (CNN) y redes neuronales recurrentes (RNN). Trabajos previos muestran que la transferencia de conocimiento incrementa el rendimiento de los modelos originales, arquitecturas como AlexNet [8], VGG-Face [9] y FaceNet [10], son ejemplos de donde se aplicaron estos métodos. El trabajo de Kaya et al en [11] propone el uso de reconocimiento de emociones en video, en este trabajo los autores emplean una arquitectura pre-entrenada VGG-Face para el reconocimiento de rostros y reentrenarla con la base de datos FER2013 [12]. Los resultados muestran un incremento aproximado del 6 % en el conjunto de datos de validación, aumentando de un 44,1 % a un 50,8 %.

Otro trabajo a considerar es el realizado por Breuner y Kimmel et al en [13], ellos usan una red neuronal convolucional (Convolutional Neural Network, CNN) para la clasificación de emociones en 3 bases de datos (FER2013 [12], CK+ [14] y NovaEmotions [15]). Un punto de interés de este trabajo es que el mapeo de zonas de la imagen que tienen mayor influencia en la decisión de si la emoción fue felicidad, tristeza, disgusto o ira, están muy relacionados con zonas especificas del rostro, mas comúnmente llamados unidades de acción de las expresiones faciales, definidos por Ekman en [16].

Sajjanhar et al en [17], investiga el uso de arquitecturas del estado del arte para el reconocimiento de expresiones faciales. Ellos evalúan el desempeño de las arquitecturas Inception-v3 y VGG-19 que inicialmente entrenadas para el reconocimiento de objetos y VGG-Face que fue entrenada para el reconocimiento de rostros. Los experimentos son realizados con 3 bases de datos (CK+ [14], JAFFE [18] y FACES [19]) y consideran 3 tipos de imágenes, (1) Una imagen del rostro denominado Región de Interés (ROI), (2) La diferencia entre la imagen en la que se muestra totalmente la expresión facial y su imagen en un estado neutral, y (3) el uso de Patrón Local Binario (LBP) de la imagen del rostro. Enfocándonos en el trabajo realizado sobre el primer conjunto de datos (Imágenes de los rostros), los autores determinan una línea base para evaluar el desempeño de las arquitecturas previamente entrenadas, esto utilizando una CNN con dos capas convolucionales y dos capas totalmente conectadas obteniendo aciertos del 73.53 %, 56.26 %, y 67.38 % para

las bases de datos CK+, JAFFE, y FACES respectivamente, en el momento de utilizar las arquitecturas preentrenadas se obtienen aciertos del 91.37 % en la base de datos CK+ en la arquitectura VGG-Face y se obtienen aciertos del 94.71 % y del 97.16 % para las base de datos JAFFE y FACES respectivamente, en la arquitectura VGG-19.

Cheng et al en [20], introdujeron un sistema de reconocimiento de expresiones faciales utilizando una CNN poco profunda y transferencia de conocimientos. La base de datos CK+ [14] se utiliza para entrenar a la CNN poco profunda y el proceso TL se realiza a la arquitectura VGG-19 que fue entrenada previamente con el conjunto de datos de ImageNet [21]. Compararon su trabajo con otros dos trabajos ([22] y [23]) donde se utilizan las mismas bases de datos y mostraron mejores resultados, con precisiones de hasta el 96 % para VGG-19 y el 88.3 % para la CNN poco profunda. Además de mejores precisiones, las arquitecturas introducidas por Cheng et al son más rápidas al entrenar la CNN poco profunda. En 2020 [24] Sonawane y Sharma presentan un resumen de las técnicas automáticas y el uso de aprendizaje de maquina en la detección emocional de expresiones faciales en los pacientes de EP. Los autores muestran que el uso de aprendizaje profundo en este campo no ha abordado en la clasificación entre personas sanas y pacientes con EP. Igualmente, ellos conducen un experimento piloto basado en el uso de una CNN entrenada desde cero para la detección de rostros con hipomimia. El experimento piloto muestra que los modelos basados en el aprendizaje profundo puede ser utilizados para la clasificación.

1.2.2. Análisis de expresiones faciales para modelar hipomimia

Simons et al en [25], realizan un estudio en 19 pacientes con EP y 25 controles sanos, los cuales realizaron tareas para evocar expresiones faciales emocionales, las cuales incluyen la visualización de videos, realización de iteraciones sociales; además posar e imitar diferentes expresiones faciales. La expresividad de los participantes se clasificó con escalas subjetivas, escalas objetivas y auto-cuestionarios. El resultado del estudio evidenció que los pacientes con EP mostraron una reducción en la expresividad facial espontánea en todas las situaciones experimentales. Además, los pacientes mostraron más dificultades en el momento de expresar e imitar las expresiones faciales que los controles. Dos años más tarde en [26], los autores presentan un trabajo donde se estudia la expresividad y la bradiquinesia. Los autores presentan la

hipótesis que los movimientos faciales voluntarios son lentos (bradiquinesia) y con menos amplitud en los movimientos con los pacientes con EP que en controles sanos. Esta hipótesis fue inspirada bajo las observaciones realizadas en la caminata de los pacientes, donde los movimientos son igualmente intencionales y estos igualmente se reducen. Videos digitalizados fueron evaluados imagen por imagen, analizando la entropía de los cambios temporales de la intensidad de los píxeles. Los autores encuentran que los pacientes con EP tienen una reducción en la entropía comparándolo con los controles sanos y fueron significativamente más lentos en alcanzar una expresión máxima ($p < 0,0001$), que está directamente asociada a la bradiquinesia. En 2016 Almutiry et al [27] presentó quizás el único estudio longitudinal en la evaluación de expresividad facial en pacientes con EP. Un total de 8 participantes (4 EP y 4 controles sanos). Los pacientes se registraron durante cinco días a la semana (una vez al día) durante seis semanas, mientras que los controles se registraron durante cinco días en una semana. Se pidió a los participantes que produjeran expresiones faciales específicas mientras estaban grabando. Los autores utilizaron dos métodos clásicos de extracción de rasgos para localizar 27 rasgos faciales: el modelo de apariencia activa (Active Appearance Model, AAM) y el modelo local restringido (Constrained Local Model, CLM). Los resultados sugieren que los pacientes con EP muestran menos movimiento que los controles, lo que confirma las observaciones hechas diez años antes por Bowers et al en [26]. En 2017, Gunnery et al [28], estudiaron la coordinación de movimientos a través de las regiones de la cara en 8 pacientes con EP (4 mujeres). Utilizaron el sistema de codificación de acción facial [29], [30] para medir las expresiones faciales espontáneas. El número de cuadros activados por unidad de acción y su intensidad fue etiquetado manualmente. Se calcularon las correlaciones para los valores de activación obtenidos a través de diferentes regiones de la cara. Los resultados mostraron que a medida que aumentaba la gravedad de la falta de expresiones faciales, había una disminución en el número, duración, intensidad y co-activación de la acción de los músculos faciales. Por otra parte, Andrea Bandini et al en [31] clasifica las emociones expresadas por 17 pacientes con EP (13 hombres) y un número iguales de controles (6 hombres). Diferentes emociones como neutral, ira, disgusto, felicidad y tristeza. Los autores consideran 49 puntos de referencia distribuidos en ojos, cejas, boca y nariz, de las cuales se extraen 20 características definidas como la combinación lineal de puntos de referencia específicos. La clasificación es realizada con una maquina de soporte vectorial

(SVM) con la que reportan aciertos en ira de 42.51 % en sanos y 29.32 % en pacientes con EP, en disgustos de 43.42 % en sanos y 35.00 % en pacientes con EP, en felicidad de 31.58 % en sanos y 18.09 % en pacientes con EP y en tristeza de 5.50 % y EP 9.30 %, siendo tristeza la emoción que presenta los menores aciertos entre las 5 emociones y siendo el único caso donde el porcentaje en pacientes con EP es mayor.

Otras contribuciones en el análisis de expresiones faciales en EP incluyen el trabajo de Kang et al [32]. Los autores evalúan las deficiencias que ocurren en los movimientos orofaciales de un grupo de 20 pacientes con EP y 20 controles, tanto en expresiones espontáneas y en expresiones voluntarias. Las activaciones musculares relacionadas con regiones específicas en el rostro, son comparadas considerando las señales de electromiografía. Ellos utilizaron la base de datos East Asian Dynamic Facial Expression Stimuli (EADFES) [33]. Los autores reportan limitaciones de los pacientes para expresar sus emociones de forma espontánea, aunque la dinámica observada en el movimiento de la cara es similar en todos los sujetos. El estudio también señala el deterioro de la calidad de vida del paciente debido a la presencia de "cara enmascarada", que afecta los aspectos sociales y psicológicos de los pacientes y aumenta su riesgo de desarrollar síntomas relacionados con la depresión. Recientemente, el trabajo de Grammatikopoulou et al en [34], se realiza un análisis de expresiones faciales a través de imágenes tomados directamente del celular, del que se extraen características geométricas del rostro y se almacenan en la nube. En este trabajo participan 34 personas divididas en 11 personas sanas y 23 pacientes con EP que son divididos en 3 grupos respecto al ítem de expresión facial en la escala MDS-UPDRS-III. La metodología que trabajan es el uso de Google Face API y Microsoft Face API para la extracción de dos conjuntos de características. El conjunto de características es conformado por los puntos de referencia distribuido en el rostro y luego el uso de estos conjuntos en dos modelos de regresión lineal con el cual se obtuvo dos valores de índice de severidad de hipomimia denominados (HSi1 y HSi2), el cual es usado para clasificar a los pacientes con EP y a la gente sana, obteniendo valores de sensibilidad y especificidad de 0.79 y 0.82 para el HSi1, y valores de sensibilidad y especificidad de 0.89 y 0.73 para el HSi2.

1.3. Objetivos

1.3.1. Objetivo General

Estudiar, implementar y evaluar arquitecturas de aprendizaje profundo para el reconocimiento de diferentes expresiones faciales aplicado a sistemas en el modelamiento de la hipomimia sufrida por pacientes con enfermedad de Parkinson.

1.3.2. Objetivos específicos

- ✓ Implementar y evaluar diferentes arquitecturas de aprendizaje profundo para la detección de diferentes gestos faciales en una persona.
- ✓ Implementar y evaluar arquitecturas de aprendizaje profundo para la verificación biométrica de identidad a través del rostro.
- ✓ Implementar y evaluar la capacidad del sistema para evaluar hipomimia.

1.4. Pregunta de investigación

Dado los diferentes estudios encontrados en la sección 1.2, la EP es una enfermedad que afecta las capacidades en la expresión facial de los pacientes en diferentes grados de severidad, con base en esto es posible realizarles un seguimiento al estado de la hipomimia en los pacientes, utilizando representaciones creadas con sistemas de aprendizaje automático. De acuerdo con lo planteado, se definen cuatro preguntas de investigación:

- ✓ ¿El uso de arquitecturas entrenadas para el reconocimiento de rostros, permiten modelar la hipomimia en pacientes con Parkinson?
- ✓ ¿El uso de arquitecturas entrenadas con transferencia de conocimientos para el reconocimiento de unidades de acción, permiten modelar la hipomimia en pacientes con Parkinson?
- ✓ ¿El uso de funciones de aprendizaje por similitud para potenciar y mejorar la extracción de información, permiten modelar la hipomimia en pacientes con Parkinson?

- ✓ ¿Los espacio de representación de diferentes dominios pueden mostrar mayor claridad en la discriminación entre pacientes y personas sanas?

1.5. Contribución de este trabajo de investigación

De acuerdo con lo mostrado en la revisión de la literatura, hay una falta de trabajo en el campo de reconocimiento de expresiones faciales para el modelamiento de la hipomimia en pacientes con EP con técnicas de aprendizaje profundo. Este estudio está enfocado en la utilización de técnicas de transferencia de conocimiento para la creación de diferentes modelos de reconocimiento de unidades de acción facial y lograr la construcción de vectores embebidos para la generalización de las características que afectan los músculos faciales que contribuyen en la hipomimia en los pacientes con EP. La principal contribución de este trabajo incluye: (1) Uno de los pocos acercamientos del aprendizaje profundo en el modelamiento de la hipomimia en pacientes con EP, (2) nuestros experimentos demuestran que, con el propósito de discriminar entre pacientes con EP y sujetos sanos, el uso de secuencias de fotogramas múltiples (es decir, con múltiples imágenes segmentadas durante la producción de expresiones faciales) es mejor que el uso de fotogramas individuales, (3) el enfoque de transferencia de conocimiento se utiliza para mejorar los modelos inicialmente entrenados para detectar unidades de acción facial y emociones, lo que resulta en un modelo adaptado para la discriminación automática entre pacientes con EP y sujetos sanos, y (4) el uso de un enfoque de triple pérdida para mejorar y adaptar el modelo encontrado con el transferencia de conocimiento para modelar la hipomimia en pacientes con EP.

1.6. Estructura del trabajo de investigación

Capitulo 2: Describe los métodos utilizados para evaluar la clasificación de pacientes con EP y personas sanas por medio de sus rostros. Además, describe los modelos de aprendizaje profundo y técnicas de representación utilizadas para modelar y predecir el estado de los pacientes con EP.

Capitulo 3: Incluye información del sistema y proceso de captura de la base de datos FacePark. Igualmente, contiene información del grupo de control y de los pacientes que participaron en las sesiones de grabación, incluyendo su evaluación en la escala MDS-UPDRS-III.

Capitulo 4: Describe la metodología implementada para la creación de los diferentes espacios de representación y las bases de datos utilizadas para la clasificación de pacientes y controles sanos, y la clasificación del estado neurológico.

Capitulo 5: Describe a detalle todo el marco experimental desarrollado en este trabajo.

Capitulo 6: Presenta detalles de los experimentos realizados con los datos descritos. Experimentos que consideran, en diferentes niveles, evaluar la capacidad de los modelos o arquitecturas de aprendizaje profundo para extraer información a través de los dominios en los que fueron entrenados.

Capitulo 7: Incluye el análisis de los resultados obtenidos en este trabajo y finaliza con las conclusiones derivadas del mismo.

Capítulo 2

Análisis de expresiones faciales en pacientes con Parkinson

El estado neurológico de los pacientes con EP es evaluada por neurólogos expertos, los cuales revisan el historial clínico y realizan examenes físicos a los pacientes. El estado neurológico de un paciente es evaluado en una escala Sociedad de Trastornos del Movimiento - Escala Unificada de Calificación de la Enfermedad de Parkinson (MDS-UPDRS-III) [7]. Esta escala considera tanto los síntomas motores como los no motores, dando una visión de la progresión de la enfermedad en un paciente. El total de la escala MDS-UPDRS-III es dividida en 4 secciones. La primer sección trabaja en los aspectos no-motores de las experiencia de la vida diaria, como deterioro cognitivo, depresión, ansiedad, fatiga, entre otros. La segunda sección considera aspectos motores de las experiencias de la vida diaria que incluyen actividades como comer, escribir y temblores en actividades. La tercer sección concierne la exploración motora que incluye lenguaje, expresión facial, rigidez y golpeteo de dedos, y la cuarta sección trata de las complicaciones motoras como el tiempo que pasa sin medicación. Todos los ítems evaluados en cada sección de la escala MDS-UPDRS tienen valoraciones entre 0 y 4, siendo 0 completamente sano y 4 completamente comprometido. Por otro lado, la evaluación de la expresión facial es sólo un ítem de la sección III de la escala MDS-UPDRS. En este ítem se definen los siguientes 5 niveles de la hipomimia:

0. Normal: Expresión facial normal.

1. Mínimo: Mínima amimia, manifestada únicamente por disminución de la frecuencia del parpadeo.

2. Leve: Además de la disminución de la frecuencia de parpadeo, también presenta amimia en la parte inferior de la cara, es decir hay menos movimientos alrededor de la boca, como menos sonrisa espontánea, pero sin apertura de los labios.

3. Moderado: Amimia con apertura de labios parte del tiempo cuando la boca está en reposo.

4. Grave: Amimia con apertura de labios la mayor parte del tiempo cuando la boca está en reposo.

2.1. Modelo de emociones usando unidades de acción

El modelamiento de las emociones básicas está limita en la capacidad de representar la complejidad y la sutileza de nuestras demostraciones sociales diarias. Dadas estas limitaciones, los modelos de descripción de expresiones faciales han sido usados para la representación de emociones. El sistema de codificación de acción facial o Facial Action Coding System (FACS) [16], es usado para la representación de emociones en términos de expresiones básicas y discretas, enfocándose en regiones musculares alrededor de los ojos, cejas, nariz, boca y demás músculos en el rostro. Esta codificación es llamada unidades de acción (AUs) y se puede apreciar 4 ejemplos de su uso en emociones en la Figura 2.1.

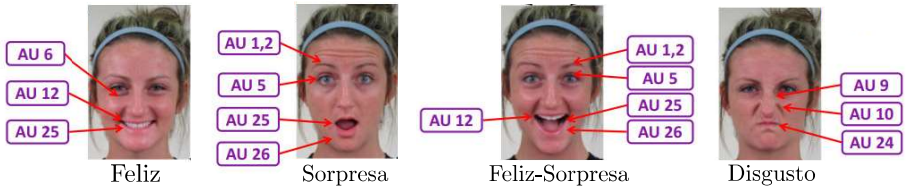


Figura 2.1. Unidades de acción del rostro aparecen al expresar una emoción o una combinación de ellas. Tomado de [35].

El sistema define un total de 32 AUs: 9 AUs se encuentran en la sección superior del rostro, 18 AUs en la sección inferior del rostro y 5 AUs que no pueden ser atribuidas de manera exclusiva a ninguna de las secciones. Gracias a las FACS, todas las posibles expresiones faciales pueden ser descritas como la combinación de diferentes AUs. La tabla Tabla 2.1 muestra una recopilación de las relaciones entre las AUs y diferentes expresiones faciales.

Tabla 2.1. Lista de AUs involucradas en algunas expresiones faciales

	AUs
FACS:	Rostro superior: 1, 2, 4-7, 43, 45, 46;
	Rostro inferior: 9-18, 20, 22-28;
	Otros: 21, 31, 38, 39.
Ira:	4, 5, 7, 10, 17, 22-26
Disgusto:	9, 10, 16, 17, 25, 26
Miedo:	1, 2, 4, 5, 20, 25, 26, 27.
Felicidad:	6, 12, 25
Tristeza:	1, 4, 6, 11, 15, 17
Sorpresa:	1, 2, 5, 26, 27

Adicionalmente, hay que comentar que la intensidad de la AU son medidas en una escala ordinal, A-B-C-D-E [30]. La interpretación de estos 5 niveles esta descripta como:

1. Nivel A: Se evidencia a un pequeño rastro de la AU.
2. Nivel B: Pruebas leves de la AU.
3. Nivel C: Pruebas marcadas o pronunciadas.
4. Nivel D: Evidencia de AU extrema.
5. Nivel E: Evidencia máxima.

2.2. Redes neuronales convolucionales

Las redes neuronales convolucionales (Convolutional Neural Network, CNN) actualmente son el estado del arte en sistemas de clasificación de imágenes. Una CNN trata de imitar los estímulos que se reciben en las células

de la corteza visual y provienen del campo visual [36]. Matemáticamente, la convolución unidimensional en tiempo discreto es definida como:

$$y[n] = (x * h)[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] \tag{2.1}$$

Donde x es una señal, h es el filtro o kernel usado en el sistema y y es la señal de salida. Las imágenes digitales se pueden considerar como una señal discreta bi-dimensional. Para estos tipos de señales la convolución es formulada como:

$$S[i,j] = (I * K)[i,j] = \sum_{m}\sum_{n} I[m,n]K[i-m, j-n] \tag{2.2}$$

Donde I es la imagen, K es el kernel que se esta usando y S es el resultado del mapeo no-lineal de la imagen. En la implementación de la capa de convolución, el kernel es más pequeño que la imagen y es aplicado deslizándolo a través de la imagen y calculando su respuesta. En algunos casos la imagen I puede tener múltiples canales de entrada, como el usual del formato RGB, en esos casos cada kernel es usado en todos los canales y el resultado es el promedio de las convoluciones por cada canal para ese kernel. Un ejemplo de este proceso se muestra en la Figura 2.2, la cual muestra el proceso de usar un kernel para la detección de bordes en un rostro.

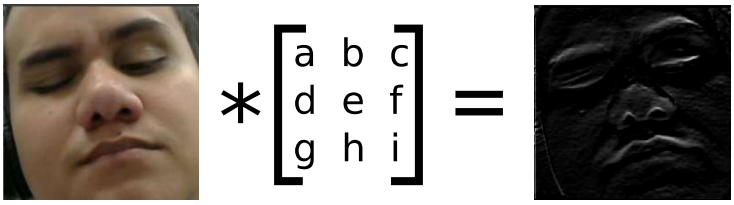


Figura 2.2. Proceso simple de filtrado para la detección de bordes.

Luego de realizar las operaciones en la capa de convolución, sigue la capa de diezmado, cuyo propósito es minimizar la alta dimensionalidad del conjunto mapeado de imágenes, este proceso consiste en aplicar un kernel y sobre este aplicar una operación para determinar el máximo, el promedio o la suma de los elementos contenidos dentro del kernel. En la Figura 2.3 se puede observar el proceso de diezmado de un conjunto de pixeles de 4x4 a un conjunto de pixeles de 2x2 usando la operación Max.

