

**EXPLICABILIDAD APLICADA A MODELOS DE
RECONOCIMIENTO DE EMOCIONES EN
IMÁGENES**

**EXPLAINABILITY APPLIED TO MODELS OF
EMOTION RECOGNITION IN IMAGES**



**TRABAJO FIN DE GRADO
CURSO 2023-2024**

**AUTOR
JAVIER GARCÍA VIANA**

**DIRECTORES
BELÉN DÍAZ AGUDO Y JUAN ANTONIO RECIO GARCÍA**

**GRADO EN INGENIERÍA INFORMÁTICA
FACULTAD DE INFORMÁTICA
UNIVERSIDAD COMPLUTENSE DE MADRID**

**EXPLICABILIDAD APLICADA A MODELOS DE
RECONOCIMIENTO DE EMOCIONES EN
IMÁGENES**

**EXPLAINABILITY APPLIED TO MODELS OF
EMOTION RECOGNITION IN IMAGES**

TRABAJO DE FIN DE GRADO EN INGENIERÍA INFORMÁTICA

**AUTOR
JAVIER GARCÍA VIANA**

**DIRECTORES
BELÉN DÍAZ AGUDO Y JUAN ANTONIO RECIO GARCÍA**

CONVOCATORIA: JUNIO 2024

**GRADO EN INGENIERÍA INFORMÁTICA
FACULTAD DE INFORMÁTICA
UNIVERSIDAD COMPLUTENSE DE MADRID**

24 DE MAYO DE 2024

DEDICATORIA

A mis padres, abuelos y hermanos.

AGRADECIMIENTOS

A mis compañeros de clase, por los momentos vividos y por ayudarme siempre.

RESUMEN

Con la popularización y el rápido crecimiento de la Inteligencia Artificial, algunos modelos de aprendizaje automático, en especial de aprendizaje supervisado, están siendo usados para sistematizar el proceso de detección de emociones en imágenes. Debido a que el enfoque es relativamente nuevo, en este trabajo se pretende explorar las posibilidades y analizar la precisión de estos modelos para finalmente determinar qué combinación de características conlleva a una mejor precisión por parte de los modelos. Aparte de esto, el problema al que se enfrentan a menudo aquellas personas que tratan con estos modelos es, a no ser capaces de explicar la atribución de emociones distintas para imágenes de entrada parecidas. Por este motivo, en este trabajo se pretende que los modelos muestren, mediante algoritmos de explicabilidad IA, las regiones de las imágenes que son decisivas en las predicciones de emociones para poder así explicar las decisiones tomadas por los modelos y dar a conocer los beneficios de aplicar algoritmos de explicabilidad.

Palabras clave

Explicabilidad, aprendizaje supervisado, imágenes, detección, emociones, interpretación de resultados.

ABSTRACT

With the popularization and fast development of Artificial Intelligence, some models of machine learning, specifically those oriented to supervised learning, are being used to systematize the process of emotion detection in images. Since this approach is relatively new, this thesis aims to explore the possibilities and analyse the accuracy of these models to determine which combination of characteristics leads the models to obtain the best accuracy. Apart from this, the problem often faced by those who deal with this kind of models is not being able to explain the attribution of different emotions for similar input images. For this reason, in this thesis is intended to show, using AI explainability algorithms, the regions of the images that are decisive in the emotion prediction to finally explain the decisions made by the models and show off the benefits of the appliance of AI explainability algorithms.

Keywords

Explainability, supervised learning, images, detection, emotions, result interpretation

ÍNDICE DE CONTENIDOS

Capítulo 1 - Introducción.....	1
1.1 Motivación	1
1.2 Objetivos.....	1
1.3 Plan de trabajo	2
1.3.1 Análisis del estado de la cuestión	2
1.3.2 Desarrollo y experimentación	2
1.3.3 Documentación.....	3
Capítulo 2 - Estado de la cuestión.....	5
2.1 Aprendizaje supervisado.....	5
2.1.1 Redes neuronales convolucionales.....	7
2.2 Inteligencia Artificial Explicable (XAI).....	13
2.3 Aplicaciones prácticas de reconocimiento de emociones en imágenes	16
2.4 Trabajos similares.....	18
Capítulo 3 - Procesamiento de los datos y estructura de los modelos	21
3.1 Carga del conjunto de imágenes	21
3.2 Adaptación del formato de las imágenes.....	22
3.2.1 Tamaño	22
3.2.2 Color.....	25
3.3 Estructura de los modelos	25
3.3.1 Modelos MLP	25
3.3.2 Modelos CNN	26
3.4 Elección del conjunto de datos de entrenamiento y prueba	28

3.5 Procesamiento de los datos para los modelos	28
Capítulo 4 - Clasificación de emociones en imágenes.....	30
4.1 Modelos MLP	32
4.1.1 Mejores modelos MLP obtenidos	35
4.2 Modelos CNN	37
4.2.1 Mejores modelos CNN obtenidos	41
Capítulo 5 - Explicabilidad.....	45
5.1 AnchorImage	45
5.2 IntegratedGradients	59
5.3 GradientSimilarity	69
Capítulo 6 - Conclusiones y trabajo futuro.....	80
6.1 Conclusiones	80
6.2 Trabajo futuro	82
Introduction.....	85
Conclusions and future work	89
Bibliografía.....	93
Abreviaturas.....	96

ÍNDICE DE ILUSTRACIONES

Ilustración 1.- Diagrama de Gantt del plan de trabajo	3
Ilustración 2.- Proceso para resolver un problema de aprendizaje automático	6
Ilustración 3.- Primera iteración de aplicar una convolución con paso 1 a una imagen en escala de grises.....	9
Ilustración 4.- Aplicación de la convolución en la primera iteración a una imagen a color	9

Ilustración 5.- Resultado de aplicar un filtro (3x3x1) con paso (1) a una imagen en escala de gris (6x6x1) con relleno (1)	10
Ilustración 6.- Resultado de aplicar max pooling de (2x2x1) con paso (2) a una imagen (4x4x1)	11
Ilustración 7.- Arquitectura de LeNet-5.....	12
Ilustración 8.- Arquitectura de AlexNet.....	12
Ilustración 9.- Arquitectura de ZFNet.....	13
Ilustración 10.- Mapa de algunas librerías de explicabilidad	15
Ilustración 11.- Resoluciones de las imágenes del conjunto de datos inicial.....	23
Ilustración 12.- Visualización de las imágenes del conjunto de datos inicial tras redimensionar. De izquierda a derecha y de arriba a abajo: (i)(1152x809) (ii)(768x539) (iii)(384x269) (iv)(96x96) (v)(48x48) (vi)(24x24)	24
Ilustración 13.- Visualización de las imágenes del conjunto de datos final tras redimensionar. De izquierda a derecha: (i)(96x96) (ii)(48x48) (iii)(24x24)	24
Ilustración 14.- Estructura de los modelos MLP usados	26
Ilustración 15.- Tiempos y rendimientos de los modelos MLP del conjunto de datos inicial tras el proceso de entrenamiento.....	33
Ilustración 16.- Tiempos y rendimientos de los modelos MLP de clasificación de tres tipos de emociones tras el proceso de entrenamiento	34
Ilustración 17.- Matriz de confusión del mejor modelo MLP del conjunto de datos inicial	36
Ilustración 18.- Matriz de confusión del mejor modelo MLP de clasificación de tres tipos de emociones	37
Ilustración 19.- Tiempos y rendimientos de los modelos CNN del conjunto de datos inicial tras el proceso de entrenamiento.....	39
Ilustración 20.- Tiempos y rendimientos de los modelos CNN de clasificación de tres tipos de emociones tras el proceso de entrenamiento	40