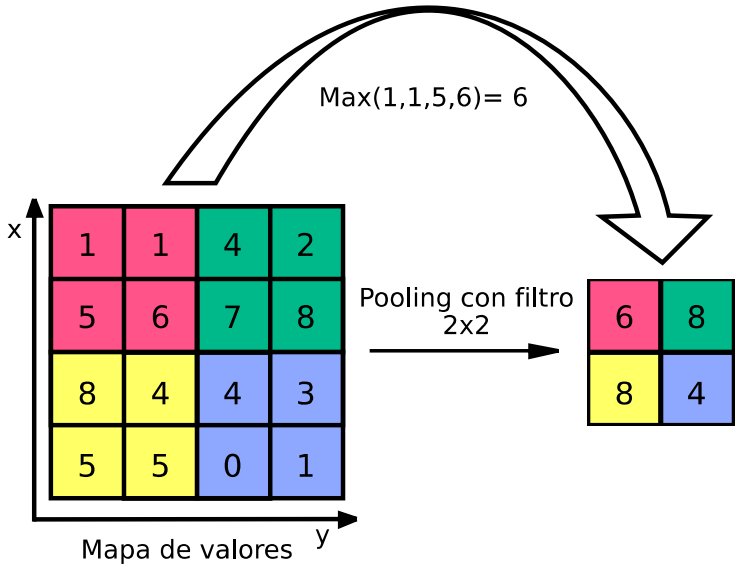


Figura 2.3. Proceso de diezmado utilizando un filtro Max de 2x2 a un conjunto de píxeles.

En la Figura 2.4 se puede observar la arquitectura de filtrado y diezmado en una capa para imagen de un rostro. El proceso de filtrado y diezmado de las imágenes se suele repetir varias veces con base en el número de capas en la CNN. Finalmente, se adiciona una capa totalmente conectada y una capa de salida para tomar la decisión final de clasificación [37].

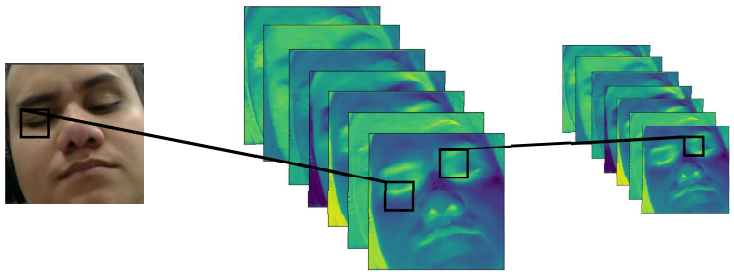


Figura 2.4. Procesos de filtrado y diezmado en las redes neuronales convolucionales.

Con base en este proceso, las CNN aprenden a diferenciar características de una imagen en cada una de sus capas ocultas. Cada capa oculta aumenta la complejidad de las características, aprendiendo e identificando características relevantes de la imagen de entrada para la clasificación, eliminando la nece-

sidad de una etapa previa de extracción de características como se realiza en el aprendizaje de máquina clásico.

2.3. Redes neuronales residuales

Las redes neuronales residuales (Residual neural network, ResNet) fueron desarrolladas por investigaciones de Microsoft en [38]. La profundidad de las redes neuronales convolucionales tiene una influencia decisiva en su aprendizaje, pero ajustar este parámetro de manera óptima es una tarea ardua. En teoría, cuando el número de capas de una red aumenta, su rendimiento también debería mejorar. Sin embargo, en la práctica la optimización de los hiper-parámetros de la red se vuelve más compleja y además, ocurre desvanecimiento del gradiente complicando el entrenamiento de neuronas profundas en la red.

Las ResNet tratan de aumentar la profundidad de la red evitando los problemas antes descritos. La idea central de las redes residuales se basa en la introducción de una función de identidad entre capas. En las redes neuronales convolucionales, hay una función no lineal de mapeo $y = F(x)$ entre las capas. En las redes neuronales residuales, se tiene una función no-lineal $y = F(x) + x$. La Figura 2.5 se muestra dos tipos de bloques residuales que han usado arquitecturas ResNet, ambos realizan operaciones de identidad donde las dimensiones de entrada y salida no cambian sus dimensiones.

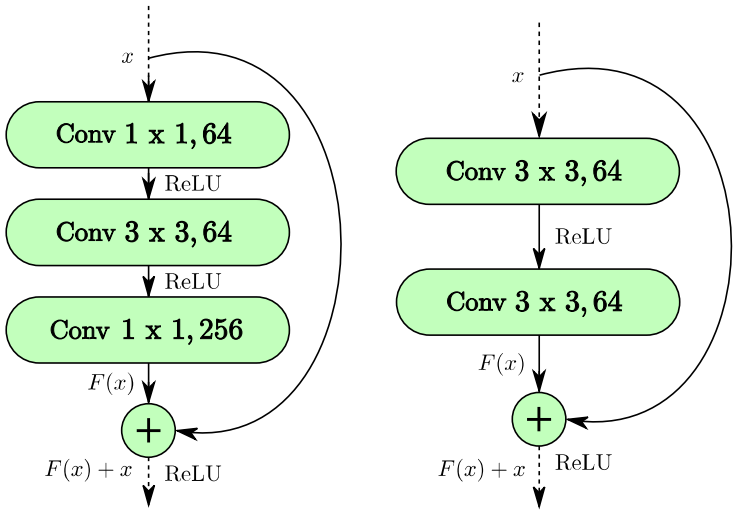


Figura 2.5. Diagrama de bloques residuales de identidad.

2.4. Función de triple pérdida

La métrica de distancia que da el origen a la función de triple perdida fue introducida como transformación lineal de los datos en [39]. Donde $S = \{(\mathbf{x}_i, y_i)\}$ denota los datos de entrenamiento con entradas $\mathbf{x}_i \in \mathbb{R}^d$ y con etiquetas de clase discretas $y_i \in \mathbb{Z}$, con $i = 1, 2, ..., N$. Donde N es el numero total de elementos en la base de datos. El objetivo es encontrar una transformación a los datos de entrada que reduzca la distancia entre pares de clases iguales y aumente la distancia de los pares con clases distintas. La métrica de distancia es definida a continuación como una distancia de Mahalanobis:

$$d_M^2(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{M} (\mathbf{x}_i - \mathbf{x}_j) \tag{2.3}$$

Donde $\mathbf{M} \in \mathbb{R}^{dxd}$ es una matriz simétrica semi definida positiva. Dado que la matriz $\mathbf{M}$ puede ser descompuesta como $\mathbf{M} = \mathbf{L}^T\mathbf{L}$, con $\mathbf{L}$ denotando una matriz de transformación lineal, la ecuación 2.3 puede ser reescrita como:

$$d_M^2(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{L}\,(\mathbf{x}_i - \mathbf{x}_j)\|_2^2 = \|\mathbf{x}_i{}' - \mathbf{x}_j{}'\|_2^2 \tag{2.4}$$

Inspirados en esto, la métrica de aprendizaje profundo usada en las redes neuronales profundas para aprender un vector de características $\mathbf{x}' = \Phi(\mathbf{x})$, el cual generaliza la transformación lineal $\mathbf{x}' = \mathbf{L}\mathbf{x}$ a un transformación no-lineal $\Phi(\mathbf{x})$. Reescribiendo la métrica de distancia:

$$d_M^2(\mathbf{x}_i, \mathbf{x}_j) = \|\Phi(\mathbf{x}_i) - \Phi(\mathbf{x}_j)\|_2^2 \tag{2.5}$$

El aprendizaje para determinar el vector de características $\Phi(\mathbf{x})$, es encontrar una transformación que haga más pequeña la distancia intraclase que la distancia interclase. En el entrenamiento se crea un set de datos basados en tripletas $\mathcal{S}_\mathrm{T}$ denotado como:

$$\mathcal{S}_\mathrm{T} = \{(\mathbf{x}^a, y^a), (\mathbf{x}^n, y^n), (\mathbf{x}^p, y^p) | \quad y^a = y^p, y^a \neq y^n\} \tag{2.6}$$

Donde a y p hace referencia a los datos que poseen etiquetas iguales y n hace referencia a los datos que poseen una etiqueta diferente a. La función de triple pérdida a minimizar es definida como:

$$\mathcal{L} = \sum_{\mathcal{S}_\mathrm{T}} [d_M^2(\mathbf{x}^a, \mathbf{x}^p) - d_M^2(\mathbf{x}^a, \mathbf{x}^n) + \alpha]_+ \tag{2.7}$$

Donde $[z]_+ = \max(z, 0)$ y $\alpha \geq 0$ es el margen mínimo que debe de existir entre clases. La Figura 2.6 muestra la distribución de la clase azul y roja en el espacio de características, antes y después de utilizar la función de triple perdidas para encontrar una representación de los datos más apropiada, creando agrupamientos en los que se encuentren datos con características similares. Además, se puede observar la distancia entre dos pares con clase similar en gris oscuro y el margen en un gris más claro.

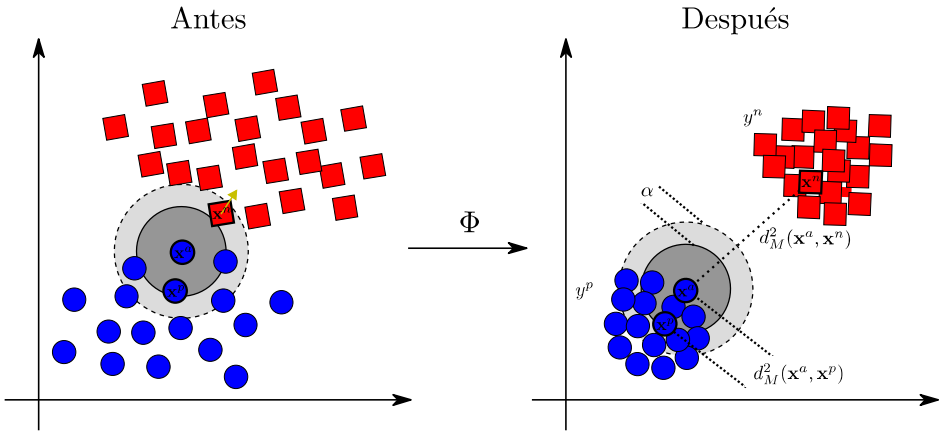


Figura 2.6. Distribución de los datos en el espacio de características antes (izquierda) y después (derecha) aplicando la transformación no-lineal encontrada con la función de triple pérdida.

A diferencia de otras funciones de perdidas como el uso de la entropía cruzada (Cross Entropy), o el Error Cuadrático Medio (Mean Square Error), cuyo objetivo es aprender la perdición de etiquetas directamente (Clasificación) o un conjunto de valores (Regresión); el objetivo de la función de triple perdida es predecir y mejorar la distancia relativa entre los datos de entradas $\mathbf{x}^a$, $\mathbf{x}^n$, $\mathbf{x}^p$.

El resultado de la función de triple pérdida puede dividir el tipo de tripleta en tres categorías:

1. **Tripletas fáciles**: Estas tripletas tienen asociado un $\mathcal{L} = 0$.
2. **Tripletas semi-duras**: Estas tripletas tienen asociado un $\alpha > \mathcal{L} > 0$.
3. **Tripletas duras**: Estas tripletas tienen asociado un $\mathcal{L} \geq \alpha$.

Cada una de estas definiciones depende del valor $d_M^2(\mathbf{x}^a, \mathbf{x}^p)$. En la Figura 2.7 se muestra las tres zonas donde puede existir $\mathbf{x}^n$, donde en la zona

verde se representa una tripleta fácil, la zona amarilla representa una tripleta semi-dura y la zona roja representa una tripleta dura.

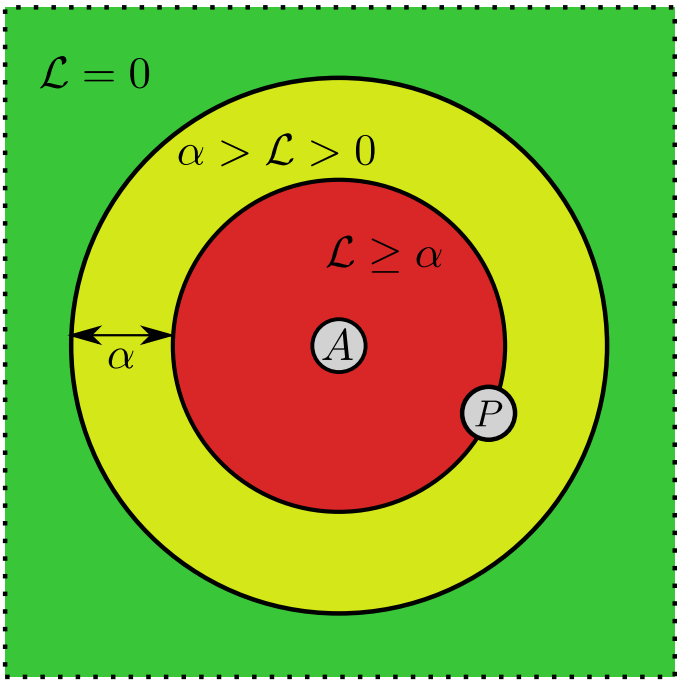


Figura 2.7. Representación de las tres categorías en el espacio de características.

2.5. Métodos de clasificación y de regresión

2.5.1. Máquina de soporte vectorial - Margen Duro

Las máquinas de soporte vectorial (Suppor Vector Machine, SVM) constituyen un método de clasificación lineal para datos que sean linealmente separables, donde a este conjunto de datos se puede encontrar un óptimo hiper plano de separación. Los vectores de características $\mathbf{x}_i \in \mathbb{R}^d$, tienen etiquetas $y_i \in \{+1, -1\}$, con $i = 1, 2, ..., N$. Donde N es el numero total de elementos en la base de datos.

Asumiendo que los datos sean linealmente separables, existen múltiples hiper-planos que dividen los datos, pero sólo uno que los divide con un margen separación más amplio. El hiper-plano de separación se define como:

$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b \tag{2.8}$$

Donde $\mathbf{w}$ es el vector normal del hiper-plano y b es el sesgo. Adicionalmente, si los datos son linealmente separables, todos los datos cumplen con las siguientes restricciones.

$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b \geq +1, \quad \text{si } y_i = +1 \tag{2.9}$$

$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b \leq -1, \quad \text{si } y_i = -1 \tag{2.10}$$

Las dos anteriores restricciones pueden ser escritas en la siguiente inecuación:

$$-y_i \left(\langle \mathbf{w}, \mathbf{x}_i \rangle + b \right) + 1 \leq 0 \tag{2.11}$$

Un dato a resaltar es que donde los vectores de características $\mathbf{x}_i$ cumplen la siguiente ecuación $y_i \left(\langle \mathbf{w}, \mathbf{x}_i \rangle + b \right) - 1 = 0$, esos vectores son denominados *vectores de soporte*. La distancia entre los *vectores de soporte* y el hiper-plano es $\frac{1}{\|\mathbf{w}\|}$, por lo tanto el margen del hiper-plano de separación es $\frac{2}{\|\mathbf{w}\|}$. El objetivo de la máquina de soporte vectorial es encontrar el hiper-plano óptimo con el que el margen de separación $\frac{2}{\|\mathbf{w}\|}$ es máximo o también puede ser escrito como la minimización de la siguiente función objetivo $\frac{1}{2} \|\mathbf{w}\|^2$. Bajo estos detalles se plantea el siguiente problema de optimización. En la Figura 2.8 se observa el caso de una SVM en dos dimensiones.

$$\underset{\mathbf{w},b}{\text{minimizar}} \quad \frac{1}{2} \|\mathbf{w}\|^2 \tag{2.12}$$

$$\text{sujeto a:} \quad -y_i \left(\langle \mathbf{w}, \mathbf{x}_i \rangle + b \right) + 1 \leq 0 \quad \forall i = 1, ..., N \tag{2.13}$$

La forma de resolver el problema de optimización planteado, es utilizando los multiplicadores de Lagrange α_i, $i = 1, 2, ..., N$.

$$L_P \left(\mathbf{w}, b, \alpha_i \right) = \frac{1}{2} \|\mathbf{w}\|^2 + \sum_{i=1}^{N} \alpha_i \left[1 - y_i \left(\langle \mathbf{w}, \mathbf{x}_i \rangle + b \right) \right] \tag{2.14}$$

$L_P \left(\mathbf{w}, b, \alpha_i \right)$ es la función de Langrange del problema primal y $\alpha_i \geq 0$ con $i = 1, 2, ..., N$ son los multiplicadores de Lagrange relacionados a la restricción del problema primal. Las variables primales $(\mathbf{w}, b)$ se deben de desvanecer y esto se logra estimando las derivadas parciales de L_p con respecto a éstas:

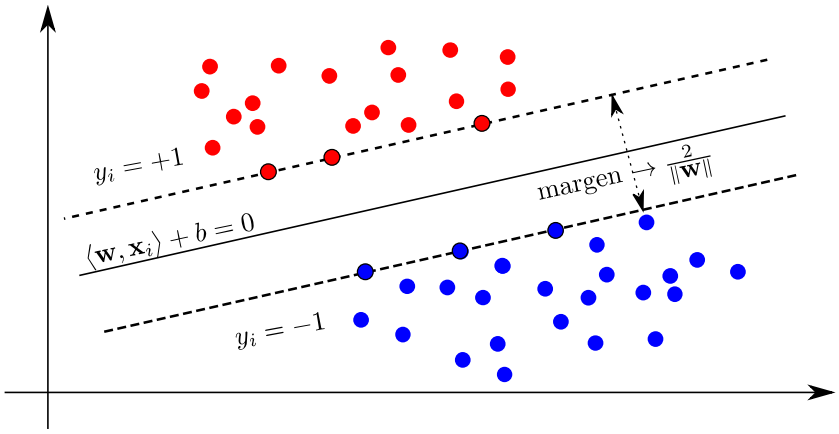


Figura 2.8. Máquinas de Soporte Vectorial - Margen Duro.

$$\frac{\partial L_p}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i = 0 \tag{2.15}$$

$$\frac{\partial L_p}{\partial b} = \sum_{i=1}^{N} \alpha_i y_i = 0 \tag{2.16}$$

Tomando en cuenta que las restricciones del problema son afines y apoyándonos en el teorema de Slater [40], el gap de dualidad es cero. Por lo tanto se puede formular el problema dual L_D, y los parámetros óptimos de L_p también lo son para L_D. Adicionalmente, las condiciones de Karush-Kuhn-Tucker (KKT) [41], [42] proveen las condiciones necesarias y de suficiencia para que un punto $\mathbf{x}^*$ sea óptimo. Estas condiciones se exponen a continuación:

1. Restricciones del problema primal

$$y_i \left(\langle \mathbf{w}, \mathbf{x}_i \rangle + b \right) - 1 \geq 0 \quad i = 1, 2, ..., N \tag{2.17}$$

2. Condición complementaria

$$\alpha_i \left[1 - y_i \left(\langle \mathbf{w}, \mathbf{x}_i \rangle + b \right) \right] = 0 \quad i = 1, 2, ..., N \tag{2.18}$$

3. Restricciones del problema dual

$$\alpha_i \geq 0 \quad i = 1, 2, ..., N \tag{2.19}$$

4. El gradiente de la función de Lagrange es cero $\nabla L_p = 0$

$$\frac{\partial L_p}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i \tag{2.20}$$

$$\frac{\partial L_p}{\partial b} = 0 \Rightarrow \sum_{i=1}^{N} \alpha_i y_i = 0 \tag{2.21}$$

Sustituyendo las condiciones en el problema primal, la función dual es encontrada como:

$$L_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle \tag{2.22}$$

Al revisar con detalle la condición complementaria $\alpha_i \left[1 - y_i \left(\langle \mathbf{w}, \mathbf{x}_i \rangle + b\right)\right] = 0$, cuando los multiplicadores de Lagrange $\alpha_i > 0$, entonces $1 - y_i \left(\langle \mathbf{w}, \mathbf{x}_i \rangle + b\right) = 0$.

Con lo que se concluye que todos los vectores $\mathbf{x}_i$ que tienen asociado un $\alpha_i > 0$ son *vectores de soporte* del hiper-plano.

2.6. Máquina de soporte vectorial - Margen Suave

En la sección anterior hemos definido las máquinas de soporte vectorial en un espacio de representación de datos idealizado, en el cual los datos son linealmente separables, pero esto no es cierto, en casos reales ser requieren funciones no lineales para lograr separar las clases. Dado que los conjuntos de datos no son perfectamente separable, se introduce un costo a la función objetivo del problema primal para la penalización de los errores. Esto es realizado agregando la variable no negativa de holgura ξ_i con $i = 1, 2, ..., N$ en las restricciones del problema de optimización, dando lugar a las siguientes restricciones:

$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b \geq +1 - \xi_i, \quad \text{si } y_i = +1 \tag{2.23}$$

$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b \leq -1 + \xi_i, \quad \text{si } y_i = -1 \tag{2.24}$$

$$\xi_i \geq 0, \quad i = 1, 2, ..., N \tag{2.25}$$

Nuevamente la restricciones 2.23 y 2.24 pueden ser reescritas en la siguiente inecuación:

$$-y_i\left(\langle \mathbf{w}, \mathbf{x}_i\rangle + b\right) + 1 - \xi_i \leq 0 \tag{2.26}$$

Los errores de la máquina de soporte vectorial existen cuando $\xi_i > 1$, lo cual podemos introducir los errores de entrenamiento de la máquina como $\sum_{i=1}^{N} \xi_i$ y agregarla en la función objetivo como una modificación al problema de optimización de los casos linealmente separables, definiendo el nuevo problema de optimización de la siguiente forma:

$$\underset{\mathbf{w},b,\xi_i}{\text{minimizar}} \quad \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i \tag{2.27}$$

$$\text{sujeto a:} \quad -y_i\left(\langle \mathbf{w}, \mathbf{x}_i\rangle + b\right) + 1 - \xi_i \leq 0 \tag{2.28}$$

$$-\xi_i \leq 0 \quad \forall i = 1, ..., N \tag{2.29}$$

Donde C en la ecuación 2.27 es un hiper-parámetro qué define qué tan alta será la penalización de los errores. La nueva función de Lagrange queda de la siguiente forma:

$$\begin{aligned} L_P\left(\mathbf{w}, b, \xi_i, \alpha_i, \mu_i\right) =& \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{i=1}^{N}\alpha_i\left[1 - \xi_i - y_i\left(\langle \mathbf{w}, \mathbf{x}_i\rangle + b\right)\right] \\ &+ C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\mu_i\xi_i \end{aligned} \tag{2.30}$$

Dada la nueva restricción, se agregó un nuevo multiplicador de Lagrange μ_i, el cual también es no negativo ($\mu_i \geq 0$). Las condiciones de KKT para este problema son las siguientes:

1. Restricciones del problema primal

$$1 - \xi_i - y_i\left(\langle \mathbf{w}, \mathbf{x}_i\rangle + b\right) \leq 0 \tag{2.31}$$

$$\xi_i \geq 0 \tag{2.32}$$

2. Condición complementaria

$$\alpha_i\left[1 - \xi_i - y_i\left(\langle \mathbf{w}, \mathbf{x}_i\rangle + b\right)\right] = 0 \tag{2.33}$$

$$\mu_i\xi_i = 0 \tag{2.34}$$

3. Restricciones del problema dual

$$\alpha_i \geq 0 \quad i = 1, 2, ..., N \tag{2.35}$$

$$\mu_i \geq 0 \quad i = 1, 2, ..., N \tag{2.36}$$

4. El gradiente de la función de Lagrange es cero $\nabla L_p = 0$

$$\frac{\partial L_p}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i \tag{2.37}$$

$$\frac{\partial L_p}{\partial b} = 0 \Rightarrow \sum_{i=1}^{N} \alpha_i y_i = 0 \tag{2.38}$$

$$\frac{\partial L_p}{\partial \xi_i} = 0 \Rightarrow C - \alpha_i - \mu_i = 0 \tag{2.39}$$

Reemplazando las nuevas condiciones en el problema primal, encontraremos la función dual que se describe a continuación:

$$L_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle \tag{2.40}$$

$$\text{sujeto a:} \quad 0 \leq \alpha_i \leq C \tag{2.41}$$

$$\sum_{i=1}^{N} \alpha_i y_i = 0 \tag{2.42}$$

Revisando la condición complementaria 2.33, se observa que los vectores $\mathbf{x}_i$ que cumplen el caso de $0 < \alpha_i \leq C$, son utilizados como *vectores de soporte* para la creación del hiper-plano de separación. De igual forma se observa que si $\alpha_i = C$, se aceptarán como *vectores de soporte* a ciertos vectores cuyo $\xi_i > 0$. El caso de esta máquina de soporte vectorial se puede ver en la Figura 2.9.

Los dos casos anteriores resumen el uso de las máquinas de soporte vectorial en la creación de hiper-planos lineales para la separación de los datos. Esto es sólo un tipo de caso, ya que existen hiper-planos de separación no lineales, para resolver este problema y para la creación de otros hiper-planos de separación, se introduce el concepto de *funciones kernel*. Inicialmente se

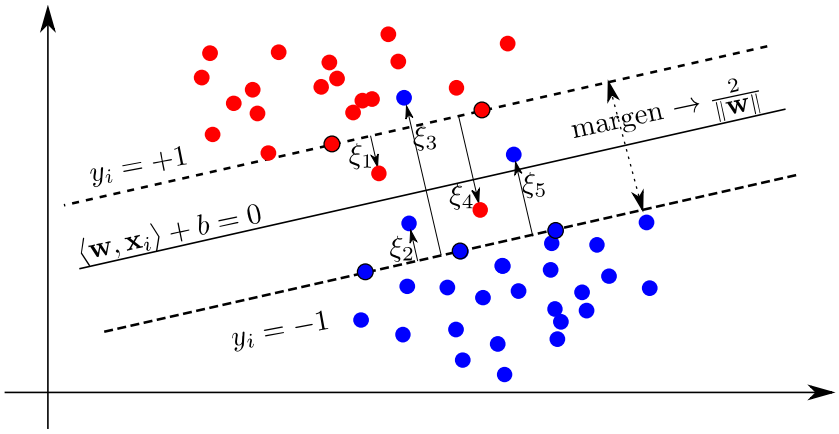


Figura 2.9. Maquinas de Soporte Vectorial - Margen Suave.

tiene un espacio vectorial $\mathbb{R}^d$ donde no se puede describir un hiper-plano de separación lineal, estos datos deberán ser mapeados a otro espacio vectorial $\mathbb{R}^m$, donde m puede ser incluso mayor que d y en este nuevo espacio vectorial se posible definir un hiper-plano de separación lineal. La función de mapeo se definida como:

$$\Phi : \mathbb{R}^d \rightarrow \mathbb{R}^m \tag{2.43}$$

En el planteamiento del problema de optimización de la máquina de soporte vectorial se usaron productos punto, lo cuales pueden ser reemplazados usando las funciones de mapeo a $\mathbb{R}^m$, denotando el producto interno como $\langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle$. Asumiendo la existencia de la denominada función kernel $K(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle$, sólo es necesario definir la función K para formular y resolver el nuevo problema de optimización. Los kernels más utilizados son el polinómico y el Gaussiano, los cuales se describen a continuación:

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\langle \mathbf{x}_i, \mathbf{x}_j \rangle + 1)^p \tag{2.44}$$

$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\frac{1}{2\sigma^2}\|\mathbf{x}_i - \mathbf{x}_j\|^2} \tag{2.45}$$

Donde p es el grado del polinomio y σ el ancho del kernel Gaussiano. En la Figura 2.10 se observa el uso de la función Kernel.

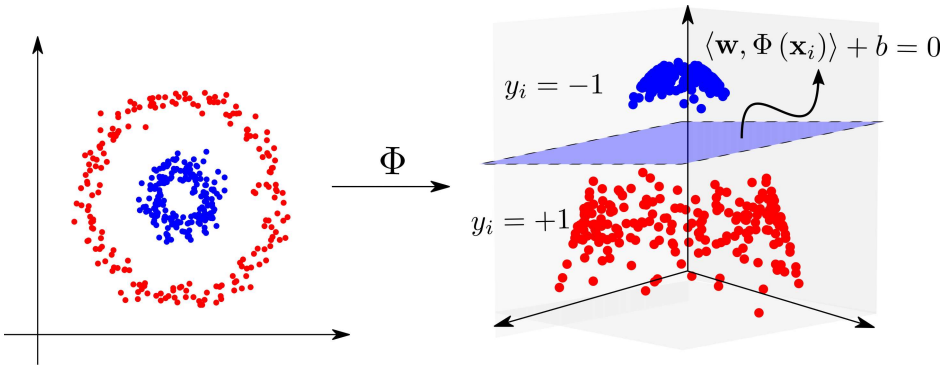


Figura 2.10. Máquinas de Soporte Vectorial con el uso de la función Kernel.

2.6.1. Regresión por vectores de soporte

Una formulación similar puede ser utilizada para resolver problemas de regresión. La regresión de soporte vectorial (Support Vector Regression, SVR), es construida para los vectores de características $\mathbf{x}_i \in \mathbb{R}^d$, las etiquetas $y_i \in \mathbb{R}$. Una formulación similar a la SVM - Margen Suave es usada para la construcción del regresor usando la función de costos ε-insensitive introducida por Vapnik en [43]. Para el conjunto de datos $(\mathbf{x}_i, y_i)$, el valor objetivo o valor real para la regresión son los $y_i \in \mathbb{R}$, y para todos $\mathbf{x}_i$ su valor de predicho es $f(\mathbf{x})$, el error de predicción en la regresión es determinado con $|y_i - f(\mathbf{x})|$. Como la predicción de puede estar por encima o por debajo del valor real, estas dos posibilidades son consideradas usando una variable que describe el margen de tal manera que $y_i - f(\mathbf{x}_i) > \varepsilon$ y $f(\mathbf{x}_i) - y_i > \varepsilon$.

La función lineal para la regresión puede ser estimada como $f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b$, el problema de optimización puede ser formulado como:

$$\text{minimizar} \quad \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^{N} |y_i - f(\mathbf{x}_i)|_\varepsilon \tag{2.46}$$

Donde C es el parámetro de penalización.

Note que la expresión 2.46 puede ser transformada a un problema de optimización con restricciones, introduciendo las variables de holgura ξ_i. Asignando el valor de ξ_i cuando $f(\mathbf{x}_i) - y_i > \varepsilon$ y ξ_i^* cuando $y_i - f(\mathbf{x}_i) > \varepsilon$, entonces podemos expresar la función objetivo del problema primal de la ε-SVR como:

$$\text{minimizar} \quad \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) \tag{2.47}$$

$$\text{sujeto a:} \quad (\langle \mathbf{w}, \mathbf{x}_i\rangle + b) - y_i \leq \varepsilon + \xi_i \tag{2.48}$$

$$y_i - (\langle \mathbf{w}, \mathbf{x}_i\rangle + b) \leq \varepsilon + \xi_i^* \tag{2.49}$$

$$\xi_i, \xi_i^* \geq 0 \quad \forall i = 1, ..., N \tag{2.50}$$

La Figura 2.11 muestra el caso de una regresión de vectores de soporte, en conjunto con la definición del tubo de ε-insensitive.

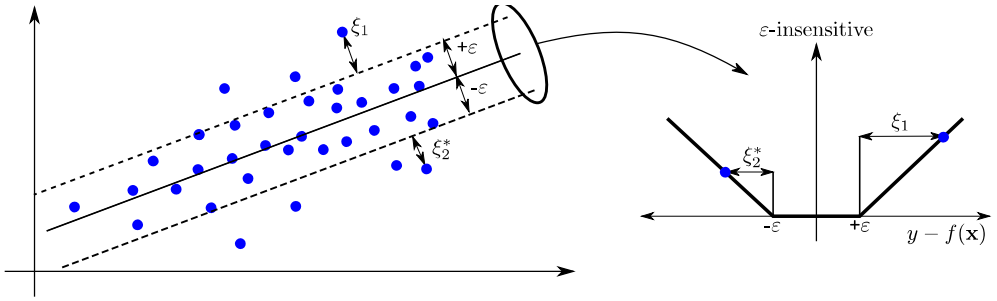


Figura 2.11. Regresión de Vectores de Soporte.

La función de Langrange del problema primal esta definida como:

$$L_P = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) - \sum_{i=1}^{N}(\eta_i\xi_i + \eta_i^*\xi_i^*) \tag{2.51}$$
$$- \sum_{i=1}^{N}\alpha_i\left[\varepsilon + \xi_i + y_i - (\langle \mathbf{w}, \mathbf{x}_i\rangle + b)\right] - \sum_{i=1}^{N}\alpha_i^*\left[\varepsilon + \xi_i^* - y_i + (\langle \mathbf{w}, \mathbf{x}_i\rangle + b)\right]$$

Donde $\{\eta_i, \eta_i^*, \alpha_i, \alpha_i^*\} \geq 0$. Las variables primales w, b, ξ_i, ξ_i^*, se tienen que desvanecer para encontrar la solución optima.

$$\frac{\partial L_p}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{N}(\alpha_i^* - \alpha_i)\,\mathbf{x}_i = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^{N}(\alpha_i^* - \alpha_i)\,\mathbf{x}_i \tag{2.52}$$

$$\frac{\partial L_p}{\partial b} = 0 \Rightarrow \sum_{i=1}^{N}(\alpha_i^* - \alpha_i) = 0 \tag{2.53}$$

$$\frac{\partial L_p}{\partial \xi_i} = C - \alpha_i - \eta_i = 0 \Rightarrow C = \alpha_i + \eta_i \tag{2.54}$$

$$\frac{\partial L_p}{\partial \xi_i^*} = C - \alpha_i^* - \eta_i^* = 0 \Rightarrow C = \alpha_i^* + \eta_i^* \tag{2.55}$$

Con estos resultados procedemos a reemplazar en el problema primal para obtener la función dual que se describe a continuación:

$$L_D = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\alpha_i^* - \alpha_i)(\alpha_i^* - \alpha_i)\left\langle \mathbf{x}_i, \mathbf{x}_j \right\rangle - \varepsilon\sum_{i=1}^{N}(\alpha_i^* + \alpha_i) + \sum_{i=1}^{N} y_i(\alpha_i^* - \alpha_i) \tag{2.56}$$

$$\text{sujeto a:} \quad \sum_{i=1}^{N}(\alpha_i^* - \alpha_i) = 0 \tag{2.57}$$

$$0 \leq \{\alpha_i, \alpha_i^*\} \leq C \tag{2.58}$$

$$\tag{2.59}$$

Dado que $\mathbf{w} = \sum_{i=1}^{N}(\alpha_i^* - \alpha_i)\,\mathbf{x}_i$, la función de regresión puede ser reescrita como:

$$f(x) = \sum_{i=1}^{N}(\alpha_i^* - \alpha_i)\left\langle \mathbf{x}_i, \mathbf{x} \right\rangle + b \tag{2.60}$$

Se puede mostrar que $\mathbf{w}$ es una combinación lineal de los puntos de entrenamiento $\mathbf{x}_i$. Adicionalmente, se evidencia que el resultado tiene la inclusión de productos puntos de los datos, el cual puede ser generalizado a una regresión basado por kernels. Ahora queda verificar las condiciones de KKT del problema de optimización.

1. Restricciones del problema primal

$$\varepsilon + \xi_i - (\langle \mathbf{w}, \mathbf{x}_i \rangle + b) + y_i \geq 0 \tag{2.61}$$

$$\varepsilon + \xi_i^* - y_i + (\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq 0 \tag{2.62}$$

$$\{\xi_i, \xi_i^*\} \geq 0 \tag{2.63}$$

2. Condición complementaria

$$\alpha_i \left(\varepsilon + \xi_i + y_i - \langle \mathbf{w}, \mathbf{x}_i \rangle + b\right) = 0 \tag{2.64}$$

$$\alpha_i^* \left(\varepsilon + \xi_i^* - y_i + \langle \mathbf{w}, \mathbf{x}_i \rangle + b\right) = 0 \tag{2.65}$$

$$\eta_i \xi_i = 0 \quad \eta_i^* \xi_i^* = 0 \tag{2.66}$$

Las cuales pueden ser reescritas como:

$$(C - \alpha_i)\xi_i = 0 \quad (C - \alpha_i^*)\xi_i^* = 0 \tag{2.67}$$

3. Restricciones del problema dual

$$\sum_{i=1}^{N} (\alpha_i^* - \alpha_i) = 0 \tag{2.68}$$

$$0 \leq \{\alpha_i, \alpha_i^*\} \leq C \tag{2.69}$$

4. El gradiente de la función de Lagrange es cero $\nabla L_p = 0$

$$\frac{\partial L_p}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^{N} (\alpha_i^* - \alpha_i)\, \mathbf{x}_i \tag{2.70}$$

$$\frac{\partial L_p}{\partial b} = 0 \Rightarrow \sum_{i=1}^{N} (\alpha_i^* - \alpha_i) = 0 \tag{2.71}$$

$$\frac{\partial L_p}{\partial \xi_i} = 0 \Rightarrow C = \alpha_i + \eta_i \tag{2.72}$$

$$\frac{\partial L_p}{\partial \xi_i^*} = 0 \Rightarrow C = \alpha_i^* + \eta_i^* \tag{2.73}$$

Revisando las condiciones complementarias, podemos observar que si $\xi_i > 0$ o $\xi_i^* > 0$, los puntos $(\mathbf{x}_i, y_i)$ corresponden a valores de $\alpha_i = C$ o $\alpha_i^* = C$ y son ubicados por fuera del tubo ε-insensitive alrededor de $f(\mathbf{x})$. Por otra parte, podemos si $0 < \alpha_i < 0$ o $0 < \alpha_i^* < 0$ y con la ayuda de la expresión 2.67, observamos que ξ_i y ξ_i^* deben ser cero. Reemplazando este hecho en las otras dos restricciones complementarias obtendremos como resultado:

$$\alpha_i \left(\varepsilon + y_i - \langle \mathbf{w}, \mathbf{x}_i \rangle + b\right) = 0 \quad \text{y} \quad \alpha_i^* \left(\varepsilon - y_i + \langle \mathbf{w}, \mathbf{x}_i \rangle + b\right) = 0 \tag{2.74}$$

Pero recordando que $\{\alpha_i, \alpha_i^*\} > 0$, podemos determinan que el lado derecho de las ecuaciones tiene que ser cero y gracias a esto hallar el sesgo b de la regresión $f(\mathbf{x})$.

$$b = y_i - \langle \mathbf{w}, \mathbf{x}_i \rangle + \varepsilon, \text{ con } 0 < \alpha_i < 0 \tag{2.75}$$

$$b = y_i - \langle \mathbf{w}, \mathbf{x}_i \rangle - \varepsilon, \text{ con } 0 < \alpha_i^* < 0 \tag{2.76}$$

Igualmente, cuando $0 < \alpha_i < 0$ o $0 < \alpha_i^* < 0$, para todos los puntos que están dentro del tubo del ε-insensitive, los multiplicadores de Lagrange α_i y α_i^* se desvanecen para satisfacer las condiciones complementarias. Observando que $\mathbf{w} = \sum_{i=1}^{N} (\alpha_i^* - \alpha_i) \mathbf{x}_i$, los datos $\mathbf{x}_i$ que tengan relacionado un α_i o α_i^* que no se desvanecen, serán tomados como *vectores de soporte*, ubicandolos en el exterior del tubo.

2.7. Estrategias de validación

2.7.1. Validación cruzada

Los resultados de los experimentos se ven influenciados por los datos elegidos para el entrenamiento y la prueba . Por lo tanto, es necesario aplicar estrategias de validación para obtener resultados independientes de los datos seleccionados, para obtener información de la capacidad de generalización en los resultados. La estrategia utilizada para este trabajo es la validación cruzada con k-divisiones (k-fold). En esta estrategia los datos son divididos en k bloques, cada una de estas divisiones es seleccionada para probar el modelo y el resto es utilizado para el entrenamiento. Este proceso se repite hasta recorrer todas las divisiones y los resultados en las pruebas para las k divisiones es promediada para tener un valor único e igualmente se calcula la desviación estándar a los k resultados obteniendo una medida de tanto se ven afectados los resultados por la selección de los datos. En la Figura 2.12 se observa el concepto de validación cruzada. Los parámetros óptimos del clasificador o regresor son calculados para cada fold de test, así que al final se tienen k conjuntos de parámetros óptimos. Con el fin de dar transparencia al proceso de entrenamiento y prueba, se reportan también estos conjuntos, sus valores promedio, desviación estándar y moda.

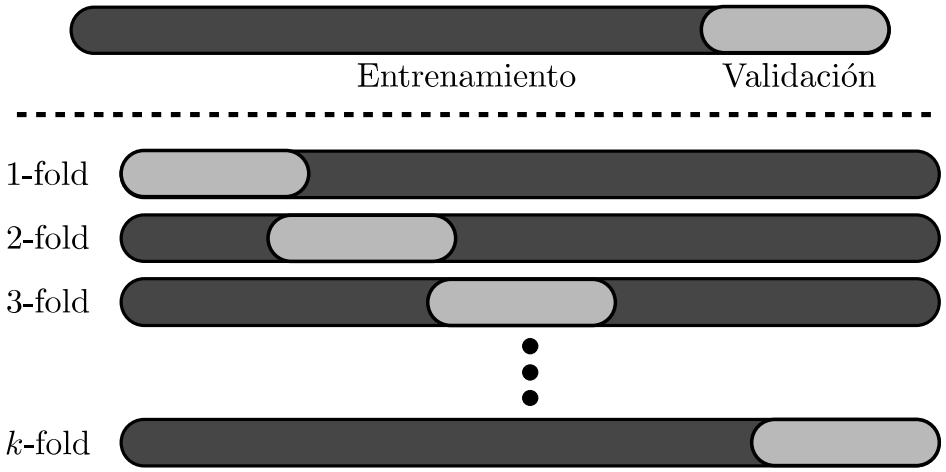


Figura 2.12. Estrategia de validación cruzada con k-folds.

Capítulo 3

Bases de Datos

3.1. VGGFace2

Esta base de datos esta compuesta por mas de 3.31 millones de rostros de 9131 usuarios diferentes. Un promedio de 362.6 imágenes por usuario están incluidos en VGGFace2 [44]. Las imágenes fueron descargadas con Google Image Search. El corpus tiene grandes variaciones en cuanto a la pose, la edad, la iluminación, la etnia y la profesión.

3.2. Base de datos EmotioNet

Esta base de datos fue originalmente introducida por investigadores de la Universidad Estatal de Ohio que realizaron el *Emotion Challenge* in 2017 [45]. Esta base de datos contiene un millón de imágenes de expresiones faciales recolectadas a través de Internet. Un total de 950,000 imágenes estan anotadas por un modelo de detección automática de AUs presentando en [45], y las 50,000 imágenes restantes fueron anotadas por expertos. Un total de 12 AUs son incluidas en este corpus.