Ilustración 21.- Tiempos y rendimientos de los modelos CNN de clasificación de siete tipos de emociones tras el proceso de entrenamiento	41
Ilustración 22.- Matriz de confusión del mejor modelo CNN del conjunto de datos inicial	42
Ilustración 23.- Matriz de confusión del mejor modelo CNN de clasificación de tres tipos de emociones	43
Ilustración 24.- Matriz de confusión del mejor modelo CNN de clasificación de siete tipos de emociones	44
Ilustración 25.- Gantt diagram of the work plan	87

ÍNDICE DE TABLAS

Tabla 1.- Compatibilidad de librerías de explicabilidad con frameworks de ML y tipos de datos admitidos	15
Tabla 2.- Características, rendimientos y enlaces de modelos usados en algunos trabajos similares	18
Tabla 3.- Capas y tamaños generales de los modelos MLP usados	25
Tabla 4.- Estructura general de los modelos de redes neuronales convolucionales usados	27
Tabla 5.- Comprobación del equilibrado del conjunto de datos inicial	30
Tabla 6.- Comprobación del equilibrado del subconjunto extraído del conjunto de datos final.....	31
Tabla 7.- Comprobación del equilibrado del conjunto de datos final.....	32
Tabla 8.- Explicabilidad por el método de AnchorImage de imágenes correspondientes a la clase “happy” para el problema de clasificación de tres emociones	46
Tabla 9.- Explicabilidad por el método de AnchorImage de imágenes correspondientes a la clase “angry” para el problema de clasificación de tres emociones	48
Tabla 10.- Explicabilidad por el método de AnchorImage de imágenes correspondientes a la clase “sad” para el problema de clasificación de tres emociones	49

Tabla 11.- Explicabilidad por el método de AnchorImage de imágenes correspondientes a la clase “angry” para el problema de clasificación de siete emociones	50
Tabla 12.- Explicabilidad por el método de AnchorImage de imágenes correspondientes a la clase “disgust” para el problema de clasificación de siete emociones.....	52
Tabla 13.- Explicabilidad por el método de AnchorImage de imágenes correspondientes a la clase “fear” para el problema de clasificación de siete emociones	53
Tabla 14.- Explicabilidad por el método de AnchorImage de imágenes correspondientes a la clase “happy” para el problema de clasificación de siete emociones	54
Tabla 15.- Explicabilidad por el método de AnchorImage de imágenes correspondientes a la clase “neutral” para el problema de clasificación de siete emociones	55
Tabla 16.- Explicabilidad por el método de AnchorImage de imágenes correspondientes a la clase “sad” para el problema de clasificación de siete emociones	57
Tabla 17.- Explicabilidad por el método de AnchorImage de imágenes correspondientes a la clase “surprise” para el problema de clasificación de siete emociones.....	58
Tabla 18.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase “happy” para el problema de clasificación de tres emociones.....	60
Tabla 19.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase “angry” para el problema de clasificación de tres emociones.....	61
Tabla 20.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase “sad” para el problema de clasificación de tres emociones.....	62
Tabla 21.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase “angry” para el problema de clasificación de siete emociones.....	63
Tabla 22.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase “disgust” para el problema de clasificación de siete emociones.....	64

Tabla 23.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase "fear" para el problema de clasificación de siete emociones.....	64
Tabla 24.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase "happy" para el problema de clasificación de siete emociones.....	65
Tabla 25.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase "neutral" para el problema de clasificación de siete emociones.....	66
Tabla 26.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase "sad" para el problema de clasificación de siete emociones.....	67
Tabla 27.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase "surprise" para el problema de clasificación de siete emociones.....	68
Tabla 28.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase "happy" para el problema de clasificación de tres emociones.....	69
Tabla 29.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase "angry" para el problema de clasificación de tres emociones.....	71
Tabla 30.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase "sad" para el problema de clasificación de tres emociones.....	71
Tabla 31.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase "angry" para el problema de clasificación de siete emociones.....	72

Tabla 32.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase “disgust” para el problema de clasificación de siete emociones.....	73
Tabla 33.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase “fear” para el problema de clasificación de siete emociones.....	74
Tabla 34.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase “happy” para el problema de clasificación de siete emociones.....	75
Tabla 35.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase “neutral” para el problema de clasificación de siete emociones.....	76
Tabla 36.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase “sad” para el problema de clasificación de siete emociones.....	77
Tabla 37.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase “surprise” para el problema de clasificación de siete emociones.....	78

Capítulo 1 - Introducción

1.1 Motivación

Desde el punto de vista computacional, una imagen es una matriz de números que representan tonos de colores o una escala de grises. A partir de la representación que los píxeles ofrecen, en ciertos casos es posible extraer relaciones significativas presentes entre un grupo de ellos, que hacen que varias imágenes puedan estar relacionadas entre sí, ya que hay patrones recurrentes compartidos. Estos patrones pueden ser detectados mediante modelos de aprendizaje supervisado del campo del aprendizaje automático (AA) de la inteligencia artificial. Existen diversos tipos de modelos que, tras buscar patrones sobre un conjunto de imágenes de entrada y sus respectivas categorías durante el proceso de entrenamiento, consiguen clasificar un conjunto de imágenes entre las distintas categorías establecidas. Ciertamente, muchas veces estos patrones y características detectadas son invisibles a los ojos y pasan desapercibidos por aquellas personas que habitualmente usan los modelos para realizar una clasificación de un conjunto de imágenes en varias categorías, ya que muchos de los modelos más usados se limitan a ofrecer, dada una imagen de entrada, la predicción de la categoría atribuida, sin mostrar las regiones de la imagen por las que ha decidido asociarla junto a imágenes de una determinada categoría u otra.

1.2 Objetivos

Se pretende realizar un estudio sobre los distintos modelos existentes de clasificación aplicados a imágenes, con el objetivo final de determinar qué modelo y qué parámetros son los ideales para resolver un problema de clasificación de emociones presentes en caras. Sin dejar atrás este objetivo, también se pretende realizar un estudio de distintos algoritmos existentes de explicabilidad aplicadas a imágenes, con el objetivo final de poder mostrar en la imagen original las zonas seleccionadas por el mejor modelo de caja negra obtenido para verificar la correcta detección de las diversas emociones presentes en las imágenes y, por tanto, la correcta clasificación de las imágenes en las categorías existentes.

1.3 Plan de trabajo

En este apartado se va a mostrar el desarrollo del trabajo. A continuación, se detallan las etapas del trabajo realizado:

1.3.1 Análisis del estado de la cuestión

Durante esta etapa se ha investigado acerca de diversos temas relacionados con el desarrollo del trabajo, que son aquellos relacionados con el aprendizaje automático de la Inteligencia Artificial, más concretamente, el aprendizaje supervisado y las redes neuronales convolucionales. También se ha realizado un estudio de la Inteligencia Artificial Explicable aplicada a imágenes y de aplicaciones prácticas del uso de modelos de detección de emociones en caras.

1.3.2 Desarrollo y experimentación

En este subapartado se va a tratar de los conjuntos de datos usados y de los modelos y características con las que se ha experimentado. En primer lugar, para realizar una aproximación inicial al trabajo, se usa un conjunto de datos pequeño con imágenes de personas que representan tres tipos de emociones. Tras esto, para realizar el trabajo se usa el conjunto de datos llamado FER2013 con imágenes de siete tipos de emociones de tamaño 48x48 en escala de grises. Este dataset contiene muchas más imágenes que el conjunto de datos anterior. Las primeras pruebas de los modelos y de la explicabilidad se han realizado sobre un subconjunto de imágenes del conjunto de datos final, concretamente sobre tres tipos de emociones. Finalmente, se han realizado las pruebas finales con el conjunto de datos completo. Tras realizar las pruebas y fijar el modelo que mejores resultados ofrece, se lleva a cabo la aplicación de varios algoritmos de explicabilidad de la librería de ALIBI para mostrar las regiones de la imagen elegidas para representar la emoción predicha.

Todo el trabajo realizado se puede consultar en el siguiente directorio de GitHub:
<https://github.com/javieg25/Explicabilidad-aplicada-a-modelos-de-reconocimiento-de-emociones-en-imagenes.git>

1.3.3 Documentación

A la vez que se expandían los conocimientos en la etapa de análisis del estado de la cuestión, y a la vez que se realizaba el desarrollo y experimentación con los modelos y algoritmos de explicabilidad, se ha ido rellenando la memoria, incluyendo los datos usados y las conclusiones a las que se han llegado tras realizar el estudio de los resultados obtenidos.

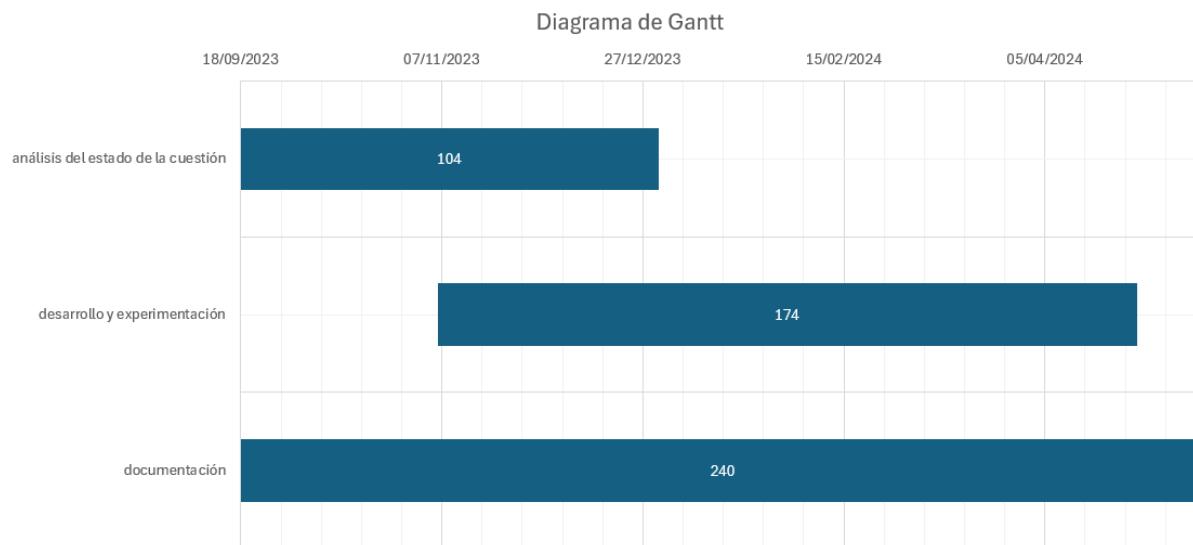


Ilustración 1.- Diagrama de Gantt del plan de trabajo

Capítulo 2 - Estado de la cuestión

En este capítulo se va a hacer una visión acerca del aprendizaje automático, en concreto, del aprendizaje supervisado y de las redes neuronales convolucionales. También se hará un repaso de la inteligencia artificial explicable y de las diferentes aplicaciones prácticas del reconocimiento de emociones en imágenes.

2.1 Aprendizaje supervisado

El aprendizaje automático (AA)¹ es una rama de la Inteligencia Artificial que tiene por objetivo aprender mediante la extracción de características relevantes de los datos proporcionados a los modelos para después hacer predicciones sobre nuevos datos de entrada [1]. Este proceso intenta imitar la forma humana de aprender para obtener resultados más precisos. Para proceder a resolver un problema de aprendizaje automático, el primer paso es recolectar los datos (textos, imágenes, datos numéricos, datos categóricos, etc.) con los que se va a trabajar. Los datos pueden ser extraídos de muchas fuentes distintas, pero las más comunes son: interfaces de programación de aplicaciones² (API), bases de datos o comunidades web. La recolección de datos es un proceso muy importante porque la presencia de datos irrelevantes o de baja calidad pueden dificultar el trabajo al modelo. Posteriormente, es necesario realizar un procesamiento de los datos, tanto para detectar datos erróneos o vacíos, como para adaptar los datos al modelo y asegurar su correcta interpretación. A continuación, se separa el conjunto de datos en dos subconjuntos, denominados conjunto de entrenamiento y conjunto de prueba. Tras esto, se procede a entrenar el modelo con los datos de entrenamiento y a validar la precisión y el rendimiento obtenidos con los datos de prueba. Por último, se realiza un análisis de los resultados obtenidos para poder obtener conclusiones de los datos.

¹ también es llamado Machine Learning (ML)

² La aplicación ofrece una interfaz para que los datos que han sido recolectados sean fácilmente consultados y extraídos.



Ilustración 2.- Proceso para resolver un problema de aprendizaje automático. Fuente: Propio

Entre los enfoques que tiene el aprendizaje automático, se encuentra el aprendizaje supervisado. Este tipo de aprendizaje recibe datos con etiquetas que guían al modelo en el proceso de entrenamiento, aumentando su precisión a la hora de predecir y tomar decisiones sobre nuevos datos. Un ejemplo de problema de aprendizaje supervisado es clasificar mensajes de correo en spam/no spam, o reconocer objetos en imágenes, similar a este trabajo, en el que se pretende detectar emociones en imágenes. El aprendizaje automático puede resolver problemas de clasificación y de regresión mediante los algoritmos que se presentan a continuación:

Regresión lineal: Un algoritmo de regresión lineal se basa en el modelo matemático de escala lineal. Este algoritmo suele tomar como entrada datos con varias variables, teniendo en cuenta más de una entrada para predecir un dato continuo. Por ejemplo, el algoritmo puede predecir el precio de un coche en función de la marca, modelo, prestaciones y tipo de combustible.

Regresión logística: Un algoritmo de regresión logística está enfocado a predecir datos categóricos teniendo en cuenta uno o varios datos de entrada. Cabe destacar que una predicción puede pertenecer a más de una clase. A modo de ejemplo, este tipo de algoritmos podría predecir si un meteorito caerá en el mar o no, teniendo en cuenta las ubicaciones de todas las caídas de meteoritos anteriores.

Redes neuronales: Una red neuronal se basa en las conexiones entre neuronas del cerebro humano para interconectar nodos. Los nodos de las redes neuronales más simples (MLP) reciben una entrada, ponderación y sesgo, y producen una salida que mandan al siguiente nodo según una función de activación. El algoritmo intenta ajustar las ponderaciones mediante la propagación hacia atrás de los valores calculados. Una red neuronal es capaz de detectar la presencia de peces o caballos en imágenes.

Naive bayes: El algoritmo Naive Bayes determina la clase a la que pertenece un dato de entrada basándose en el Teorema de Bayes. De este modo, trata de forma

independiente las distintas características que se tienen en cuenta. Este algoritmo podría clasificar las reseñas de un videojuego en positivas o negativas.