3.3. Base de datos FacePark-GITA

La recolección de datos para este proyecto fue desarrollada a través de la plataforma FacePark. La plataforma FacePark es un servicio web con ayuda del Framework Django de Python y fue desarrollado por el grupo de investigación GITA de la Universidad de Antioquia, esta plataforma fue desarrollada para la ayuda a la terapia de paciente con Parkinson en campos como la voz,

movimientos motores y expresiones faciales. La plataforma fue desarrollada en un proyecto conjunto a la University of Cincinnati Academic Health Center, ubicado en Cincinnati, Ohio. La plataforma posee diferentes actividades para la captura del habla como frases cortas, vocales sostenidas, vocales moduladas, la repetición de palabras como /pataka,petaka,pakata/, un texto de 36 palabras y la captura de un monologo. En la Figura 3.1 se puede observar un ejemplo de captura de una frase corta en la plataforma FacePark-GITA[1].

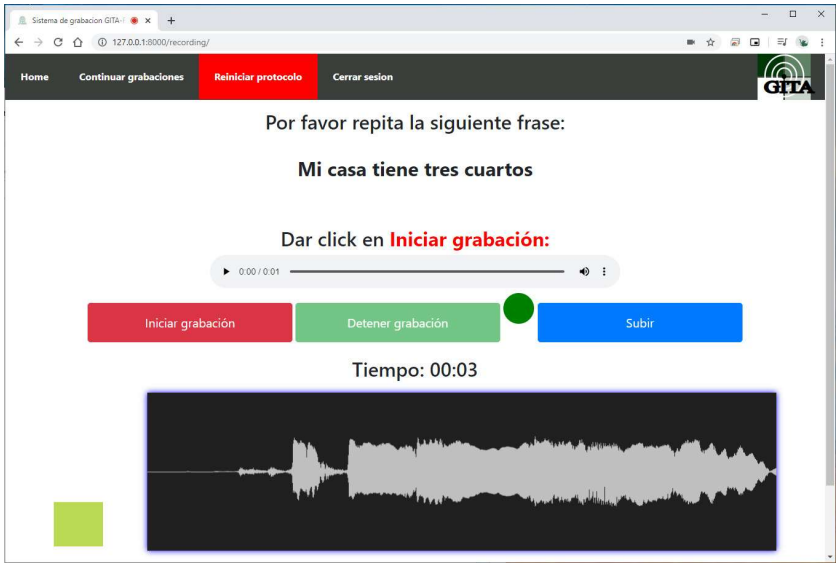


Figura 3.1. Proceso de captura de la frase "Mi casa tiene tres cuarto" con la ayuda de la plataforma FacePark-GITA.

La plataforma igualmente captura movimientos motores finos en video como los toques rápidos entre el dedo pulgar y dedo índice en cada mano como se puede observar en la Figura 3.2.

[1]https://gita.udea.edu.co:8080/

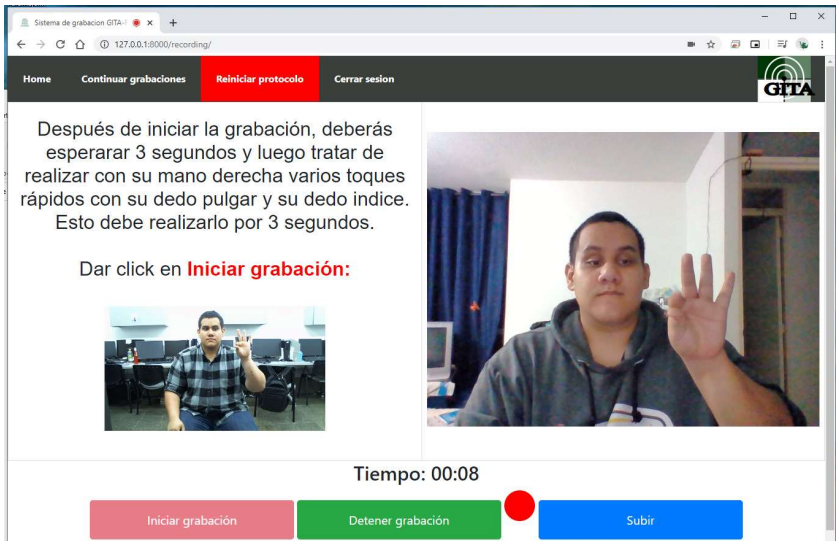


Figura 3.2. Proceso de captura del toque del dedo pulgar y el dedo índice con ayuda de la plataforma FacePark-GITA.

En la captura de expresiones faciales, muestra a los usuarios una serie de 8 actividades en forma de ejemplo para que realicen mientras son grabados. Las actividades grabadas son: realizar un guiño con el ojo derecho, realizar un guiño del ojo izquierdo, mostrar una cara sonriente, mostrar una expresión de ira, mostrar una expresión de sorpresa y realizar un serie movimientos con la cabeza. Además, se captura en video la lectura de un texto de 36 palabras y un monologo. La plataforma se puede observar en la Figura 3.3.

Figura 3.3. Proceso de captura de la expresión de enojo, con ayuda de la plataforma FacePark.

Todos los audios fueron grabados a 16 bits por muestra a una frecuencia de muestro de 48kHz y los videos fueron grabados a una velocidad de 30 FPS (Frames per second) y fueron grabados en diferentes condiciones de iluminación. La base de datos utilizada para este trabajo contiene un total de 30 pacientes con Parkinson y 24 participantes sanos. Los pacientes con Parkinson fueron diagnosticados y evaluados por neurólogos expertos según la escala MDS-UPDRS-III. En la Tabla 3.1 se muestra un resumen de la información clínica y demográfica de los participantes.

Tabla 3.1. Información de los participantes en la base de datos FacePark.

	Pacientes con Parkinson		**Participantes sanos**	
	Masculino	Femenino	Masculino	Femenino
Participantes	18	12	12	12
Edad ($\mu \pm \sigma$)	70,2 ± 10,4	67,4 ± 10,9	65,3 ± 8,7	65,2 ± 10,1
Rango de edades	52 - 90	53 - 87	49 - 83	49- 80
t ($\mu \pm \sigma$)	8,7 ± 5,7	14,3 ± 16,7	—	—
Rango de t	2 - 20	1 - 44	—	—
MDS-UPDRS-III ($\mu \pm \sigma$)	35,4 ± 13,9	29,7 ± 12,3	—	—
Rango MDS-UPDRS-III	16 - 65	15 - 54	—	—
H&Y ($\mu \pm \sigma$)	2,4 ± 0,5	2,5 ± 0,5	—	—
Rango H&Y	2 - 3	2 - 3	—	—

MDS-UPDRS: Movement Disorder Society - Unified Parkinson's Disease Rating Scale. H&Y: Escala Hoehn & Yahr. t: Años de diagnostico

Todos los participantes dieron su consentimiento informado por escrito. El estudio está de acuerdo con la Declaración de Helsinki y fue aprobado por el Comité de Ética de la Investigación de la Universidad de Antioquia.

Se pidió a los participantes de este estudio que produjeran diferentes expresiones faciales mientras eran grabados. Se incluyen un total de cinco grabaciones de video: guiño del ojo derecho, guiño del ojo izquierdo, sonrisa, expresión de ira y expresión de sorpresa. Los posibles sesgos introducidos por edad o género fueron descartados mediante una prueba estadística de chi-cuadrado ($p = 0{,}44$) y una prueba de Welch ($p = 0{,}15$), respectivamente.

Capítulo 4

Métodos

4.1. Reconocimiento de rostros para el modelamiento de la hipomimia

Este trabajo considera el modelo VGGFace2 como la base para el reconocimiento facial. El modelo se basa en la arquitectura ResNet50 [38], con 50 capas y 25,6 millones de parámetros. Fue propuesto originalmente para las tareas de reconocimiento de imágenes en general y luego fue entrenado con la base de datos VGGFace2 para obtener el reconocimiento facial. El modelo VGGFace2 se basa en la capacidad de aprender diferentes características de la imagen en cada capa. El modelo termina con un red completamente conectada. Un modelo pre-entrenado de VGGFace2 tiene capas con pesos ya entrenados según la información extraída de la base de datos VGGFace2. Los pesos de las capas pueden ser usados como extractores de características simplemente eliminando su capa de decisión y usando características para modelar otras tareas o para alimentar un clasificador.

La Figura 4.1 muestra un ejemplo del uso del modelo VGGFace2 como un extractor de características. El lado izquierdo se muestra el proceso de entrenamiento del modelo VGGFace2. El lado derecho se muestra el uso del modelo VGGFace2 como extractor de características, removiendo la ultima capa de decisión y usando estas nuevas salidas como entradas a el clasificador

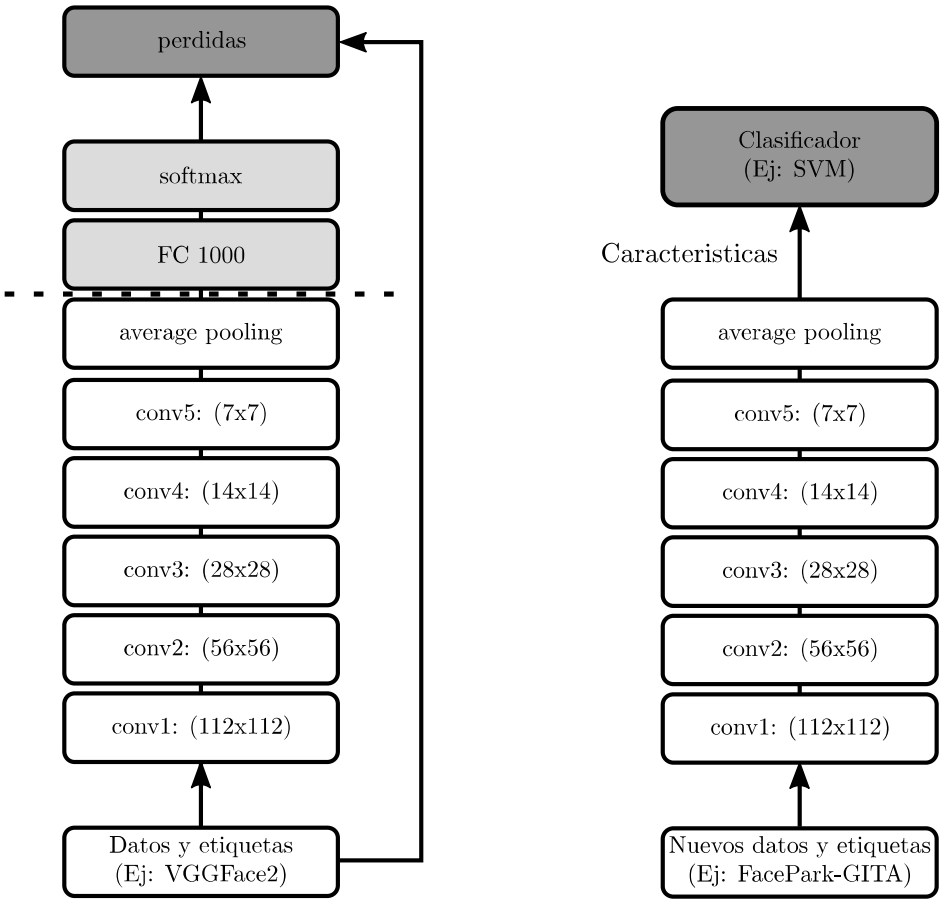


Figura 4.1. (izq.) Entrenamiento del modelo VGGFace2. (der.) VGGFace2 usado como extractor de características y usando dichas caracteristicas en un clasificador.

4.2. Transferencia de aprendizaje en CNN

Los transferencia de aprendizaje (Transfer Learning, TL) son métodos en los que los pesos de un modelo originalmente desarrollado para una tarea se utilizan para reentrenar otro modelo para abordar una nueva.

Una de las técnicas de TL permite realizar el congelamiento de capas intermedias las cuales conservarán la capacidad de extraer características y en re-entrenar capas superiores. El reentrenamiento de estas capas superiores permiten modificar el espacio de características original para mejorar la clasificación en nuevas tareas.

El método es adecuado para problemas en los que se dispone de pequeñas

bases de datos, lo que permite su uso en una amplia variedad de aplicaciones, incluido el modelado de la hipomimia en pacientes que sufren de EP, en los que no existen grandes conjuntos de datos.

La Figura 4.2 muestra la aplicación del TL en un modelo VGGFace2. El lado izquierdo muestra el entrenamiento del modelo VGGFace2. El lado derecho muestra el entrenamiento de un nuevo modelo, usando el modelo VGGFace2 como punto de partida. La Figura 4.2 también muestra la congelación de los primeros bloques residuales y la sustitución de las capas de conexión y decisión para crear el nuevo modelo.

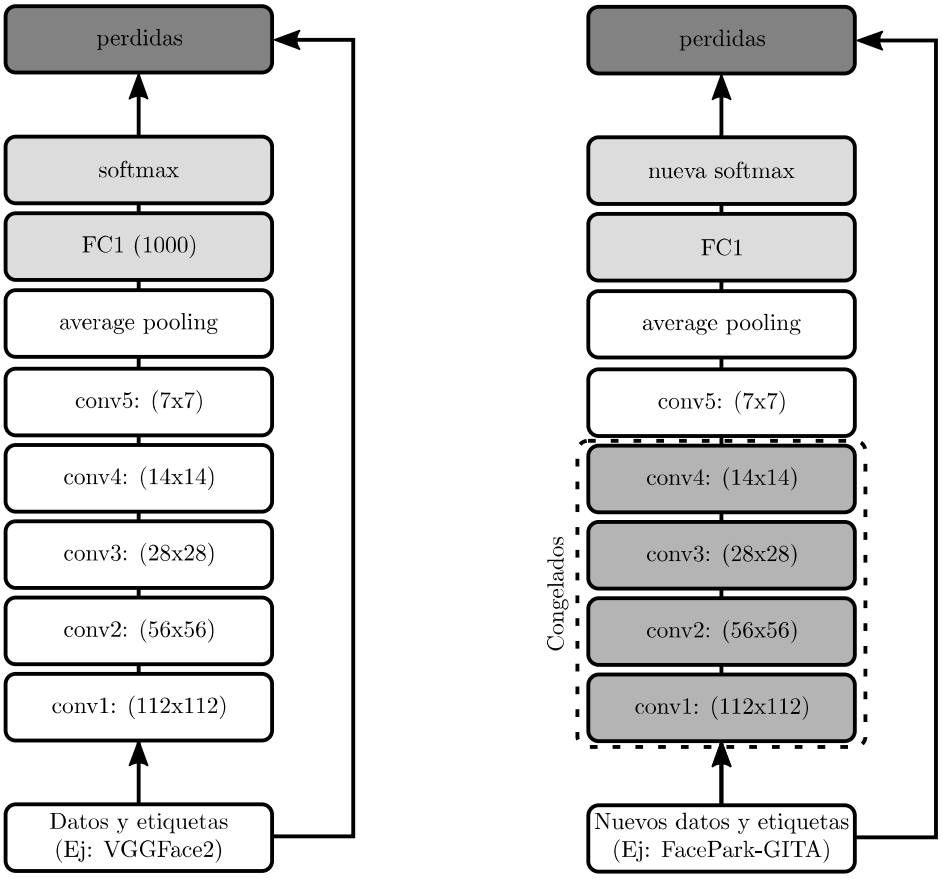


Figura 4.2. (izq.) Entrenamiento del modelo VGGFace2. (der) Modelo VGGFace2 entrenado con una nueva base de datos y realizando el congelamiento de las primeras bloques residuales.

4.3. Creando modelos desde cero para el reconocimiento de AUs

En general, los algoritmos de aprendizaje de maquina y de aprendizaje profundo están tradicionalmente diseñados para crear modelos desde cero y entrenarlos para resolver una tarea en especifico. En este trabajo, nosotros tenemos un interés en emplear 2 arquitecturas del estado del arte para el reconocimiento de imágenes basados en capas convolucionales y en capas residuales. Este enfoque ha mostrado resultados exitosos en diferentes temas, incluyendo los algoritmos de detección de rostros [46], [47].

VGG-8: Este modelo contiene 8 capas convolucionales dividido en grupos de 2 capas. Seguido de cada grupo, existen capas Max. Las capas convolucionales aplican una variedad de filtros a las imágenes y las capas de diezmado Max reducen el tamaño de las imágenes filtradas. Adicionalmente, son usadas las capas de regularización de Dropout, cuyo trabajo es apagar neuronas de forma aleatoria para evitar el sobre-entrenamiento del modelo. La ultima capa convolucional tiene 6 capas totalmente conectadas antes de la capa de decisión con 1024, 512, 256, 128, 64, y 32 neuronas por cada capa. El número total de parámetros en los filtros convolucionales es de 3'074.072.

ResNet-7: El modelo Resnet está compuesta de un total de 7 bloques residuales cada bloque puede ser definido como un identity-block o un conv-block. Los identity-block son los bloques estándar usados en ResNet, estos poseen una serie de filtros convolucionales y una conexión de atajo la cual salta estos bloques. Este bloque tiene las mismas dimensiones entrada y salida. Los conv-block son los tipos de bloque donde las dimensiones de entrada y salida no coinciden. La diferencia con el identity-block es una capa de convolucional en el atajo hacia la salida. Los beneficios de estas arquitecturas frente a las tradicionales, es que en los tradicionales al tener una alta cantidad de capas en el entrenamiento, surge el problema del desvanecimiento del error. Los modelos ResNet con sus conexiones entre capas previas son efectivas resolviendo este problema [38].

4.4. Funciones de aprendizaje por similitud: una estrategia con triple pérdidas

Como se introdujo en la sección 2.4. La métrica de distancia que da origen a la función de triple perdida fue introducida inicialmente como una transformación lineal de los datos en [39], que luego fue generalizada en el aprendizaje profundo para poder encontrar una transformación no lineal de los datos $\Phi(\mathbf{x})$. Para determinar la transformación adecuada, la función de triple perdida $\mathcal{L}$ fue introducida y es definida como:

$$\mathcal{L} = \sum_{\mathcal{S}_\mathrm{T}} [d_M^2(\mathbf{x}^a, \mathbf{x}^p) - d_M^2(\mathbf{x}^a, \mathbf{x}^n) + \alpha]_+ \tag{4.1}$$

Donde $\mathcal{S}_\mathrm{T}$ es el conjunto de tripletas para el entrenamiento el cual se denotado como:

$$\mathcal{S}_\mathrm{T} = \{(\mathbf{x}^a, y^a), (\mathbf{x}^n, y^n), (\mathbf{x}^p, y^p) | \quad y^a = y^p, y^a \neq y^n\} \tag{4.2}$$

El objetivo final de esta función es encontrar una transformación no lineal $\Phi(\mathbf{x})$, que haga más pequeña la distancia intraclase en comparación de la distancia interclase en el conjunto de datos $\mathcal{S}_\mathrm{T}$.

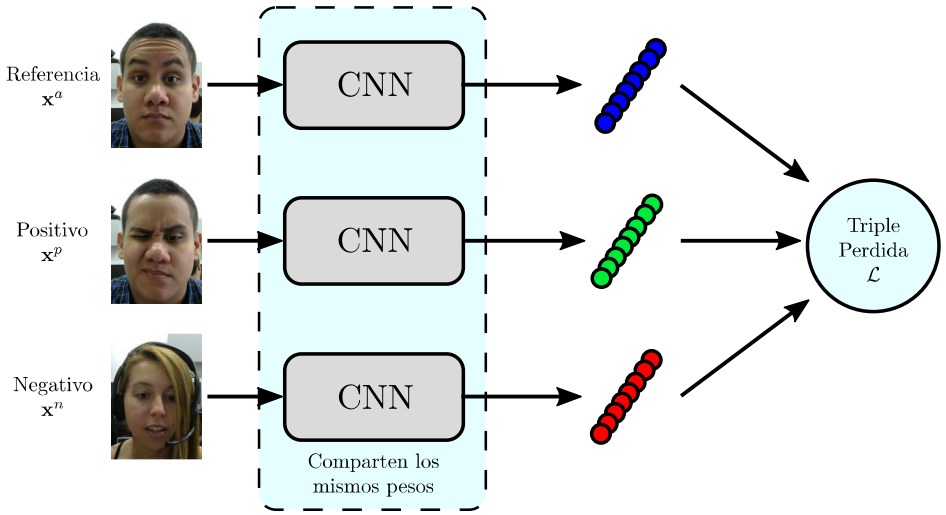


Figura 4.3. Arquitectura usada para el entrenamiento utilizando la función de triple perdida.

La Figura 4.3 muestra la arquitectura planteada en este trabajo, en ella se muestra un ejemplo de los mas usados en el reconocimiento de rostros, donde

se diseña una nuevas base de datos creada con lotes de imágenes agrupados en tripletas, la arquitectura es diseñada con el objetivo de mejorar la distancia relativa entre los datos de entradas $\mathbf{x}^a$, $\mathbf{x}^n$, $\mathbf{x}^p$; acercando los rostros similares y alejando los diferentes.

4.5. Optimización de hiper-parámetros

La clasificación automática entre personas sanas y pacientes con EP se realiza mediante máquinas de soporte vectorial (Support Vector Machine, SVM). La clasificación de los pacientes con diferentes grados de deterioro se realiza utilizando SVM optimizadas bajo la estrategia de uno contra todos y también con clasificadores de Bosque Aleatorio (Random Forest, RF). En los experimentos de clasificación binaria con SVM, se consideran los kernels lineales y Gaussianos. La optimización de los hiperparámetros se realiza en una malla de búsqueda hasta potencias de diez con $C \in \{10^{-4}, 10^{-3}, 10^{-2}, \ldots, 10^{2}, 10^{3}\}$ y $\gamma \in \{10^{-4}, 10^{-3}, 10^{-2}, \ldots, 10^{3}\}$ para el kernel gaussiano, y para el kernel lineal la búsqueda consideró $C \in \{10^{-1}, 1, 10^{1}, 10^{2}, 10^{3}, 10^{4}\}$. En la clasificación de múltiples clases sólo se considera el kernel lineal. Los hiperparámetros del clasificador de RF también se optimizan siguiendo una malla de búsqueda con el número de árboles $N_T \in \{100, 300, ..., 1200\}$ profundidad $D \in \{5, 8, 12, 15, 20, 25, 30\}$ y mínimo de muestras por división $S_m \in \{5, 15, 30, ...,100\}$. La optimización y evaluación de los modelos se realiza siguiendo una estrategia de validación cruzada de 5-folds. Los resultados de la clasificación binaria se reportan en términos de acierto (Acc), sensibilidad (Sens), especificidad (Spec), F1-score (F1), y área bajo la curva de la característica de operación del receptor (AUC). Los resultados de la clasificación multiclase se informan en términos de acierto (Acc), F1-score (F1), coeficiente Kappa (κ), y matriz de confusión. En todos los casos, los resultados incluyen valores de los hiperparámetros óptimos que se encuentran como la moda a lo largo de los parámetros considerados a lo largo de los folds de pruebas de cada experimento.

Capítulo 5

Marco Experimental

Proponemos un marco experimental en el que se abordan diferentes niveles de análisis. A continuación se presenta la lista de esos niveles y sus correspondientes hipótesis de base.

- ✓ **Nivel de reconocimiento (N1):** Nuestro protocolo de adquisición introdujo tareas faciales que incluían estados emocionales evocados (sonrisa, ira y sorpresa) y gestos coordinados (guiño del ojo derecho, guiño del ojo izquierdo). Se evalúa la utilidad de los métodos de reconocimiento facial para detectar la hipomimia. *Hipótesis (H1):* Las respuestas mostradas en la grabaciones intensifican las características necesarias para modelar la hipomimia en pacientes con Parkinson. *Experimento:* La arquitectura típica de ResNet50 entrenada con la base de datos VGGFace2 se utiliza para clasificar entre pacientes con EP y sujetos sanos.

- ✓ **Nivel de transferencia de aprendizaje (N2):** Proponemos mejorar el modelo anterior (ResNet50 + VGGFace2) haciendo transferencia de aprendizaje del dominio de las unidades de acción con el conjunto de datos de EmotioNet. *Hipótesis (H2):* La detección automática de la hipomimia mejora cuando se incorpora la información del dominio de las emociones. *Experimento:* Los pesos del modelo ResNet50 + VGGFace2 se actualizan con información extraída del conjunto de datos de EmotioNet. El modelo resultante se utiliza para clasificar entre pacientes con EP y sujetos sanos.

- ✓ **Nivel de función de costos (N3):** Evaluaremos las funciones de aprendizaje afectivo para mejorar el rendimiento de los modelos de

clasificación entrenados. *Hipótesis (H3):* Las funciones de aprendizaje de similitudes diseñadas para mejorar la información de los gestos de la cara servirán para mejorar la capacidad de detectar la hipomimia. *Experimento:* Las funciones de triple pérdida se utilizan para crear modelos de aprendizaje de similitudes que mejoran el rendimiento de la clasificación del modelo creado en el análisis con la norma L2.

✓ **Nivel de clasificación de estado neurológico (N4):** Las representaciones faciales de mejor rendimiento se utilizan para clasificar a los pacientes con diferentes niveles de deterioro neurológico de acuerdo con las puntuaciones del MDS-UPDRS-III. *Hipótesis (H4):* Las representaciones faciales aprendidas en modelos anteriores tienen información para identificar y evaluar los diferentes niveles de deterioro neurológico en los pacientes con la enfermedad de Parkinson. *Experimento:* Los modelos creados para representar la hipomimia se utilizan para evaluar tres estados neurológicos diferentes según las puntuaciones de la MDS-UPDRS-III.

El marco experimental se resume en la Figura 5.1. Los detalles de los métodos implementados para validar cada hipótesis se presentan en la Sección 4.

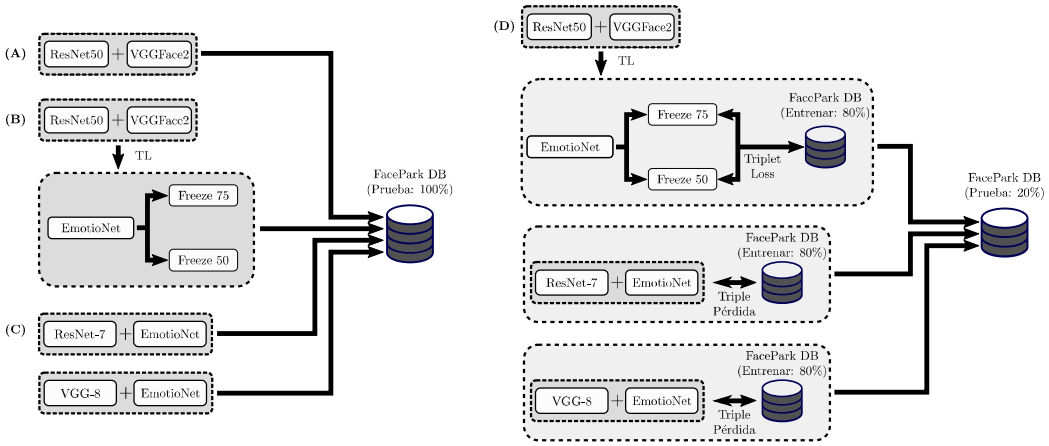


Figura 5.1. Marco experimental propuesto para el desarrollo de este trabajo. TL: Transfer Learning.

Capítulo 6

Experimentos y resultados

Todos los experimentos se realizaron con videos que evocan expresiones faciales como sonrisa, ira, sorpresa, guiño del ojo izquierdo, y guiño del ojo derecho. Se extrajeron cinco imágenes por video con el software Affectiva[1]. La curva de valencia proporcionada por el software se utiliza como criterio para seleccionar la siguiente secuencia de cinco imagenes por participante en cada expresión: (i) neutral; (ii) transición de neutral a Apex (Onset); (iii) Apex; (iv) transición Apex a neutral (Ofset); y (v) neutral. La secuencia de imágenes y su relación directa con la curva de valencia se ilustran en la Figura 6.1.

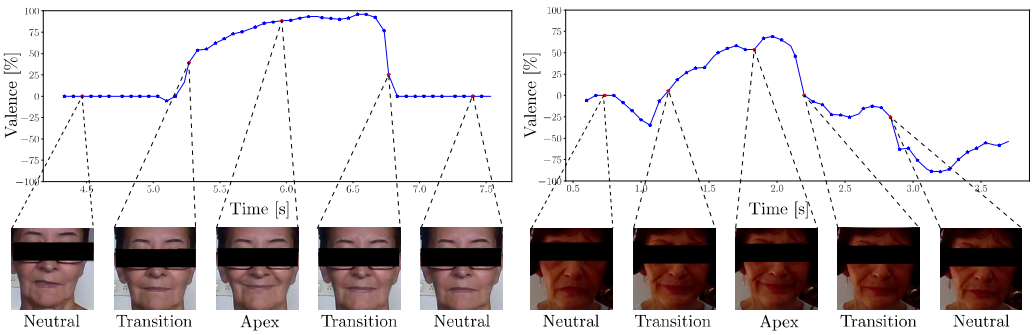


Figura 6.1. Múltiples etapas en la generación de una expresión midiendo la Valencia. (izquierda) Mujer sana de aproximadamente 63 años, (derecha) Mujer con enfermedad de Parkinson de aproximadamente 67 años con un valor de 2 en el ítem de expresión facial en la escala MDS-UPDRS-III.

[1]https://www.affectiva.com/

6.1. Experimentos 1: Nivel de reconocimiento

Imágenes individuales correspondientes a cada estado de la expresión mostradas en la Figura 6.1 serán consideradas para evaluar si cada imagen proporciona información relevante para discriminar entre los pacientes con EP y los sujetos sanos. El vector de caracteristicas es obtenido desde la ultima capa convolucional del modelo VGGFace2. La Tabla 6.1 resume todos los resultados.

Tabla 6.1. Resultados de clasificación utilizando una simple imagen de la secuencia de imágenes extraída

Expresión	**Kernel***	**Acc**[%]	**Sens**[%]	**Spec**[%]	**F1**[%]
Neutral	C=1e+01; γ=1e-04	69.0 ± 10.1	74.0 ± 11.6	63.0 ± 9.7	67.8 ± 10.1
Apex	C=1e+01; γ=1e-04	70.0 ± 9.1	84.4 ± 7.9	53.3 ± 24.0	61.0 ± 18.6
Onset	C=1e+01; γ=1e-04	71.4 ± 3.2	88.6 ± 7.0	50.0 ± 9.0	63.1 ± 6.6
Offset	C=1e+01; γ=1e-04	71.6 ± 5.2	79.5 ± 3.3	61.9 ± 13.5	68.6 ± 8.2
Neutral	C=1e-03	70.8 ± 9.6	77.3 ± 10.2	63.0 ± 9.7	69.3 ± 9.7
Apex	C=1e-03	70.8 ± 9.1	83.7 ± 7.3	55.7 ± 21.6	63.8 ± 16.3
Onset	**C=1e-02**	**72.9 ± 4.2**	**88.6 ± 7.8**	**53.4 ± 7.7**	**66.1 ± 5.9**
Offset	C=1e-01	72.8 ± 4.3	81.5 ± 4.5	61.9 ± 13.5	69.2 ± 7.9

Primeras tres filas: kernel Gaussiano . Ultimas tres filas: kernel lineal.

*Columna con los hiperparámetros óptimos.

Obsérvese que casi no hay diferencia entre los aciertos obtenidos con los imágenes de cada etapa de expresión. Quizás lo único que hay que destacar es la alta sensibilidad (88,6 %) de la etapa de Onset, lo que probablemente indica que esta etapa es quizás una buena elección para modelar la hipomimia en cuadros específicos dentro de un video. Esta observación preliminar será elaborada más adelante en los próximos experimentos.

6.1.1. Secuencia de múltiples imágenes.

Dada la poca información que proporcionan los experimentos de expresión individual, proponemos el uso de secuencias de múltiples imágenes como

una forma de capturar información dinámica durante la producción de expresiones faciales. La idea general fue propuesta por primera vez en [48] para el análisis de señales de habla, donde el autor planteo la hipótesis de que los pacientes de EP tienen mas dificultades para iniciar o detener el movimiento de los músculos durante la producción del habla. La idea se extendió mas tarde a otros movimientos como la escritura y la marcha [49].

Al igual que en el caso del habla, la marcha y la escritura, creemos que la misma hipótesis se mantiene durante la producción de las expresión faciales. Por lo tanto, el análisis de múltiples imágenes durante la producción de la expresión facial proporciona información útil para discriminar entre los pacientes con EP y los sujetos sanos.

- NOnA: Neutral, Onset, y Apex.
- AOffN: Apex, Offset, y Neutral.
- NOnAOffN: Neutral, Onset, Apex, Offset, y Neutral.

La Tabla 6.2 muestra los resultados obtenidos cuando la dinámica en la producción de expresiones faciales es incorporada extrayendo vectores de características de las secuencias de múltiples imágenes.

Tabla 6.2. Resultados de clasificación utilizando las secuencias de imágenes extraídas durante la producción de la expresión facial.

Secuencia	**Kernel***	**Acc**[%]	**Sens**[%]	**Spec**[%]	**F1**[%]
NOnA	C=1e+02; γ=1e-04	77.4 ± 8.7	89.3 ± 4.6	63.0 ± 16.1	72.9 ± 11.2
AOffN	C=1e+01; γ=1e-04	76.3 ± 8.0	86.8 ± 12.0	63.5 ± 22.4	69.2 ± 17.8
NOnAOffN	C=1e+01; γ=1e-04	77.2 ± 8.6	86.1 ± 14.8	67.2 ± 10.3	74.2 ± 8.5
NOnA	C=1e-03	78.2 ± 9.8	90.1 ± 5.2	63.8 ± 17.1	73.8 ± 12.6
AOffN	C=1e-03	77.8 ± 9.1	88.8 ± 9.4	64.2 ± 24.1	70.4 ± 20.5
NOnAOffN	**C=1e-03**	**78.4 ± 7.1**	**87.8 ± 11.4**	**67.7 ± 11.6**	**75.4 ± 7.9**

Primeras tres filas: kernel Gaussiano . Ultimas tres filas: kernel lineal.

*Columna con los hiperparámetros óptimos.

Los resultados con las secuencias son mejores que los obtenidos con los cuadros individuales. La mejora es del alrededor del 7 % y el mejor resultado

se obtiene en los dos casos en los que se incluye la secuencia NOnA, que se centra en modelar la información en la transición entre el estado neutral y la producción de alguna expresión. También cabe destacar que la sensibilidad se acerca al 90 % en todos los casos, mientras que la especificidad es bastante baja (alrededor del 64 %). Esto indica que el enfoque propuesto es bueno para detectar pacientes pero no tan bueno para detectar controles sanos.

Con el fin de mostrar la mejora obtenida al considerar las secuencias de múltiples imágenes, los resultados obtenidos con las imágenes Onset y con las secuencias de NOnA se muestran en las curvas Característica Operativa del Receptor (Receiver Operating Characteristic, ROC) de la Figura 6.2. Nótese que cuando se considera la secuencia NOnA, el valor de AUC es 5 % puntos más alto que cuando sólo se procesa individualmente en las imágenes Onset. Este resultado valida la hipótesis sobre la existencia de información útil en la dinámica durante la producción de ciertas expresiones faciales. Dada esta clara mejora, los próximos experimentos incluirán solo vectores de características extraídos de secuencia de múltiples imágenes.

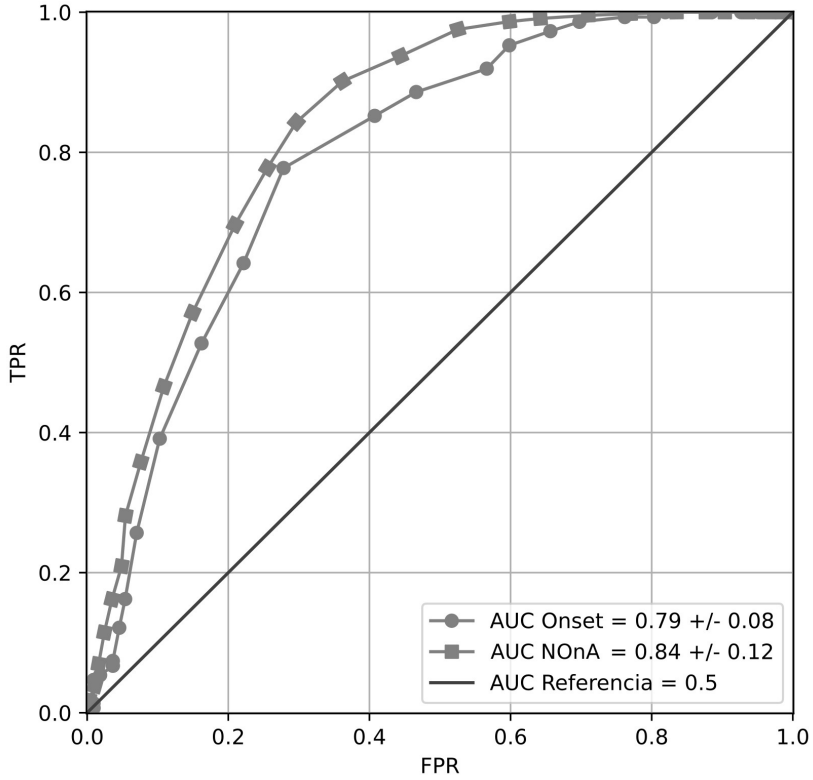


Figura 6.2. Comparación de las curvas ROC obtenidas con imágenes individuales Onset vs. la secuencia NOnA.

La Figura 6.3 muestra el resultado de proyectar el espacio de representación creado con vectores de características de la secuencia NOnAOffN a un espacio de 2D. El Análisis de Componentes Principales (Principal Component Analysis, PCA) se utiliza en este caso con el objetivo de proporcionar una ilustración visual del espacio de representación generado. El hiperplano resultante después de optimizar la SVM, se incluye sólo como una referencia visual para posteriores comparaciones con los métodos que se van a evaluar en los próximos experimentos. La parte inferior de la Figura 6.3 también muestra la distribución de los puntajes en clasificación, que se refieren a la distancia de cada muestra al hiperplano de separación en el espacio de representación. Con el fin de determinar la heterogeneidad entre los pacientes y los controles sanos, se realiza la prueba Mann-Whitney U en la distribución de las puntuaciones obteniendo un valor ($p \ll 0.01$).

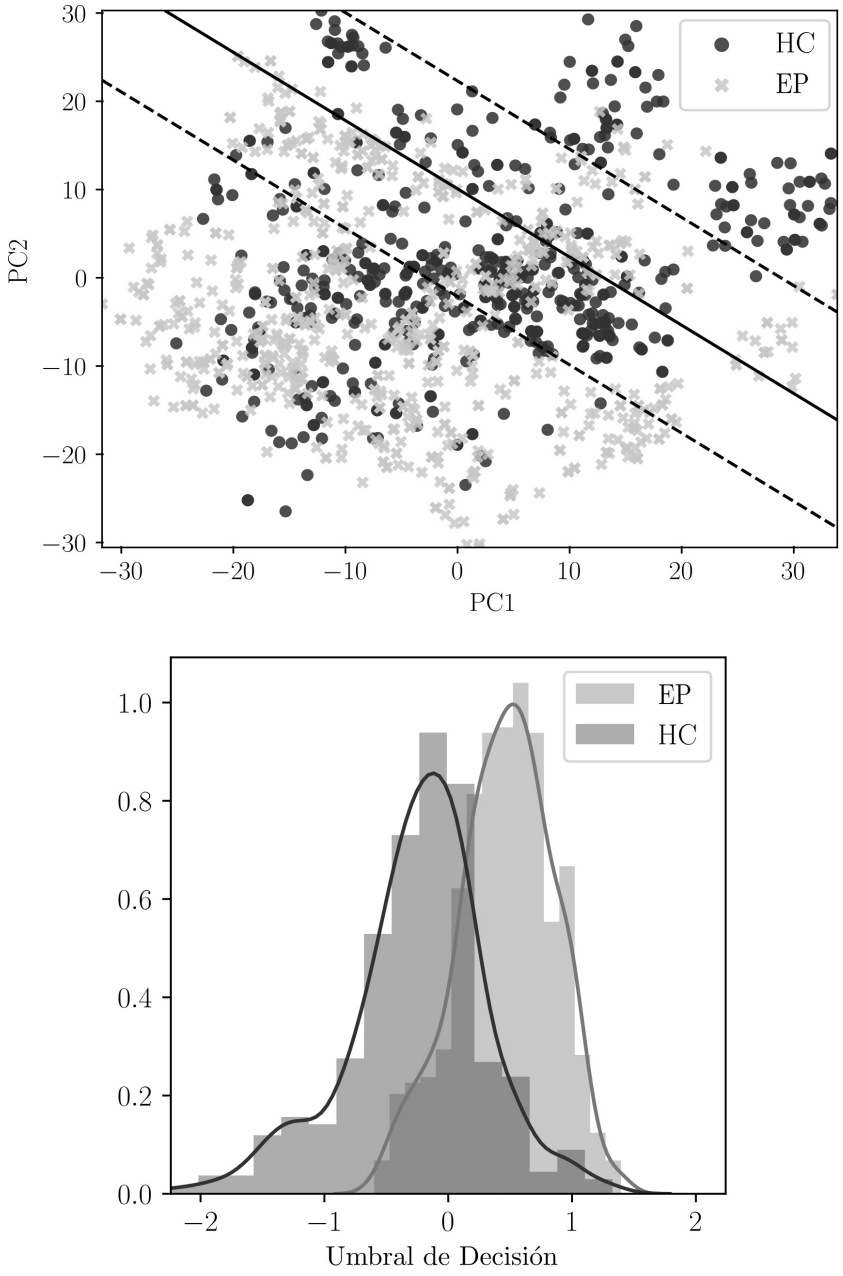
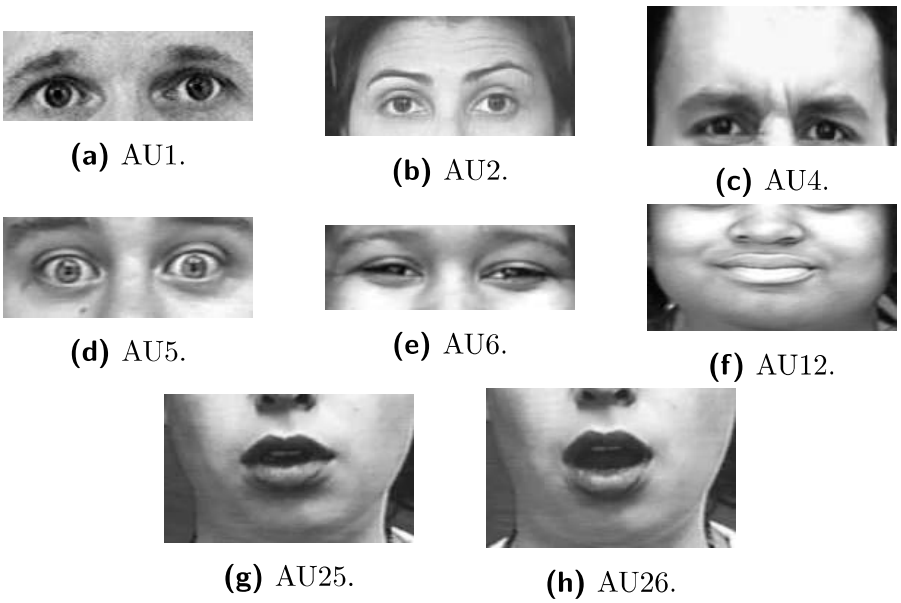


Figura 6.3. (Arriba) Espacio de representación en 2D creado con PCA sobre los vectores de características extraídos de la secuencia NOnAOffN. (Abajo) Distribución de las puntuaciones del SVM para pacientes con EP y personas sanas.