KNN: El algoritmo de k vecinos más cercanos trata los datos como puntos. Realiza la clasificación teniendo en cuenta la distancia (manhattan o euclídea, entre otras) siguiendo la idea de que datos similares están próximos. Por ejemplo, este algoritmo puede predecir el riesgo de una persona de padecer un cierto tipo de cáncer.

Random Forest: El algoritmo Random Forest crea varios árboles de decisión para finalmente predecir la clase del árbol que mejor se adapta al dato de entrada. Por ejemplo, este algoritmo puede predecir un tipo de medicamento según sus características químicas.

SVM³: El algoritmo de máquina de vectores de soporte, al igual que el algoritmo KNN, trata los datos como puntos. Usa un hiperplano en el que los datos de dos clases están separados lo máximo posible entre sí. Por ejemplo, este algoritmo podría clasificar tipos de aperitivos en dulces y salados.

2.1.1 Redes neuronales convolucionales

Las redes neuronales convolucionales CNN tienen un mejor rendimiento sobre problemas de clasificación de imágenes que las redes MLP porque se basan en la multiplicación de matrices sobre píxeles para poder identificar patrones. A veces es necesaria una unidad GPU para llevar a cabo la resolución del problema debido a la gran cantidad de información de los píxeles. Su funcionamiento es el siguiente: al principio reconoce características simples como líneas y curvas, pero a medida que avanza en el procesamiento, identifica partes cada vez más grandes de la imagen. Este tipo de red neuronal no usa capas totalmente conectadas en su estructura interna⁴

³ Desarrollado por Vladimir Vapnik

⁴ También denominadas “fully connected”. El modelo que se ha construido para este trabajo sí tiene una capa totalmente conectada, que sirve para mostrar la clase predicha.

como otras redes neuronales porque al estar orientadas a imágenes, no tiene sentido relacionar píxeles lejanos, sino adyacentes para encontrar regiones relevantes.

2.1.1.1 Capa convolucional

Las CNN son las redes neuronales que utilizan una convolución matemática en al menos una de sus capas. Una convolución es una operación conmutativa que involucra dos funciones con valores reales como argumento.

En el contexto de las redes convolucionales aplicadas a imágenes, la convolución se define como:

$$g[i,j] = (f * x)[i,j] = \sum_{u=-\infty}^{\infty} \sum_{v=-\infty}^{\infty} f[u,v]x[i-u,j-v]$$

Su propósito es aplicar un filtro a una parte de la imagen para así obtener ciertas características o patrones. La entrada $f [u, v]$ es una imagen en forma de vector o matriz de datos que contienen información de los píxeles, y el filtro $x[i - u, j - v]$ es un vector o matriz de parámetros que se ajustan durante el proceso de aprendizaje. En este contexto, las imágenes se denominan "tensores". Una red neuronal convolucional usa los píxeles de la imagen de entrada, por lo que una imagen en escala de grises (24x24x1) necesitará 576 neuronas para la capa de entrada, y una imagen a color (24x24x3), 1728 neuronas.

La convolución consiste en ir aplicando un filtro –cuyos valores se suelen inicializar de forma aleatoria, con cierto paso “stride” a un conjunto de píxeles contiguos en la imagen de entrada. El filtro recorre todas las neuronas de entrada y genera una nueva matriz de salida llamada “mapa de activación”. La red neuronal ajustará los valores del filtro conforme avanza en el procesamiento para obtener patrones relevantes en la imagen. El paso indica cada cuantos píxeles aplicar el filtro. De esta forma, las dimensiones del resultado de aplicar un filtro ($k \times k \times d$) con paso (s) a una imagen de entrada ($h \times w \times d$) serán: $h' = ((h - k)/s) + 1$ y $w' = ((w - k)/s) + 1$. Para que funcione, es necesario que el número de canales del filtro y de la imagen (d) coincidan. Después de cada convolución, se aplica una transformación de unidad lineal rectificada (ReLU) al mapa de activación para introducir no linealidad en el modelo.

El modelo aprende mediante propagación hacia atrás,⁵ que consiste en ajustar en cada iteración los valores de los filtros usados en la convolución.

Para imágenes en escala de grises, se va aplicando el filtro a cada región de la imagen.

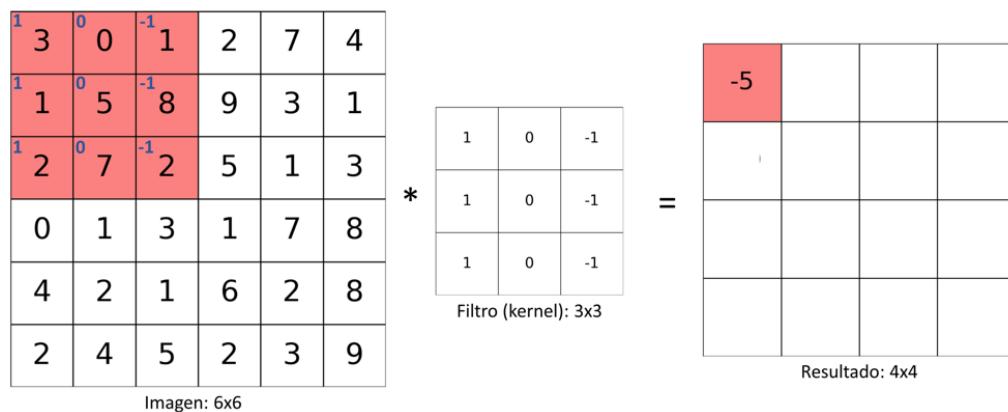


Ilustración 3.- Primera iteración de aplicar una convolución con paso 1 a una imagen en escala de grises.

Fuente: <https://www.codificandobits.com/img/posts/2019-03-30/resultado-convolucion-primer-iteracion.png>

Para imágenes a color que constan de 3 canales⁶ (RGB), se aplica un filtro distinto a cada canal. De este modo, el resultado de aplicar un filtro (3x3x3) a una imagen (24x24x3) es un tensor (8x8x3). Cada elemento del tensor final es el resultado de aplicar la suma a los valores de la aplicación de la convolución a cada canal.

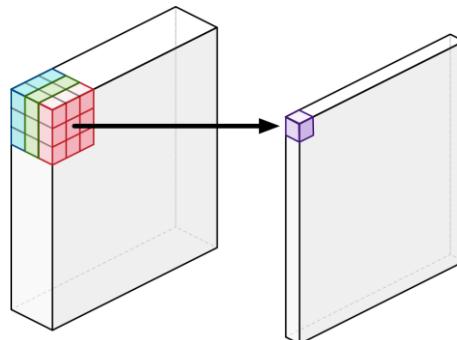


Ilustración 4.- Aplicación de la convolución en la primera iteración a una imagen a color. Fuente:

https://www.researchgate.net/figure/A-pointwise-convolution_fig6_330351722

⁵ Comúnmente conocida como backpropagation

⁶ También llamados tensores 3D

Como se puede observar, el tamaño de la imagen resultante es menor que el de la imagen de entrada. Para evitar esto, se añade un relleno “padding” a la imagen de entrada. El relleno amplía la imagen añadiendo píxeles con valor 0 en los bordes. De esta forma, las dimensiones del resultado de aplicar un filtro ($k \times k$) con paso (s) a una imagen de entrada ($h \times w$) con relleno (p) serán: $h' = ((h - k + 2 * p)/s) + 1$ y $w' = ((w - k + 2 * p)/s) + 1$