6.2. Experimento 2: Nivel de transferencia de aprendizaje

Este experimento pretende incorporar información del Dominio de Computación Afectiva (AC-D) en la tarea de reconocimiento facial. En este caso, la base de datos EmotioNet se utiliza para crear un espacio de representación apropiada para el AC-D. El primer paso consiste en seleccionar aquellas AUs que proporcionen información adecuada para realizar la clasificación automática de pacientes con EP y sujetos sanos. Seleccionamos un subconjunto de AUs según [30], de tal manera que permita un mejor modelado de las expresiones faciales incluidas en las tareas de registro de la base de datos FacePark-GITA. La Figura 6.4 muestra el conjunto de AUs seleccionadas.



(a) AU1. **(b)** AU2. **(c)** AU4. **(d)** AU5. **(e)** AU6. **(f)** AU12. **(g)** AU25. **(h)** AU26.

Figura 6.4. Unidades de acción utilizadas en el entrenamiento de los modelos propuesto en esta sección. Imágenes tomadas de [29].

6.2.1. Reentrenamiento de modelos: Congelamiento de capas

El proceso de reentrenamiento de los modelos convolucionales consiste en congelar diferentes porcentajes de las capas y reentrenar la porción restante. Los datos con las AUs seleccionadas del conjunto de datos de EmotioNet se usan aquí para reentrenar los modelos. En este caso consideramos tres

escenarios: congelando el 50 % (Freeze 50), el 75 % (Freeze 75), y el 100 % (Freeze 100). Observé que el modelo Freeze 100 corresponde al caso en el que no se incorpora el AC-D. Después de las capas convolucionales se agrega una capa totalmente conectas para la clasificación de las 8 AUs. El resultado del reentrenamiento y su rendimiento para clasificar AUs se muestra en Tabla 6.3 en términos de los valores de AUC y el EER.

El reentrenamiento fue realizado con el paquete Tensorflow 2.0 en un entorno Python 3.8 con un tamaño de lote (batch size) de 128, un optimizador ADAM con tasa de aprendizaje $lr = 0.001$ y con parámetros $\beta_1 = 0.9$, $\beta_2 = 0.999$, y $\epsilon = 1e{-}7$. Además, se implemento la regularización a través de la interrupción temprana (early stopping) con una paciencia de 20 épocas monitoreando los aciertos en validación.

Tabla 6.3. Resultados del reentrenamiento del modelo VGG-Face2 con la base de datos EmotioNet.

Modelos	Métricas	AU 1	AU 2	AU 4	AU 5	AU 6	AU 12	AU 25	AU 26
Freeze 100	AUC	0.83	0.83	0.87	0.80	0.94	0.95	0.92	0.80
	EER [%]	24.58	23.78	21.01	27.13	12.82	12.11	15.38	27.32
Freeze 75	AUC	0.84	0.84	0.86	0.84	0.92	0.93	0.95	0.85
	EER [%]	21.84	20.80	19.90	21.65	14.34	10.42	8.63	22.48
Freeze 50	AUC	0.84	0.87	0.87	0.87	0.93	0.95	0.90	0.83
	EER [%]	20.56	19.29	18.92	19.53	13.22	10.58	10.99	24.32

Los modelos reentrenados se utilizan además para clasificar entre los pacientes con EP y los sujetos sanos del corpus FacePark-GITA. Los resultados obtenidos con los modelos Freeze 75 y Freeze 50 se muestran en la Tabla 6.4 y en la Tabla 6.5 respectivamente. Los resultados para el modelo Freeze 100 corresponden a los mostrados anteriormente en la Tabla 6.2. Los hiperparámetros óptimos encontrados en validación cruzada de 5-folds también se incluyen en cada experimento.

Tabla 6.4. Resultados de clasificación utilizando el modelo Freeze 75.

Secuencia	**Kernel***	**Acc**[%]	**Sens**[%]	**Spec**[%]	**F1**[%]
NOnA	C=1e+01; γ=1e-04	84.2 ± 5.4	90.0 ± 8.3	77.2 ± 10.8	82.3 ± 6.3
AOffN	C=1e+02; γ=1e-04	81.6 ± 8.6	87.8 ± 7.4	73.9 ± 11.5	80.0 ± 9.5
NOnAOffN	C=1e+02; γ=1e-04	86.7 ± 8.9	91.2 ± 4.7	81.6 ± 15.5	85.5 ± 10.2
NOnA	C=1e-01	84.7 ± 5.4	89.5 ± 9.4	78.9 ± 11.3	82.9 ± 6.5
AOffN	C=1e-01	82.6 ± 9.6	87.8 ± 8.3	76.1 ± 13.3	81.2 ± 10.4
NOnAOffN	**C=1e-01**	**87.3 ± 8.0**	**90.6 ± 5.0**	**83.6 ± 13.1**	**86.6 ± 8.8**

Primeras tres filas: kernel Gaussiano . Ultimas tres filas: kernel lineal.

*Columna con los hiperparámetros óptimos.

Tabla 6.5. Resultados de clasificación utilizando el modelo Freeze 50.

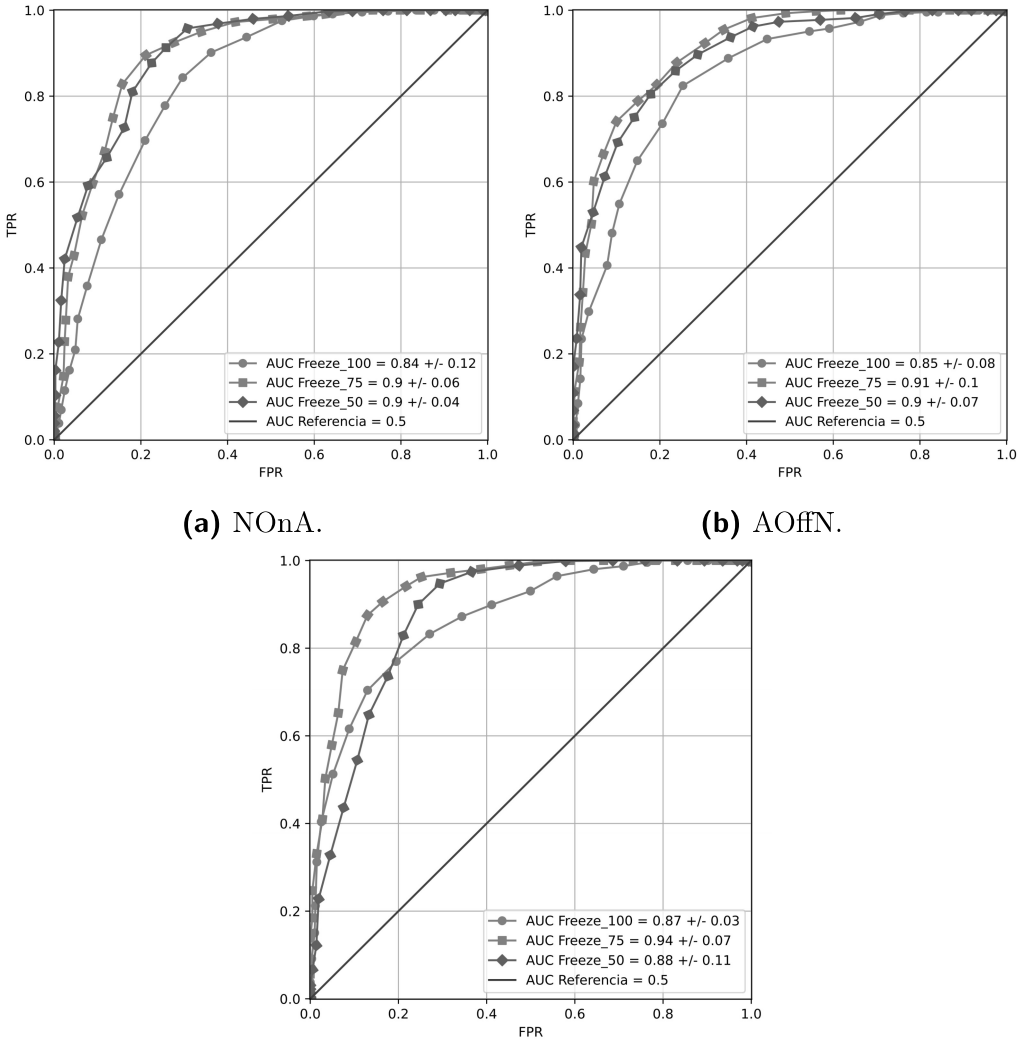
Secuencia	**Kernel***	**Acc**[%]	**Sens**[%]	**Spec**[%]	**F1**[%]
NOnA	**C=1e+01; γ=1e-04**	**83.1 ± 6.0**	**87.7 ± 12.4**	**77.5 ± 10.2**	**81.1 ± 6.5**
AOffN	C=1e+01; γ=1e-04	81.3 ± 7.5	86.3 ± 13.0	75.6 ± 3.6	80.1 ± 6.8
NOnAOffN	C=1e+00; γ=1e-04	81.9 ± 9.2	97.4 ± 2.5	63.4 ± 17.7	75.5 ± 14.3
NOnA	C=1e-01	82.1 ± 6.8	85.0 ± 13.8	78.6 ± 11.0	80.2 ± 7.7
AOffN	C=1e-01	80.0 ± 7.6	83.4 ± 12.7	76.1 ± 4.4	79.1 ± 7.2
NOnAOffN	C=1e-01	80.2 ± 11.1	84.3 ± 8.5	75.3 ± 19.1	78.3 ± 13.0

Primeras tres filas: kernel Gaussiano . Ultimas tres filas: kernel lineal.

*Columna con los hiperparámetros óptimos.

Obsérvese que el Freeze 75 presenta mayores aciertos que el Freeze 50, lo que indica que todavía se requiere una considerable información del dominio del reconocimiento facial para obtener buenos resultados en la clasificación entre los pacientes de EP y los sujetos sanos. Más interesantes es que los mejores aciertos obtenidos con el modelo Freeze 75 en la Tabla 6.4 (87.3 %) es 8.9 % mas alto que el mejor resultado obtenido cuando solo se considera un enfoque de solo reconocimiento facial (Tabla 6.2). Este resultado apoya la principal contribución de este trabajo, donde la idea de transferir la información del AC-D a la tarea clásica de reconocimiento facial se presenta

como un buen enfoque para detectar la hipomimia en pacientes con EP. Los beneficios de incluir información del dominio afectivo también se muestran en la Figura 6.5, donde se presentan las curvas ROC obtenidas con los modelos Freeze 50, Freeze 75 y Freeze 100.



(a) NOnA. **(b)** AOffN.

(c) NOnAOffN.

Figura 6.5. Desempeño de las diferentes secuencias de entrada en los modelos Freeze reentrenados.

Nótese que los modelos utilizados hasta este punto del estudio se basan en arquitecturas con un alto número de parámetros en sus capas convolucionales. Ahora queremos evaluar si es posible obtener un rendimiento similar cuando

se utilizan arquitecturas más pequeñas con muchos menos parámetros para optimizar en la red.

6.2.2. Modelos VGG-8 y ResNet-7

El escenario anterior en el que se introdujo la estrategia de congelación consideraba 22'650,882 y 18'182,146 parámetros para Freeze 50 y Freeze 75, respectivamente. Por el contrario, cuando se considera la arquitectura del VGG-8, sólo se requiere optimizar un total de 295,448 parámetros. Este caso es similar para la arquitectura Resnet-7, en la que se consideran 366,626 parámetros. Estas arquitecturas reducidas se entrenan con los mismos datos que los considerados anteriormente para volver a entrenar los modelos Freeze 50 y Freeze 75. La Tabla 6.6 muestra los resultados con los valores de AUC obtenidos cuando se detectan las diferentes AUs. Obsérvese que estos resultados son superiores a los reportados en la Tabla 6.3 donde se optimiza un mayor número de parámetros. Este resultado indica que un modelo más simple proporciona un rendimiento de discriminación de AUs lo suficientemente alto para ser usado en la clasificación posterior entre pacientes con EP y controles sanos.

Tabla 6.6. Resultados del entrenamientos de los modelos VGG-8 y Resnet-7 con la base de datos EmotioNet.

Modelos	Métricas	AU 1	AU 2	AU 4	AU 5	AU 6	AU 12	AU 25	AU 26
Resnet-7	AUC	0.92	0.93	0.91	0.91	0.96	0.97	0.97	0.91
	EER [%]	15.25	14.21	16.20	13.58	10.05	8.42	7.39	16.32
VGG-8	AUC	0.89	0.87	0.89	0.90	0.96	0.96	0.96	0.90
	EER [%]	16.59	16.08	16.88	14.87	9.51	8.11	7.83	16.55

La Tabla 6.7 y Tabla 6.8 muestran los resultados obtenidos cuando los modelos mencionados, creados con las arquitecturas reducidas, se usan para discriminar entre pacientes con EP y sujetos sanos. Nótese que no se realiza ningún entrenamiento adicional con los datos de los pacientes con enfermedad de Parkinson. Los mejores resultados se obtienen cuando se considera la arquitectura de Resnet-7 con características extraídas de la secuencia NOnAOffN. Aunque el 78.8 % podría considerarse un cantidad de aciertos, todavía está lejos del mejor resultado obtenido con el modelo Freeze 75 (87.3 %

en la Tabla 6.4), lo que indica que hay espacio para más experimentos en los que se considere la función de triple pérdida.

Tabla 6.7. Resultados de clasificación utilizando el modelo VGG-8.

Secuencia	**Kernel***	**Acc**[%]	**Sens**[%]	**Spec**[%]	**F1**[%]
NOnA	C=1e+01; γ=1e-02	58.3 ± 3.7	94.6 ± 4.8	14.1 ± 6.3	24.0 ± 9.8
AOffN	C=1e+01; γ=1e-03	65.6 ± 8.6	80.6 ± 8.0	47.6 ± 16.4	58.1 ± 12.9
NOnAOffN	C=1e+01; γ=1e-04	62.7 ± 8.3	66.4 ± 10.0	58.2 ± 13.1	60.9 ± 8.6
NOnA	C=1e-02	67.4 ± 8.3	72.4 ± 9.4	61.3 ± 9.8	66.0 ± 8.2
AOffN	**C=1e-02**	**67.6 ± 5.8**	**70.6 ± 7.4**	**63.9 ± 13.5**	**65.9 ± 7.3**
NOnAOffN	C=1e-02	64.9 ± 7.7	71.0 ± 4.5	57.7 ± 16.1	62.2 ± 11.0

Primeras tres filas: kernel Gaussiano . Ultimas tres filas: kernel lineal.

*Columna con los hiperparámetros óptimos.

Tabla 6.8. Resultados de clasificación utilizando el modelo ResNet-7.

Secuencia	**Kernel***	**Acc**[%]	**Sens**[%]	**Spec**[%]	**F1**[%]
NOnA	C=1e+03; γ=1e-04	73.0 ± 9.5	75.9 ± 18.7	69.7 ± 17.8	68.9 ± 12.3
AOffN	C=1e+01; γ=1e-02	73.4 ± 9.9	81.7 ± 15.6	63.6 ± 8.9	70.5 ± 9.5
NOnAOffN	**C=1e+03; γ=1e-04**	**78.8 ± 6.4**	**79.3 ± 9.8**	**78.2 ± 12.8**	**77.6 ± 6.7**
NOnA	C=1e-02	74.1 ± 6.9	82.2 ± 19.4	64.5 ± 11.4	69.3 ± 6.1
AOffN	C=1e-02	72.4 ± 10.8	84.2 ± 16.5	58.2 ± 8.6	68.1 ± 9.6
NOnAOffN	C=1e-01	78.3 ± 7.3	80.1 ± 10.6	76.2 ± 10.1	77.3 ± 7.4

Primeras tres filas: kernel Gaussiano . Ultimas tres filas: kernel lineal.

*Columna con los hiperparámetros óptimos.

La Figura 6.6 muestra tres curvas ROC donde se comparan los resultados con Freeze 75, ResNet-7, y VGG-8. Se observa claramente la superioridad del modelo Freeze 75, lo que apoya las ventajas de hacer un aprendizaje de transferencia en lugar de crear modelos desde cero.

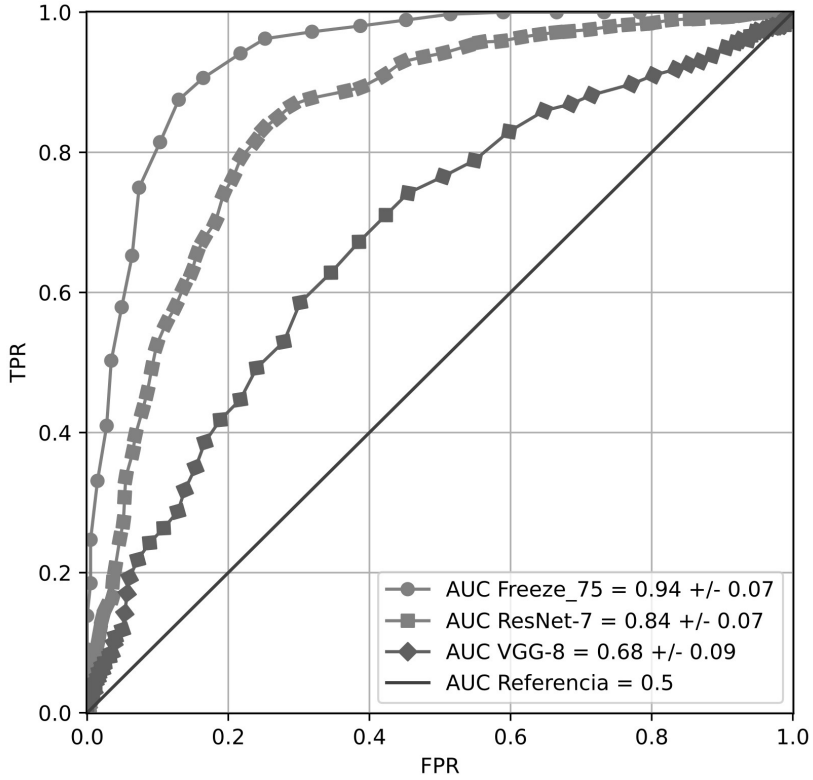


Figura 6.6. Comparación entre el desempeño de los clasificadores utilizando la secuencia NOnAOffN en los modelos Freeze 75, Resnet-7 y VGG-8.

Como en los experimentos anteriores, con el objetivo de proporcionar al lector una visión visual, PCA se utiliza para crear una representación 2D del espacio. La Figura 6.7 muestra el resultado obtenido con el modelo Freeze 75 cuando se consideran cuadros de la secuencia NOnAOffN. La parte inferior de la figura muestra la distribución del puntaje en la clasificación. La prueba Mann-Whitney U para comprobar la heterogeneidad de las poblaciones es realizada en la distribución de las puntuaciones de la SVM, obteniendo un valor ($p \ll 0{,}01$).