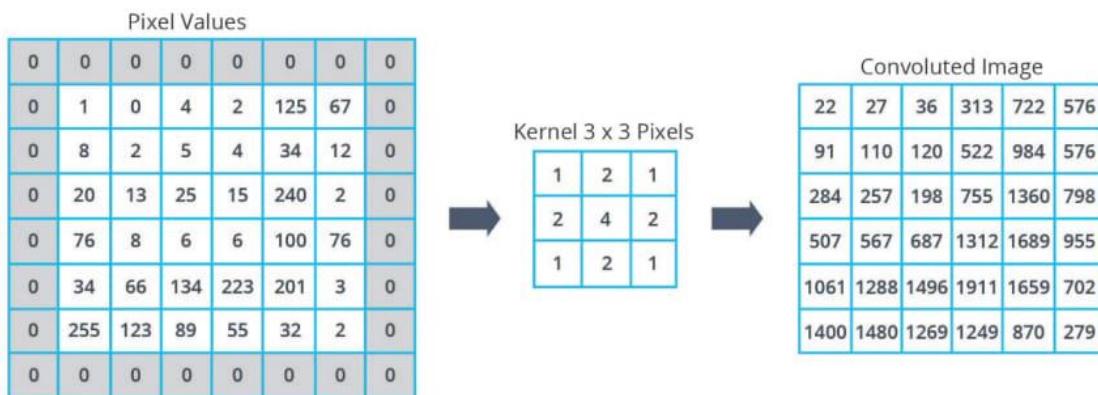


Ilustración 5.- Resultado de aplicar un filtro (3x3x1) con paso (1) a una imagen en escala de gris (6x6x1) con relleno (1) Fuente: https://res.cloudinary.com/practicaldev/image/fetch/s--nUoflRuG--/c_limit%2Cf_auto%2Cfl_progressive%2Cq_auto%2Cw_880/ <https://i.ibb.co/KG5vPdn/final-cnn.png>

También se pueden aplicar n filtros a una misma imagen de entrada, obteniendo n mapas de activación. La capa convolucional en la que se aplican n filtros a una imagen de entrada ($h \times w \times d$) tendrá $h^*w^*d^*n$ neuronas.

2.1.1.2 Capa de agrupación⁷

Antes de poder aplicar una nueva capa convolucional, hay que reducir el número de neuronas de la anterior capa porque de no ser así, el número de neuronas crecería mucho según se realiza el procesamiento. Esta capa reduce el tamaño de la imagen resultado de la convolución, preservando las características importantes de la imagen. Para ello, se aplica un filtro a los valores de la región de la imagen con la que se está trabajando. De esta forma, las dimensiones del resultado de aplicar un filtro ($k \times$

⁷ También llamada “pooling layer”

k) con paso (s) a una imagen de entrada ($h \times w$) serán: $h' = ((h - k)/s) + 1$ y $w' = ((w - k)/s) + 1$.

Se pueden aplicar varios tipos de agrupaciones:

- **Max pooling:** En cada iteración se selecciona el píxel con mayor valor.

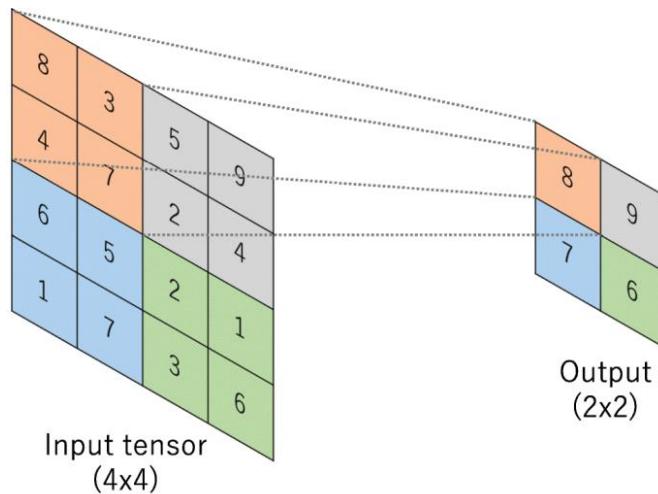


Ilustración 6.- Resultado de aplicar max pooling de (2x2x1) con paso (2) a una imagen (4x4x1) Fuente: <https://medium.com/sukunda-ab-media/the-preliminary-introduction-to-convolutional-neural-network-25004172a289>

- **Average pooling:** En cada iteración se calcula el valor promedio de los valores del subgrupo de píxeles seleccionados.

2.1.1.3 Capa totalmente conectada

Para obtener la clasificación de la imagen de entrada, es necesario aplicar una capa totalmente conectada. Esta capa permite relacionar cada una de las características extraídas con cada una de las clases disponibles. Finalmente, se aplica una función softmax para generar una probabilidad entre 0 y 1 para cada clase.

2.1.1.4 Modelos de redes neuronales convolucionales

Durante los últimos años, debido a la creación de muchos nuevos conjuntos de datos, han surgido nuevos modelos convolucionales:

LeNet-5: Este modelo está entrenado sobre el conjunto de imágenes MNIST y es capaz de detectar números escritos a mano. [2]

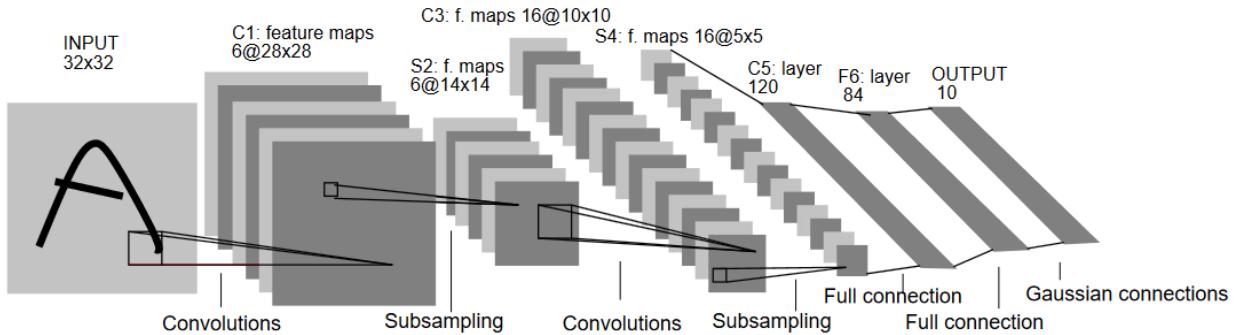


Ilustración 7.- Arquitectura de LeNet-5. Fuente: <http://yann.lecun.com/exdb/publis/pdf/lecun-01a.pdf>

AlexNet: Este modelo, que está entrenado sobre el conjunto de datos ImageNet y LSVRC-2010, consigue clasificar imágenes en 1000 categorías diferentes. [3]