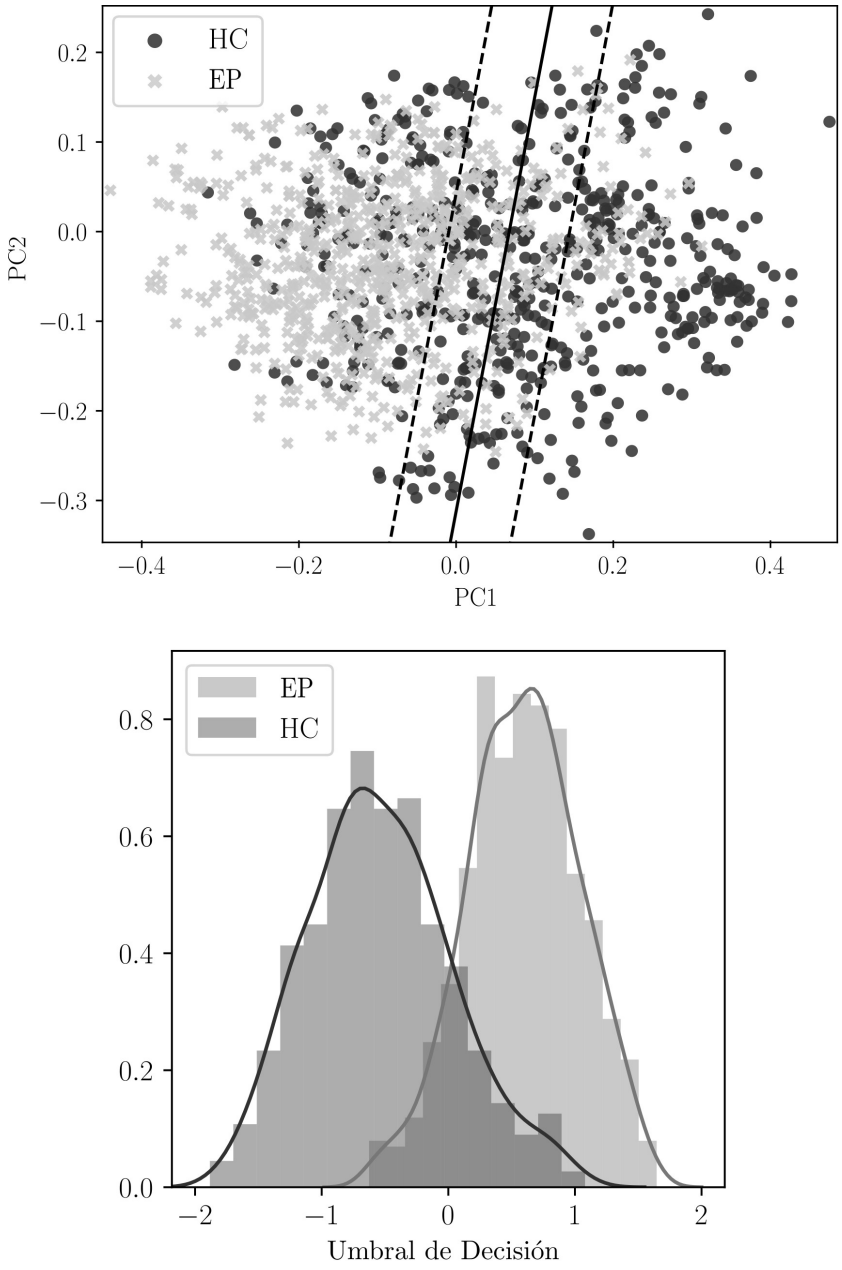


Figura 6.7. (Arriba) Espacio de representación 2D creado con PCA sobre características del modelo Freeze 75 extraído de la secuencia NOnAOffN. (Abajo) Distribución de las puntuaciones del SVM para pacientes con EP y personas sanas.

6.3. Experimento 3: Nivel de función de costos

Este enfoque se incluye en este trabajo con el fin de evaluar la conveniencia de incorporar la función de triple pérdida para mejorar los resultados de la clasificación. La función de triple pérdida tiene por objeto modificar el espacio de representación original de manera que se aumente la distancia entre clases y se reduzca la distancia dentro de cada clase. Los vectores de características modificados se denominan *vectores embebidos*.

6.3.1. Creación de vectores embebidos basados en modelos Freeze

Los modelos Freeze 75 y Freeze 50 serán entrenados con esta nueva estrategia creando dos nuevos modelos denominados Triplet 75 y Triplet 50 respectivamente. Los resultados para el conjunto de vectores embebidos se puede observar en la Tabla 6.9 y en la Tabla 6.10.

Tabla 6.9. Resultados de clasificación utilizando el modelo Triplet 75.

Secuencia	**Kernel***	**Acc**[%]	**Sens**[%]	**Spec**[%]	**F1**[%]
NOnA	C=1e+01; γ=1e-04	85.2 ± 7.4	87.6 ± 5.8	82.5 ± 12.6	84.5 ± 8.2
AOffN	C=1e+01; γ=1e-04	86.0 ± 6.1	91.4 ± 6.9	79.5 ± 7.1	84.9 ± 6.2
NOnAOffN	C=1e+01; γ=1e-04	86.0 ± 9.0	92.1 ± 6.9	78.7 ± 13.4	84.5 ± 10.1
NOnA	C=1e-01	84.4 ± 6.6	87.4 ± 4.4	80.9 ± 13.3	83.4 ± 7.6
AOffN	C=1e-01	85.0 ± 5.9	90.3 ± 6.4	78.7 ± 7.1	84.0 ± 6.1
NOnAOffN	**C=1e-01**	**86.1 ± 9.6**	**91.4 ± 7.5**	**79.9 ± 13.5**	**85.0 ± 10.5**

Primeras tres filas: kernel Gaussiano . Ultimas tres filas: kernel lineal.

*Columna con los hiperparámetros óptimos.

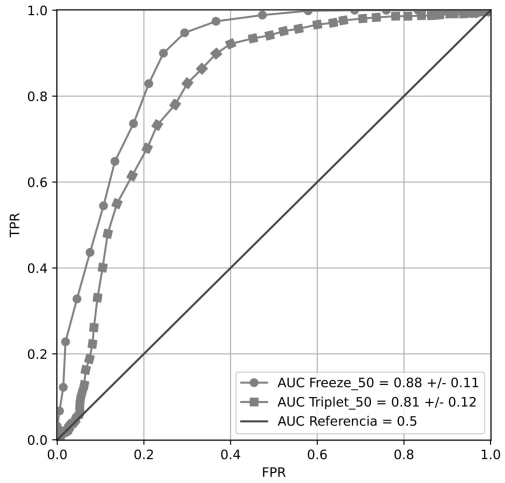
Tabla 6.10. Resultados de clasificación utilizando el modelo Triplet 50.

Secuencia	**Kernel***	**Acc**[%]	**Sens**[%]	**Spec**[%]	**F1**[%]
NOnA	C=1e+01; γ=1e-04	78.9 ± 5.5	84.3 ± 10.9	72.4 ± 11.3	76.7 ± 6.1
AOffN	C=1e+03; γ=1e-04	73.2 ± 8.7	69.1 ± 16.9	78.3 ± 4.0	72.2 ± 8.3
NOnAOffN	C=1e+02; γ=1e-04	75.8 ± 11.8	77.4 ± 15.5	74.3 ± 16.2	74.2 ± 12.5
NOnA	**C=1e-01**	**80.7 ± 6.6**	**86.4 ± 13.2**	**73.9 ± 11.8**	**78.1 ± 7.4**
AOffN	C=1e-01	76.3 ± 8.7	79.1 ± 17.4	73.3 ± 7.4	74.5 ± 8.6
NOnAOffN	C=1e-01	77.1 ± 10.2	83.0 ± 10.7	69.9 ± 19.8	73.9 ± 13.2

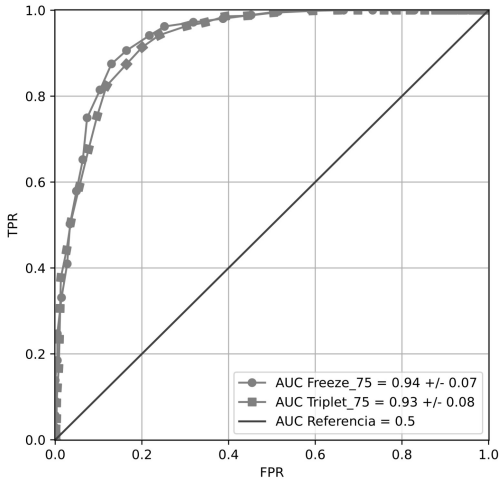
Primeras tres filas: kernel Gaussiano . Ultimas tres filas: kernel lineal.

*Columna con los hiperparámetros óptimos.

Observe que el modelo Triplet 75 exhibe mayores aciertos (86.1 %) que el Triplet 50 (80.7 %). Dado que los mejores aciertos en los experimentos anteriores con los modelos Freeze 75 y Freeze 50 fueron del 87.3 % y 83.1 %, estos nuevos resultados obtenidos con la estrategia de triple perdida probablemente indican que el enfoque de embedimentos no ofrece ventajas sobre el uso de transferencia de aprendizaje y congelación de capas. Esta observación también se apoya en el hecho de que no se ha reducido el número de parámetros que deben optimizarse, por lo que, en principio, no hay razón para utilizar la función de triple perdida en estos dos escenarios. La Figura 6.8 resume los resultados y muestra que hay una gran diferencia en el rendimiento de Freeze 50 vs. Triplet 50, pero casi no hay diferencia entre Freeze 75 vs. Triplet 75. Aunque no hay ninguna mejora al aplicar la estrategia de triple perdida sobre los modelos Freeze, vale la pena ver si la estrategia tiene un impacto positivo en los modelos reducidos, en los que se requieren menos parámetros para optimizar la red.



(a) Curva ROC de los modelos Freeze 50 y Triplet 50 utilizando la secuencia NOnAOffN.



(b) Curva ROC de los modelos Freeze 75 y Triplet 75 utilizando la secuencia NOnAOffN.

Figura 6.8. Efectos sobre el desempeño dado por la curva ROC, antes y después de utilizar la función de triple perdida.

El espacio 2D, resultante después de aplicar la transformación PCA sobre el modelo Triplet 75 con características extraídas de la secuencia NOnAOffN se presenta en la Figura 6.9. Aunque todavía hay varios errores de clasificación, se observa claramente la mejora en la separación de las dos clases. Además, la parte inferior de la figura muestra la distribución de los puntajes de clasificación. La prueba Mann-Whitney U para comprobar la heterogeneidad de las poblaciones es realizada en la distribución de las puntuaciones de la SVM, obteniendo un valor ($p \ll 0{,}01$).

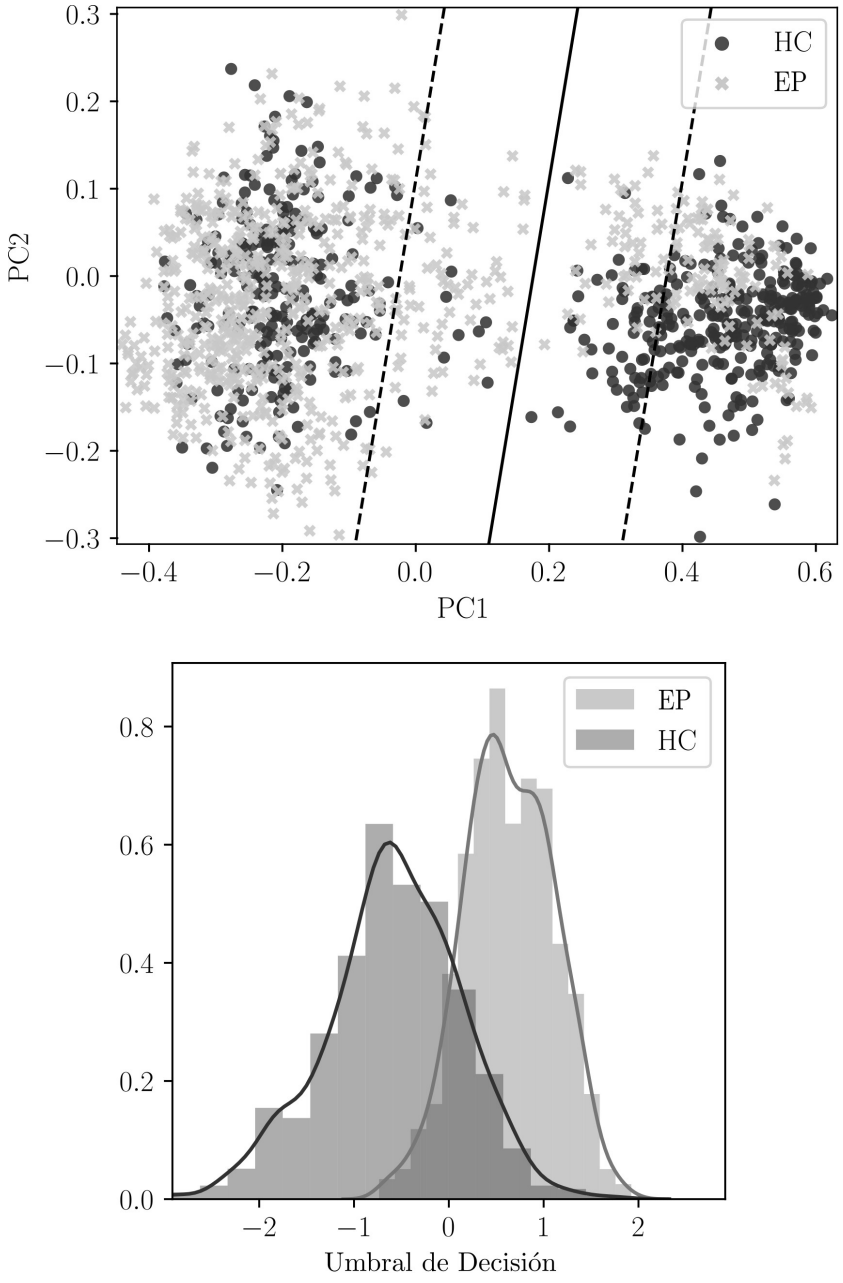


Figura 6.9. (Arriba) Espacio de componentes principales creado a partir de las características del modelo Triplet 75. (Abajo) Distribucción de las puntuaciones del SVM de pacientes con EP y personas sanas.

6.3.2. Creación de vectores embebidos basados en modelos VGG-8 y ResNet-7

Los modelos VGG-8 y ResNet-7 serán reentrenados con la función de triple perdida, creando dos nuevos modelos que denominaremos Triplet-VGG8 y Triplet-ResNet7 respectivamente. Estos nuevos modelos se utilizan para extraer vectores embebidos para una mejor clasificación entre pacientes con EP y sujetos sanos. Los resultados obtenidos con los vectores integrados Triplet-VGG8 y Triplet-ResNet7 se muestran en la Tabla 6.11 y en la Tabla 6.12.

Tabla 6.11. Resultados de clasificación utilizando el modelo Triplet-VGG8.

Secuencia	**Kernel***	**Acc**[%]	**Sens**[%]	**Spec**[%]	**F1**[%]
NOnA	C=1e+01; γ=1e-04	71.2 ± 8.8	76.4 ± 14.0	64.9 ± 12.8	68.7 ± 8.2
AOffN	C=1e+03; γ=1e-03	69.9 ± 9.6	67.4 ± 8.2	72.9 ± 13.1	69.8 ± 9.6
NOnAOffN	C=1e+00; γ=1e-03	66.0 ± 8.4	79.0 ± 10.5	50.7 ± 21.0	58.2 ± 14.9
NOnA	**C=1e-02**	**72.7 ± 7.2**	**80.8 ± 13.4**	**62.6 ± 11.5**	**69.1 ± 7.9**
AOffN	C=1e+01	70.3 ± 7.0	74.9 ± 9.4	64.8 ± 13.2	68.3 ± 7.8
NOnAOffN	C=1e+01	65.3 ± 5.1	65.0 ± 3.9	65.4 ± 13.7	64.1 ± 6.8

Primeras tres filas: kernel Gaussiano . Ultimas tres filas: kernel lineal.

*Columna con los hiperparámetros óptimos.

Tabla 6.12. Resultados de clasificación utilizando el modelo Triplet-ResNet7.

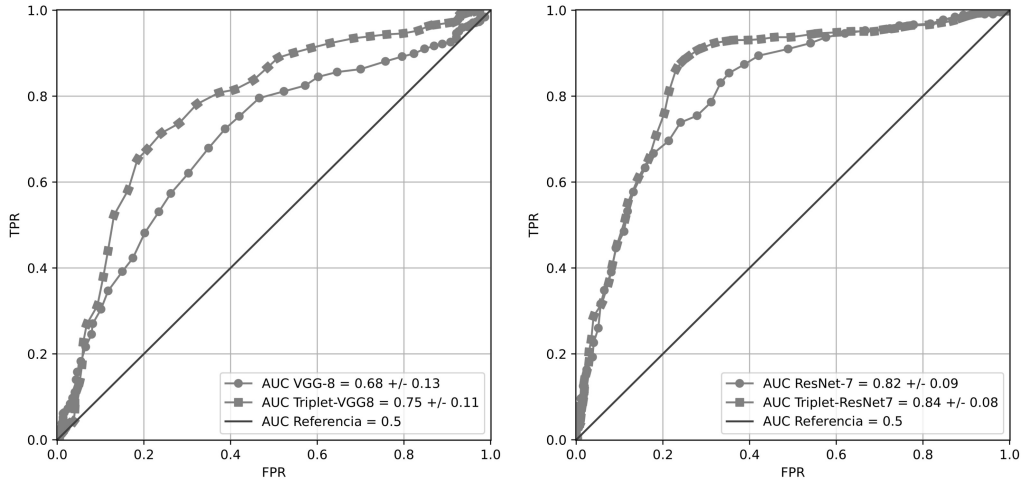
Secuencia	**Kernel***	**Acc**[%]	**Sens**[%]	**Spec**[%]	**F1**[%]
NOnA	C=1e+03; γ=1e-04	82.1 ± 8.8	87.2 ± 7.4	76.0 ± 14.3	80.5 ± 10.1
AOffN	C=1e+02; γ=1e-03	78.2 ± 12.9	79.6 ± 13.6	76.3 ± 16.3	77.3 ± 13.0
NOnAOffN	C=1e-01; γ=1e-03	69.9 ± 10.8	82.8 ± 15.2	54.7 ± 22.0	61.8 ± 17.9
NOnA	**C=1e-01**	**82.4 ± 8.5**	**89.2 ± 5.9**	**74.1 ± 12.6**	**80.7 ± 9.7**
AOffN	C=1e-01	76.2 ± 11.0	78.9 ± 12.5	72.8 ± 12.7	75.3 ± 11.0
NOnAOffN	C=1e-02	79.6 ± 5.4	89.0 ± 11.0	68.6 ± 10.3	76.5 ± 5.1

Primeras tres filas: kernel Gaussiano . Ultimas tres filas: kernel lineal.

*Columna con los hiperparámetros óptimos.

Obsérvese que hay una mejora en ambos modelos en comparación con los basados en VGG-8 y ResNet-7 sin aplicar la función de triple perdida. En el primer caso la mejora es de alrededor del 5.1 % (del 67.6 % a 72.7 %) y en el segundo caso es de alrededor del 3.6 % (del 78.8 % a 82.4 %). No sólo es interesante destacar la mejora lograda al utilizar la función de triple pérdida, sino también observar que el mejor resultado obtenido con el modelo Triple-ResNet7 es competitivo en comparación con la mejor precisión obtenida anteriormente con el modelo Freeze 75. Aunque los aciertos en el segundo sigue siendo un 4.9 % superior a la del primero, Freeze 75 requiere 17.815.520 parámetros más para ser optimizado que Triplet-ResNet7, lo que podría indicar una mejor capacidad de generalización. Se necesitan más experimentos con datos adicionales para validar esta hipótesis.

Con el objetivo de mostrar de forma más compacta la mejora conseguida cuando se utiliza la función de pérdida de trillizos, la Figura 6.10a muestra las curvas ROC correspondientes a los modelos VGG-8 y Triple-VGG8, y la Figura 6.10b muestra las curvas ROC para el ResNet-7 y el Triple-ResNet7. Estas dos figuras también dejan clara la ventaja de aplicar la función de triple perdida para mejorar los aciertos de la discriminación entre pacientes con EP y sujetos sanos.



(a) Curva ROC de los modelos VGG-8 y Triplet-VGG8 utilizando la secuencia NOnAOffN.

(b) Curva ROC de los modelos ResNet-7 y Triplet-ResNet7 utilizando la secuencia NOnAOffN.

Figura 6.10. Impacto de la función de triple perdida en el desempeño de los modelos reducidos.

6.4. Experimento 4: Nivel de clasificación de estado neurológico

Dados los buenos resultados obtenidos con los experimentos presentados anteriormente, especialmente con los modelos Freeze 75 y Triple-ResNet7, con precisiones del 87.3 % y 82.4 %, respectivamente, en la discriminación automática entre pacientes con EP y sujetos sanos, queremos evaluar en este apartado la idoneidad de dichos modelos para discriminar tres grados diferentes de deterioro: leve (EP-1), intermedio (EP-2) y grave (EP-3). Estos tres grupos se definen considerando las puntuaciones del MDS-UPDRS-III proporcionadas por el neurólogo experto. El grupo leve incluye pacientes con puntuaciones en el rango de 0 a 23, el grupo intermedio se define para pacientes con puntuaciones entre 23 y 33, y el grupo grave para pacientes con puntuaciones superiores a 33. La Figura 6.11 muestra la distribución de las puntuaciones de MDS-UPDRS-III para los tres grupos de pacientes.

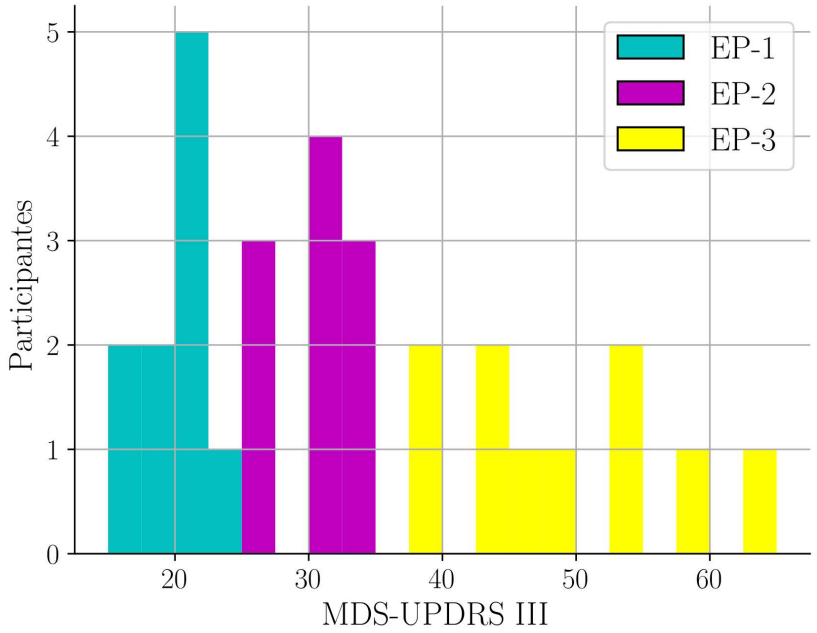


Figura 6.11. Histograma del estado neurológico de los pacientes. Estado inicial (cian), estado intermedio (magenta) y estado severo (amarillo).