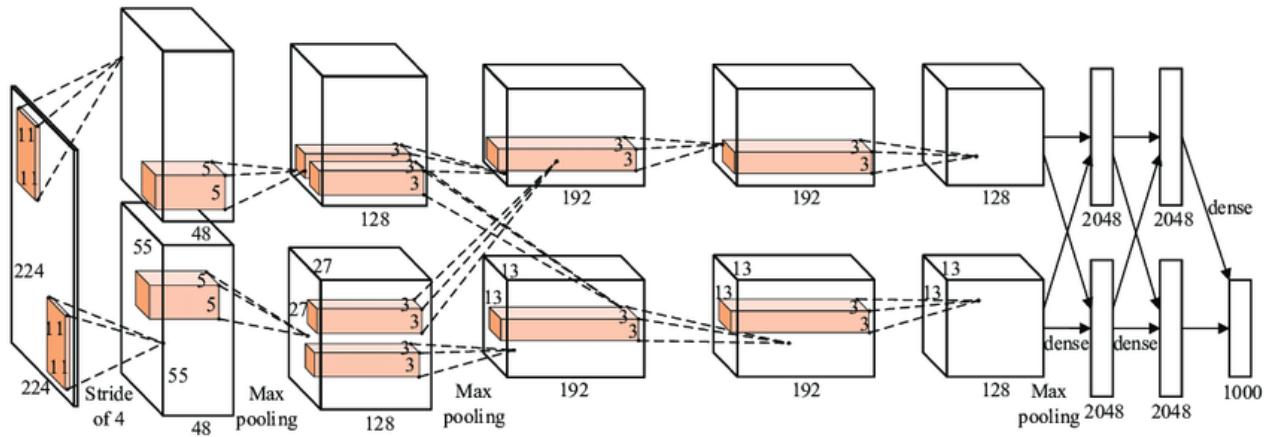


Ilustración 8.- Arquitectura de AlexNet. Fuente: <https://doi.org/10.3390/electronics8030295>

ResNet: Este modelo evalúa redes residuales con una profundidad máxima de 152 capas. Está entrenado sobre el conjunto de datos ImageNet, y ha realizado pruebas para los conjuntos de datos CIFAR-10 y COCO. [4]

VGGNet: Este modelo usa una arquitectura sencilla, compuesta por capas convolucionales, capas de agrupamiento y una capa totalmente conectada. Se caracteriza por usar filtros 3x3. [5]

GoogLeNet: Este modelo usa la arquitectura de incepción para probar varios tamaños de filtro en cada bloque de imagen procesada, que luego concatena para

pasar a la siguiente capa y reducir a la mitad su resolución. Ha sido probado con conjuntos de datos como MNIST, CIFAR e ImageNet. [6]

ZFNet: Este modelo es muy parecido a AlexNet pero éste en cambio, reduce el tamaño de los filtros y reduce el stride. [7]

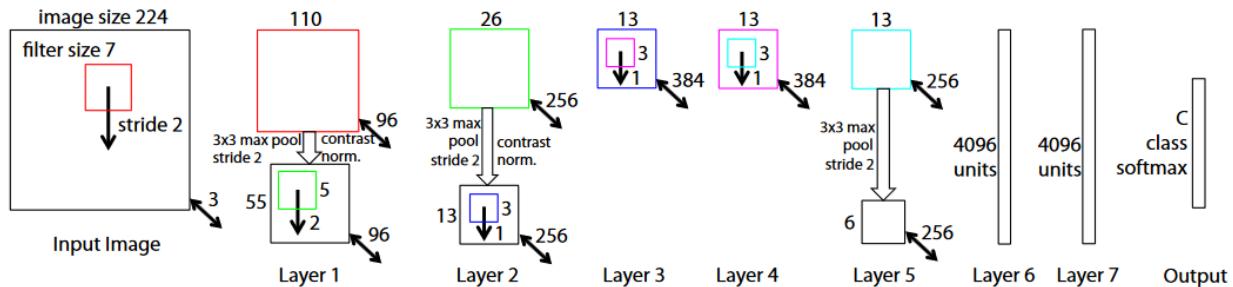


Ilustración 9.- Arquitectura de ZFNet. Fuente: <https://arxiv.org/abs/1311.2901v3>

2.2 Inteligencia Artificial Explicable (XAI)

El problema al que a menudo se enfrentan las personas que trabajan con modelos de caja negra⁸ es a no comprender la razón de las decisiones tomadas por el modelo. En algunos algoritmos de aprendizaje automático como redes neuronales, resulta demasiado complicado determinar la razón de la decisión del algoritmo, debido a su compleja estructura. Los algoritmos de explicabilidad ayudan a comprender por qué los modelos de caja negra determinan salidas distintas para datos de entrada parecidos. El objetivo de este trabajo es valorar estos modelos, no solamente teniendo en cuenta la precisión que consiguen, sino también la calidad de las decisiones que toman.

Actualmente existen métodos XAI que permiten explicar algunos algoritmos de aprendizaje automático:

LIME: Es un algoritmo que muestra explicaciones locales agnósticas al modelo. Trabaja con modelos de caja negra que son capaces de procesar datos tabulares, textos e imágenes. [8]

⁸ También denominados black box models

SHAP: Es una librería que, al igual que LIME, muestra explicaciones locales agnósticas, pero ésta en cambio, puede trabajar sobre cualquier modelo de aprendizaje automático. Se basa en la teoría de juegos para asignar puntuaciones a cada característica relevante. [9]

ANCHOR: Es una librería que ancla localmente la predicción, de forma que los cambios externos a la zona fijada en el mismo dato no se tienen en cuenta. De este modo, datos que fijan el mismo ancla tendrán predicciones muy parecidas. Trabaja sobre datos tabulares y textos para cualquier clasificador de caja negra. [10]

XRAI: Es un método que muestra explicaciones locales basándose en gradientes integrados. Solamente puede ser aplicado a imágenes. [11]

Integrated Gradients: Es un método que se basa en la sensibilidad e invariabilidad de implementación. Muestra explicaciones mediante llamadas al operador de gradiente. Solamente es aplicable a modelos que trabajan con textos e imágenes. [12]

Grad-CAM: Es un método que primero calcula los gradientes de una imagen para después pasárselos a una capa convolucional y así producir un mapa de localización de las regiones relevantes. Funciona sobre cualquier red neuronal convolucional aplicada a imágenes. [13]

RISE: Es un método orientado a explicar imágenes que construye un mapa para indicar la importancia de cada píxel en la predicción. [14]

FORGrad: Es un método creado para modelos de caja blanca sobre imágenes que filtra el ruido de los gradientes del mapa de atribución calculado por el método de gradiente. [15]

ALIBI Explain: Es una librería open source que interpreta modelos de aprendizaje automático de caja blanca y negra. A pesar de que se pueda aplicar sobre modelos que resuelven problemas de clasificación y de regresión sobre diversos tipos de datos, en este trabajo se ha enfocado su uso para modelos de caja negra que resuelven problemas de clasificación de imágenes. El ámbito que se tomará será local, es decir, se mostrará sobre cada imagen las regiones relevantes para la toma de decisión del modelo. Esta librería es la que se va a usar en este trabajo porque engloba muchos de

los métodos comentados anteriormente, como Integrated Gradients y AnchorImage aplicados a imágenes. [16]

A continuación, se va a mostrar un mapa con las librerías de explicabilidad más prometedoras hasta el momento y una tabla comparativa de frameworks entre distintas librerías:

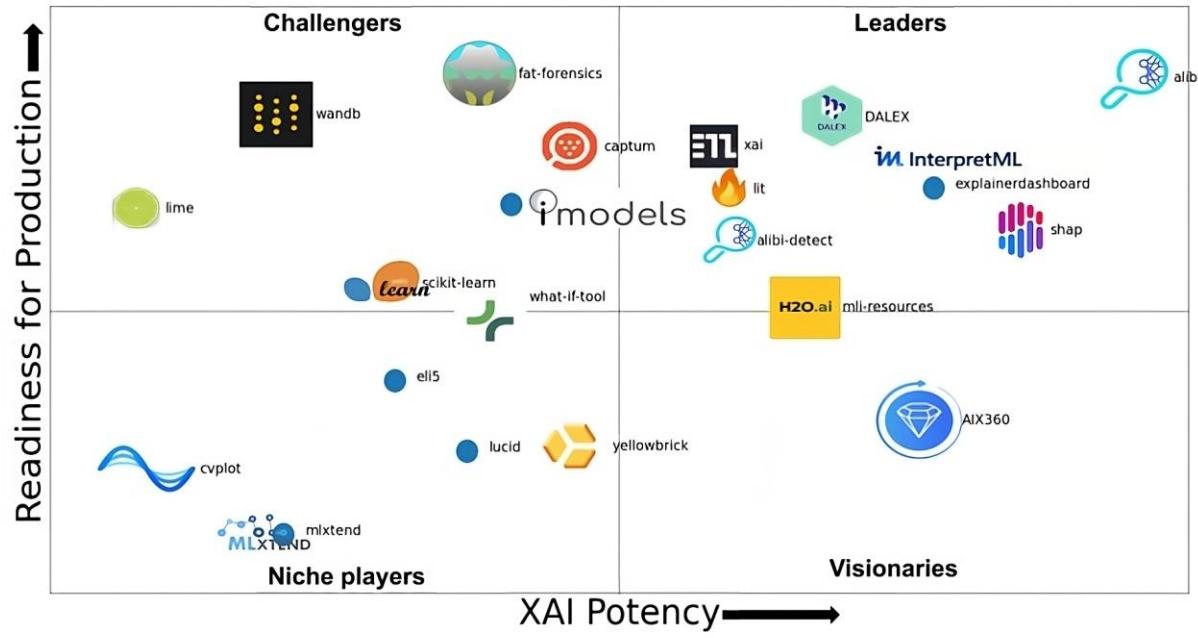


Ilustración 10.- Mapa de algunas librerías de explicabilidad. Fuente: [https://assets-global.website-files.com/6299b185f76ad4ffc9cdbc/62a72b2ed63903a5ec279339_627a5fa6ecda298d1a40f2a0_Existing%2520XAI%2520libraries%2520Overview\(1\)%2520\(1\).png](https://assets-global.website-files.com/6299b185f76ad4ffc9cdbc/62a72b2ed63903a5ec279339_627a5fa6ecda298d1a40f2a0_Existing%2520XAI%2520libraries%2520Overview(1)%2520(1).png)

Tabla 1.- Compatibilidad de librerías de explicabilidad con frameworks de ML y tipos de datos admitidos.
Fuente: Propio

Librería	Compatible con Scikit-Learn	Compatible con TensorFlow	Compatible con PyTorch
LIME	Datos tabulares	Sí	Imágenes
ANCHOR	Datos tabulares y textos	Sí	Imágenes

ALIBI	Sí	Sí	Sí
SHAP	Datos tabulares	Sí	Textos e imágenes
XRAI	No	Imágenes	Imágenes
Integrated gradients	No	Datos tabulares e imágenes	Datos tabulares e imágenes
GradCam	No	No	Imágenes
ForGRad	No	Imágenes	Imágenes
RISE	No	Imágenes	Imágenes

2.3 Aplicaciones prácticas de reconocimiento de emociones en imágenes

Muchas personas que prueban nuevos productos no tienden a expresar lo que realmente piensan acerca de éste, por lo que no permite la mejora del producto. Por otra parte, casi la mitad de los pacientes con patologías no expresan verbalmente su dolor, causando la ralentización de la labor de los médicos. El reconocimiento de emociones en caras por parte de modelos de aprendizaje automático y su posterior explicación pueden solventar problemas como los comentados anteriormente. Algunos colegios e institutos, como por ejemplo el True Light College en Hong Kong [17], han empezado a implementar algunos modelos de IA de caja negra con el objetivo de analizar las emociones de los alumnos durante la clase. Gracias a la visualización de las regiones relevantes de la imagen, detectadas por el modelo, que ofrecen los algoritmos de explicabilidad, los tutores, psicólogos y orientadores pueden enfocar su trabajo, no solamente para aumentar el rendimiento en el aula, sino para detectar casos de acoso escolar de forma precoz. Estos modelos también pueden ayudar a detectar emociones de personas autistas y reducir los accidentes de tráfico mediante la detección de las emociones que experimenta el conductor, como el cansancio o la fatiga.

Actualmente existen empresas tecnológicas y entidades educativas que han desarrollado herramientas de Inteligencia Artificial relacionadas con el reconocimiento de emociones en imágenes:

Azure AI Face service: Microsoft ofrece algoritmos de análisis de caras, detección de caras falsas, identificación de personas y búsqueda de caras similares. [18]

Amazon Rekognition: Ofrece algoritmos de verificación, búsqueda, detección y reconocimiento de características en caras, y detección de contenido inapropiado, logotipos, famosos y texto. [19]

Google Cloud Vision: Ofrece un algoritmo que detecta ojos, cejas, bocas, etc. de varias caras en una imagen. También es capaz de detectar complementos como sombreros, gafas, gorros o pendientes, entre otros. [20]

Watson: Es un asistente de IBM que ofrece muchas funcionalidades, entre ellas, reconocimiento de detección de emociones en textos. [21]

EnableX Face AI: Es una herramienta que analiza las emociones de caras en tiempo real sobre una o muchas personas a la vez. [22]

Face Mirror: Es un producto de la compañía Bismart que detecta y analiza el estado de ánimo de la persona que tiene delante. [23]

Emotongue: Es una aplicación móvil creada por una investigadora de la UAH que detecta emociones mediante el escaneo facial, ofreciendo recomendaciones sobre cómo lidiar con las emociones detectadas. [24]

A pesar de que algunas de las herramientas presentadas anteriormente se comportan de forma parecida a los modelos creados en este trabajo, se ha decidido no aplicar la explicabilidad sobre estas herramientas, ya que se pretenden adquirir conocimientos acerca del funcionamiento y estructura de algunos modelos de IA de detección de emociones en imágenes.

2.4 Trabajos similares

En la página web de Kaggle⁹ se pueden consultar algunos trabajos similares que crean o adaptan modelos para trabajar sobre el conjunto de datos FER2013 y poder así detectar emociones en imágenes. Muchos de los modelos disponibles que utilizan este conjunto de imágenes son modelos con estructura muy similar a la que se va a desarrollar en este trabajo, por lo tanto, es interesante estudiar las precisiones obtenidas para cada modelo. A continuación, se muestra una tabla en la que se muestran las características importantes, rendimientos, y enlaces de algunos modelos interesantes:

Tabla 2.- Características, rendimientos y enlaces de modelos usados en algunos trabajos similares

Características importantes del modelo	Rendimiento	Enlace
Sequential() Función de activación: "Relu" 6 capas convolucionales con 32, 64, 128, 256, 512 y 512 filtros de tamaño (3,3) Tamaño de agrupación: (2,2) BatchNormalization() Dropout(0.25)	10 épocas de entrenamiento Entrenamiento: 0.5856 Prueba: 0.5552	https://www.kaggle.com/code/tharanitharanm/fer-using-cnn
Transfer learning con VGGNET VGG16 + Flatten() + Dense(1024, activation='relu') + Dropout(0.5) +	50 épocas de entrenamiento Entrenamiento: 0.5472 Prueba: 0.55	https://www.kaggle.com/code/tharanitharanm/fer-using-cnn