Los experimentos de clasificación de tres clases se realizan teniendo en cuenta los vectores de características extraídos con el modelo Freeze 75 en la secuencia NOnAOffN, y el modelo Triple-ResNet7 en la secuencia NOnA. La optimización de los hiperparámetros se realiza como se indica en la Sección 4.5. Las matrices de confusión y los resultados obtenidos con los modelos Freeze 75 y Triplet-ResNet7 se muestran en la Tabla 6.13 y en la Tabla 6.14, respectivamente. Obsérvese que cada tabla incluye resultados con los clasificadores SVM y RF. Los valores de acierto, la puntuación F1 y el índice κ se incluyen en la parte inferior de cada tabla.

Tabla 6.13. Métricas de desempeño y matriz de confusión de la SVM (izquierda) y del RF (derecha) en la clasificación de los estados neurológicos de los pacientes usando el modelo Freeze 75 con la secuencia NOnAOffN.

	SVM: C=1e-03			RF: N_T=800 ; D=25 ; S_m=10		
	PD-1	PD-2	PD-3	PD-1	PD-2	PD-3
PD-1	45.80	35.30	18.90	45.25	29.24	25.50
PD-2	33.47	42.26	24.27	33.43	34.48	32.09
PD-3	15.45	39.49	45.06	23.69	34.08	42.23

SVM: Acc= 44 %, F1= 0.45, κ= 0.17.

RF: Acc= 41 %, F1= 0.41, κ= 0.11.

Tabla 6.14. Métricas de desempeño y matriz de confusión de la SVM (izquierda) y del RF (derecha) en la clasificación de los estados neurológicos de los pacientes usando el modelo Triplet-ResNet7 con la secuencia NOnA.

	SVM: C=1e-03			RF: N_T=300 ; D=5 ; S_m=2		
	PD-1	PD-2	PD-3	PD-1	PD-2	PD-3
PD-1	30.07	39.60	30.33	59.13	12.27	28.60
PD-2	34.60	20.53	44.87	32.80	23.87	43.33
PD-3	10.28	18.62	71.10	27.10	28.00	44.90

SVM: Acc= 40 %, F1= 0.38, κ= 0.11.

RF: Acc= 43 %, F1= 0.41, κ= 0.14.

Estos resultados muestran que el modelo Freeze 75 es mejor que el Triplet-ResNet7. Hay una diferencia de 4 % de puntos en el acierto, y 0.07 en la F1-score cuando se utiliza el SVM. Al comparar los dos modelos con el clasificador de RF, los resultados son similares a los obtenidos con el SVM pero ligeramente mejores con el modelo Triplet-ResNet7. Al igual que en los experimentos de clasificación binaria, el espacio 2D producido con PCA se crea con el objetivo de proporcionar al lector una idea más intuitiva de la separa-

ción lograda con el espacio de representación. En este caso el espacio reducido se crea con vectores de características del modelo Freeze 75 extraídos de la secuencia NOnAOffN. La Figura 6.12 muestra el espacio proyectado, en el que se puede observar una separación relativamente clara entre los pacientes leves y los graves. En medio de este a grupos están las muestras del grupo intermedio, que no están clasificadas con precisión pero aparecen claramente en medio.

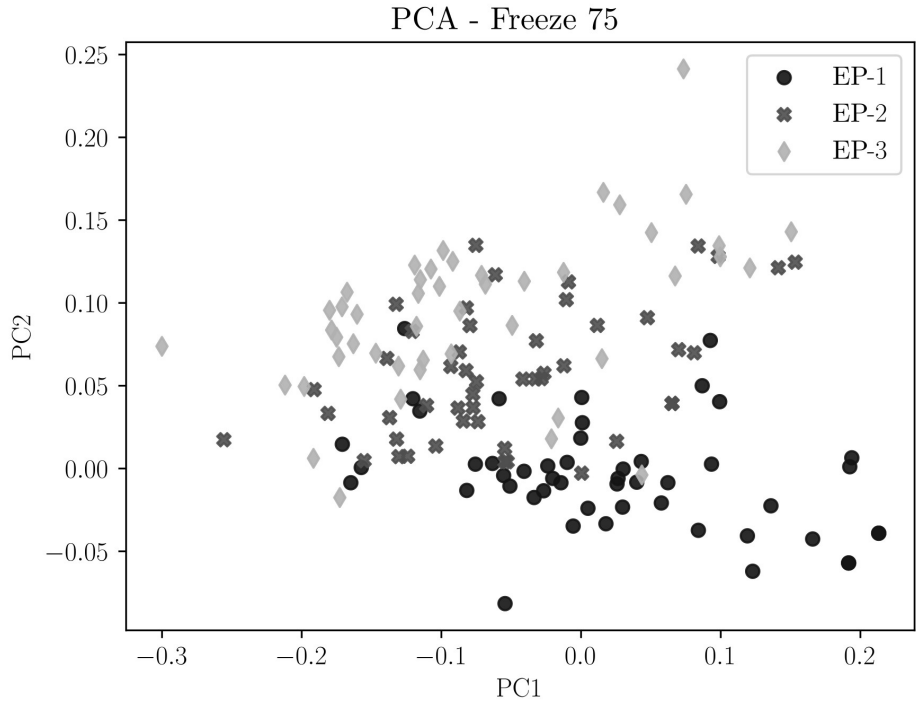


Figura 6.12. Espacio de representación en 2D creado con PCA sobre los vectores de características extraídos del modelo Freeze 75 de los 3 grupos de pacientes con diferentes grados de deterioro.

Capítulo 7

Conclusiones y trabajo futuro

Los resultados encontrados en este trabajo demuestran la capacidad de obtener mejoras en la clasificación entre pacientes y personas sanas creando nuevos dominios de representación, basándonos en la transferencia de conocimiento y el uso de métricas de aprendizaje por similitud en arquitecturas creadas para el reconocimiento de rostros y el reconocimiento de unidades de acción facial. El desempeño de los modelos aumenta debido a que el nuevo espacio embebido generado muestra una mejora en la separación entre las clases. Las conclusiones de este trabajo están sujetas a las limitaciones impuestas por los datos disponibles en la base de datos FacePark.

El modelo de reconocimiento de rostros aporta información de características faciales, en estas se obtuvo un desempeño en aciertos en clasificación del 72.9 % al momento de analizar características individuales como rostros neutrales, transiciones y pico de expresión. Por otra parte, el análisis de secuencias nos provee un cambio en el desempeño de los clasificadores, aumentando aproximadamente en 5 % la cantidad de aciertos para las secuencias NOnA, AOffN y NOnAOffN. Este aumento en el desempeño se debe a que al tratar de procesar información de los diferentes eventos obtenemos cambios temporales que influyen en las decisión del clasificador de pacientes y sujetos sanas.

Con base en esto, el cambio de dominio de reconocimiento de rostro a un dominio donde se pudieran explotar las expresiones faciales llevaría a obtener mejoras en el desempeño de los clasificadores. Esto se analiza desde la perspectiva de que en los sistemas de reconocimiento de rostros, las expresiones faciales de los usuarios no deben de interferir en gran medida en los resultados y es por eso que estos cambios de expresiones de los rostros

no se buscan explotar en estas redes. La metodología planteada de transferencia de conocimiento y el uso de la base de datos EmotioNet mejoró en gran medida los resultados, aumentando los aciertos en la clasificación a un 87.3 % en comparación con el 78,4 % del resultado original para la secuencia NOnAOffN. Por otro lado, el uso de modelos VGG-8 y ResNet-7 con poca cantidad de parámetros muestra que los resultados alcanzados por estos modelos no fueron tan altos, obteniendo aciertos del 67.6 % para el modelo VGG-8, y del 78.8 % para el modelo ResNet-7.

Para la creación de vectores embebidos, tenemos que observar los resultados obtenidos con los modelos Triplet-Freeze los cuales tuvieron una reducción en su desempeño del 87.3 % al 86.1 % para Triplet 75 y del 83.1 % al 80.7 % para Triplet 50. Adicionalmente hay que mencionar que el entrenamiento del modelo VGGFace2 para el reconocimiento de los rostros es realizado con la función de triple perdida. Una hipótesis que se evidencia en este trabajo es que al utilizar nuevamente la función de triple perdida en el entrenamiento podemos obtener dos posibles resultados para la base de datos FacePark: (1) Que el modelo no tenga mejoras significativa al realizar el entrenamiento, obteniendo valores de desempeño con muy poca varianza respecto a los originales ó (2) La cantidad de información que provee los datos puede deteriorar el desempeño de los modelos. Adicionalmente mostramos que el uso de la función de triple perdidas al ser utilizada en modelos pequeños como VGG-8 y ResNet-7, aumentan el desempeño general de cada modelo. Los resultados muestran mejoras del 3.6 % para el modelo Triplet-ResNet7, y aumentos del 5.1 % para el modelo Triplet-VGG8. Algo que resaltar en este trabajo es el desempeño general de las secuencias, donde el mejor de los resultados fue obtenido al utilizar la secuencia completa NOnAOffN, pero al momento de analizar las secuencias NOnA y la secuencia AOffN, la secuencia NOnA se desempeña mejor que la secuencia AOffN, lo que nos dice que existe mas información discriminante al iniciar una expresión facial que al terminarla acompañando a las hipótesis planteadas en [48], [49].

El trabajo futuro deberá incluir un análisis de secuencia temporales con modelos adicionales como el uso de redes neuronales recurrentes de larga memoria a corto plazo o el uso de arquitecturas de redes neuronales convolucionales de 3-Dimensiones, combinando el análisis espacio-temporal en todas sus capas. Además de incluir posibles análisis multimodales con la creación de tensores que combinen otras características como la voz, el análisis de texto, entre otros. Igualmente la metodología de este trabajo podría imple-

mentarse en aplicaciones móviles, para la ayuda a la terapia de pacientes con EP, dándole el uso a sesiones de grabación diarias, lo cual permitiría dar a los pacientes realimentación sobre su capacidad de producir determinadas expresiones faciales.

Índice de figuras

Bibliografía

[1] M. de Rijk, L. Launer, K. Berger, M. Breteler, J. Dartigues, M. Baldereschi, L. Fratiglioni, A. Lobo, J. Martinez-Lage, C. Trenkwalder y A. Hofman, "Prevalence of Parkinson's disease in Europe: A collaborative study of population-based cohorts. Neurologic Diseases in the Elderly Research Group", *Neurology*, vol. 54, S21-3, 2000.

[2] L. O. Ramig, C. Fox y S. Sapir, "Speech treatment for Parkinson's disease", *Expert Review of Neurotherapeutics*, vol. 8, n.º 2, págs. 297-309, 2008.

[3] E. Dorsey, R. Constantinescu, J. Thompson, K. Biglan, R. Holloway, K. Kieburtz, F. Marshall, B. Ravina, G. Schifitto, A. Siderowf y col., "Projected number of people with Parkinson disease in the most populous nations, 2005 through 2030", *Neurology*, vol. 68, n.º 5, págs. 384-386, 2007.

[4] J. L. Sanchez, O. Buritica, D. Pineda, C. Santiago Uribe y L. Guillermo Palacio, "Prevalence of Parkinson's disease and parkinsonism in a Colombian population using the capture-recapture method", *international Journal of Neuroscience*, vol. 114, n.º 2, págs. 175-182, 2004.

[5] M. Bologna, G. Fabbrini, L. Marsili, G. Defazio, P. D. Thompson y A. Berardelli, "Facial bradykinesia", *J Neurol Neurosurg Psychiatry*, vol. 84, n.º 6, págs. 681-685, 2013.

[6] D. G. Theodoros, G. Constantinescu, T. G. Russell, E. C. Ward, S. J. Wilson y R. Wootton, "Treating the speech disorder in Parkinson's disease online", *Journal of Telemedicine and Telecare*, vol. 12, n.º 3_suppl, págs. 88-91, 2006.

[7] C. G. Goetz, B. C. Tilley, S. R. Shaftman, G. T. Stebbins, S. Fahn, P. Martinez-Martin y col., "Movement Disorder Society-sponsored revision of the Unified Parkinson's Disease Rating Scale (MDS-UPDRS): scale presentation and clinimetric testing results", *Movement disorders: official journal of the Movement Disorder Society*, vol. 23, n.º 15, págs. 2129-2170, 2008.

[8] A. Krizhevsky, I. Sutskever y G. E. Hinton, "Imagenet classification with deep convolutional neural networks", en *Advances in neural information processing systems*, 2012, págs. 1097-1105.

[9] O. M. Parkhi, A. Vedaldi, A. Zisserman y col., "Deep face recognition.", en *bmvc*, vol. 1, 2015, pág. 6.

[10] F. Schroff, D. Kalenichenko y J. Philbin, "Facenet: A unified embedding for face recognition and clustering", en *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, págs. 815-823.

[11] H. Kaya, F. Gürpınar y A. A. Salah, "Video-based emotion recognition in the wild using deep transfer learning and score fusion", *Image and Vision Computing*, vol. 65, págs. 66-75, 2017.

[12] I. J. Goodfellow, D. Erhan, P. L. Carrier, A. Courville, M. Mirza, B. Hamner, W. Cukierski, Y. Tang, D. Thaler, D.-H. Lee y col., "Challenges in representation learning: A report on three machine learning contests", en *International Conference on Neural Information Processing*, Springer, 2013, págs. 117-124.

[13] R. Breuer y R. Kimmel, "A deep learning perspective on the origin of facial expressions", *arXiv preprint arXiv:1705.01842*, 2017.

[14] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar e I. Matthews, "The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression", en *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, IEEE, 2010, págs. 94-101.

[15] A. Mourão y J. Magalhães, "Competitive affective gaming: winning with a smile", en *Proceedings of the 21st ACM international conference on Multimedia*, ACM, 2013, págs. 83-92.

[16] R. Ekman, *What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS)*. Oxford University Press, USA, 1997.

[17] A. Sajjanhar, Z. Wu y Q. Wen, "Deep learning models for facial expression recognition", en *2018 Digital Image Computing: Techniques and Applications (DICTA)*, IEEE, 2018, págs. 1-6.

[18] M. J. Lyons, S. Akamatsu, M. Kamachi, J. Gyoba y J. Budynek, "The Japanese female facial expression (JAFFE) database", en *Proceedings of third international conference on automatic face and gesture recognition*, 1998, págs. 14-16.

[19] N. C. Ebner, M. Riediger y U. Lindenberger, "FACES—A database of facial expressions in young, middle-aged, and older women and men: Development and validation", *Behavior research methods*, vol. 42, n.º 1, págs. 351-362, 2010.

[20] S. Cheng y G. Zhou, "Facial Expression Recognition Method Based on Improved VGG Convolutional Neural Network", *International Journal of Pattern Recognition and Artificial Intelligence*, pág. 2 056 003, 2019.

[21] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li y L. Fei-Fei, "Imagenet: A large-scale hierarchical image database", en *2009 IEEE conference on computer vision and pattern recognition*, Ieee, 2009, págs. 248-255.

[22] S. Ouellet, "Real-time emotion recognition for gaming using deep convolutional network features", *arXiv preprint arXiv:1408.3750*, 2014.

[23] S. Xie, H. Hu e Y. Wu, "Deep multi-path convolutional neural network joint with salient region attention for facial expression recognition", *Pattern Recognition*, vol. 92, págs. 177-191, 2019.

[24] B. Sonawane y P. Sharma, "Review of automated emotion-based quantification of facial expression in Parkinson's patients", *The Visual Computer*, jun. de 2020, ISSN: 1432-2315. DOI: 10.1007/s00371-020-01859-9.

[25] G. Simons, M. C. S. Pasqualini, V. Reddy y J. Wood, "Emotional and nonemotional facial expressions in people with Parkinson's disease", *Journal of the International Neuropsychological Society*, vol. 10, n.º 4, págs. 521-535, 2004.

[26] D. Bowers, K. Miller, W. Bosch, D. Gokcay, O. Pedraza, U. Springer y M. Okun, "Faces of emotion in Parkinson's disease: Micro-expressivity and bradykinesia during voluntary facial expressions", *Journal of the International Neuropsychological Society*, vol. 12, págs. 765-773, 2006.

[27] R. Almutiry, S. Couth, E. Poliakoff, S. Kotz, M. Silverdale y T. Cootes, "Facial Behaviour Analysis in Parkinson's Disease", *Lecture Notes in Computer Science*, vol. 9805, págs. 329-339, 2016.

[28] S. Gunnery, E. Naumova, M. Saint-Hilaire y L. Tickle-Degnen, "Mapping spontaneous facial expression in people with Parkinson's disease: A multiple case study design", *Cogent Psychology*, vol. 4, págs. 1-15, 2017.

[29] E. Friesen y P. Ekman, *Facial Action Coding System: A technique for the measurement of facial movement.* 1978.

[30] P. Ekman, W. Friesen y J. Hager, *The facial action coding system.* Salt Lake City, UT: A Human Face, 2002.

[31] A. Bandini, S. Orlandi, H. J. Escalante, F. Giovannelli, M. Cincotta, C. A. Reyes-Garcia, P. Vanni, G. Zaccara y C. Manfredi, "Analysis of facial expressions in Parkinson's disease through video-based automatic methods", *Journal of neuroscience methods*, vol. 281, págs. 7-20, 2017.

[32] J. Kang, D. Derva, D.-Y. Kwon y C. Wallraven, "Voluntary and spontaneous facial mimicry toward other's emotional expression in patients with Parkinson's disease", *PloS one*, vol. 14, n.º 4, 2019.

[33] Y.-J. Lim, Y.-G. Ko, H.-C. Shin e Y. Cho, "Prevalence and correlates of complete mental health in the South Korean adult population", en *Mental well-being*, Springer, 2013, págs. 91-109.

[34] A. Grammatikopoulou, N. Grammalidis, S. Bostantjopoulou y Z. Katsarou, "Detecting hypomimia symptoms by selfie photo analysis: for early Parkinson disease detection", en *Proceedings of the 12th ACM International Conference on PErvasive Technologies Related to Assistive Environments*, 2019, págs. 517-522.

[35] S. Du, Y. Tao y A. M. Martinez, "Compound facial expressions of emotion", *Proceedings of the National Academy of Sciences*, vol. 111, n.º 15, E1454-E1462, 2014.

[36] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position", *Biological cybernetics*, vol. 36, n.º 4, págs. 193-202, 1980.

[37] Y. LeCun, Y. Bengio y G. Hinton, "Deep learning", *nature*, vol. 521, n.º 7553, pág. 436, 2015.

[38] K. He, X. Zhang, S. Ren y J. Sun, "Deep residual learning for image recognition", en *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, págs. 770-778.

[39] K. Q. Weinberger y L. K. Saul, "Distance metric learning for large margin nearest neighbor classification", *Journal of Machine Learning Research*, vol. 10, n.º Feb, págs. 207-244, 2009.

[40] M. Slater, "Lagrange multipliers revisited", en *Traces and emergence of nonlinear programming*, Springer, 2014, págs. 293-306.

[41] W. Karush, "Minima of functions of several variables with inequalities as side constraints", *M. Sc. Dissertation. Dept. of Mathematics, Univ. of Chicago*, 1939.

[42] H. W. Kuhn y A. W. Tucker, "Nonlinear programming", en *Traces and emergence of nonlinear programming*, Springer, 2014, págs. 247-258.

[43] V. Vapnik, *The nature of statistical learning theory*. Springer science & business media, 1995.

[44] Q. Cao, L. Shen, W. Xie, O. M. Parkhi y A. Zisserman, "Vggface2: A dataset for recognising faces across pose and age", en *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, IEEE, 2018, págs. 67-74.

[45] C. Fabian Benitez-Quiroz, R. Srinivasan y A. M. Martinez, "Emotionet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild", en *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, págs. 5562-5570.

[46] M. Yan, M. Zhao, Z. Xu, Q. Zhang, G. Wang y Z. Su, "Vargfacenet: An efficient variable group convolutional neural network for lightweight face recognition", en *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019.

[47] R. Ranjan, S. Sankaranarayanan, A. Bansal, N. Bodla, J.-C. Chen, V. M. Patel, C. D. Castillo y R. Chellappa, "Deep learning for understanding faces: Machines may be just as good, or better, than humans", *IEEE Signal Processing Magazine*, vol. 35, n.º 1, págs. 66-83, 2018.

[48] J. Orozco-Arroyave, *Analysis of speech of people with Parkinson's disease*. Logos-Verlag, Berlin, 2016.

[49] J. Vásquez-Correa, T. Arias-Vergara, J. Orozco-Arroyave, B. Eskofier, J. Klucken y E. Nöth, "Multimodal assessment of Parkinson's disease: a deep learning approach", *IEEE Journal of Biomedical and Health Informatics*, vol. 23, n.º 4, págs. 1618-1630, 2019.