⁹ www.kaggle.com

Dense(512, activation='relu') + Dropout(0.5)		
Transfer learning con RESNET-50 ResNet50V2() + Dropout(0.25) + BatchNormalization() + Flatten() + Dense(64,activation='relu') + BatchNormalization() + Dropout(0.5)	9 épocas de entrenamiento Entrenamiento: 0.6127 Prueba: 0.6080	https://www.kaggle.com/code/tharanitharanm/fer-using-cnn
Sequential() Función de activación: "Relu" 4 capas convolucionales con 32, 64, 128 y 256 filtros de tamaño (3,3) Tamaño de agrupación: (2,2) BatchNormalization() Dropout(0.25)	20 épocas de entrenamiento Entrenamiento: 0.6592 Prueba: 0.6236	https://www.kaggle.com/code/ojaspjoshi/emotion-detection
Transfer learning con RESNET-50 (simplificado) Función de activación: "Relu" 4 capas convolucionales con 64, 128, 256 y 512 filtros y pasos de 1, 2, 2 y 2. Tamaño de agrupación: (2,2) BatchNormalization() MaxPooling (3,3) con paso 3 AveragePooling (2,2)	40 épocas de entrenamiento Entrenamiento: 0.7074 Prueba: 0.6131	https://www.kaggle.com/code/dishaasinghi/resnet-simplified

Sequential() Función de activación: "Relu" 4 capas convolucionales con 32, 64, 128 y 256 filtros de tamaño (3,3) Tamaño de agrupación: (2,2) BatchNormalization() MaxPooling2D(2, 2) Dropout(0.25)	150 épocas de entrenamiento Entrenamiento: 0.7360 Prueba: 0.6680	https://www.kaggle.com/code/aslinaslin/emotion-detection
Transfer learning con RESNET-50v2 ResNet50V2() + Dropout(0.25) + BatchNormalization() + Flatten() + Dense(64,activation='relu') + BatchNormalization() + Dropout(0.5) + Dense(7,activation='softmax')	22 épocas de entrenamiento Entrenamiento: 0.8906 Prueba: 0.7962	https://www.kaggle.com/code/nexuswho/emotion-detection

Algunos de estos trabajos se limitan a transformar modelos previamente entrenados como VGGNET o RESNET para aprovechar todas sus ventajas y así obtener rendimientos algo más altos, pero en general, todos los modelos obtienen una precisión de cercana a 0.6. Esto se puede deber a la limitación de los datos, ya que el conjunto no está balanceado. Algunos modelos necesitan muchas épocas para el proceso de entrenamiento, pero realizando una comparación entre ellos, el número de épocas idóneo es 40.

Capítulo 3 - Procesamiento de los datos y estructura de los modelos

En este capítulo se va a realizar un estudio de los datos para verificar si es necesario llevar a cabo un procesamiento de estos. Se detallará el proceso de tratamiento de los datos realizado, así como la especificación de las características elegidas para construir los modelos usados en los siguientes capítulos.

3.1 Carga del conjunto de imágenes

El conjunto de datos que se usa para realizar la primera aproximación al trabajo es un conjunto pequeño de imágenes de diferentes tamaños a color de personas con tres tipos de emociones: felicidad, tristeza y enfado; y para realizar la aproximación final del trabajo se usa un conjunto más amplio de imágenes 48x48 en escala de grises, en concreto, con siete tipos de emociones, llamado FER2013.

El primer paso para que un modelo logre clasificar imágenes es cargar el conjunto de datos con el que se va a entrenar y, posteriormente probar el modelo. Como el clasificador que se va a construir se trata de un modelo de aprendizaje supervisado, es necesario cargar junto a las imágenes de entrada las clases asociadas a las mismas que, en este caso, serán las emociones expresadas en cada imagen.

Dado que ambos conjuntos de datos están organizados en varias carpetas que contienen las imágenes de cada emoción, el proceso de carga de las imágenes es el siguiente: se realiza un recorrido por el directorio que contiene las imágenes, y se guarda en una estructura de datos bidimensional¹⁰ la ruta de la imagen junto al nombre de la carpeta contenedora de la imagen (será la emoción presente en la imagen). Tras la carga de las imágenes, es conveniente comprobar si el conjunto de datos está equilibrado, es decir, si hay aproximadamente el mismo número de muestras para todas las clases, porque de lo contrario, el modelo puede aprender a favorecer la clase

¹⁰ Concretamente en un Pandas dataframe de la librería pandas.

mayoritaria o incluso sesgar las predicciones a favor de cierta clase en vez de aumentar la precisión de todas las clases existentes.

3.2 Adaptación del formato de las imágenes

El formato de las imágenes de entrada del modelo también es un factor bastante importante a tener en cuenta, ya que influirá significativamente en el rendimiento del modelo. Si se pretende que un modelo sea rápido en predecir una entrada, será necesario que dicha entrada solamente contenga datos imprescindibles y no superfluos de los que el modelo sea capaz de extraer características. Por ejemplo, una imagen en tonos de color (RGB) puede conservar todas sus características en muchos menos datos tras ser convertida a escala de grises, pasando de tener tres canales a solamente uno, o incluso tras ser redimensionada a un tamaño relativamente menor del original. La adaptación del formato de las imágenes dependerá del conjunto de datos usado.

3.2.1 Tamaño

El tamaño de las imágenes y la información de los píxeles influye significativamente en el rendimiento del modelo. Como un modelo no puede trabajar con imágenes de diferentes tamaños, se ha realizado un estudio de las resoluciones de las imágenes del conjunto de datos para determinar si hace falta realizar algún procesamiento a las imágenes para adaptarlas a un tamaño final. Si es necesario procesarlas, entonces, dependiendo del tamaño elegido, algunas imágenes perderán resolución al reducir su número de píxeles (downscaling) y otras añadirán información no representativa (upscaleing).

Las imágenes del conjunto de datos inicial tienen tamaños muy variados. La única característica común a las imágenes es su alta resolución, que será un problema a la hora de entrenar los modelos. Cada punto de la gráfica siguiente representa una imagen del conjunto de datos inicial.

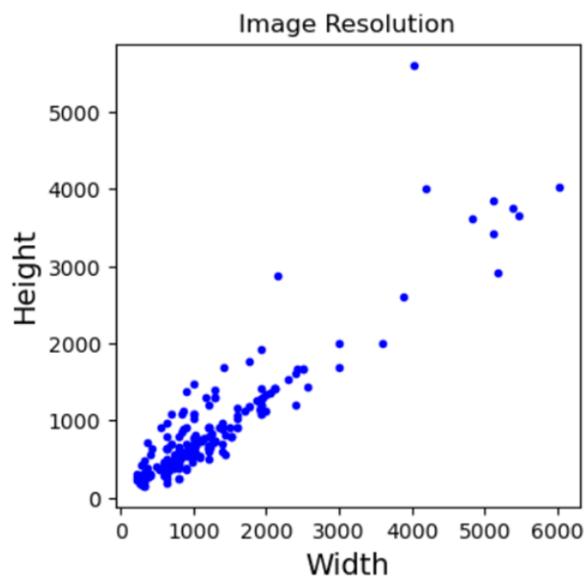


Ilustración 11.- Resoluciones de las imágenes del conjunto de datos inicial. Fuente: Propio

Para reducir el tiempo de entrenamiento del modelo, los tamaños elegidos para realizar pruebas con los modelos serán: la media de las resoluciones (1152, 809), la media*(2/3) (768, 539), la media/3 (384, 269) y otros tamaños más pequeños como (96, 96), (48, 48) y (24, 24)



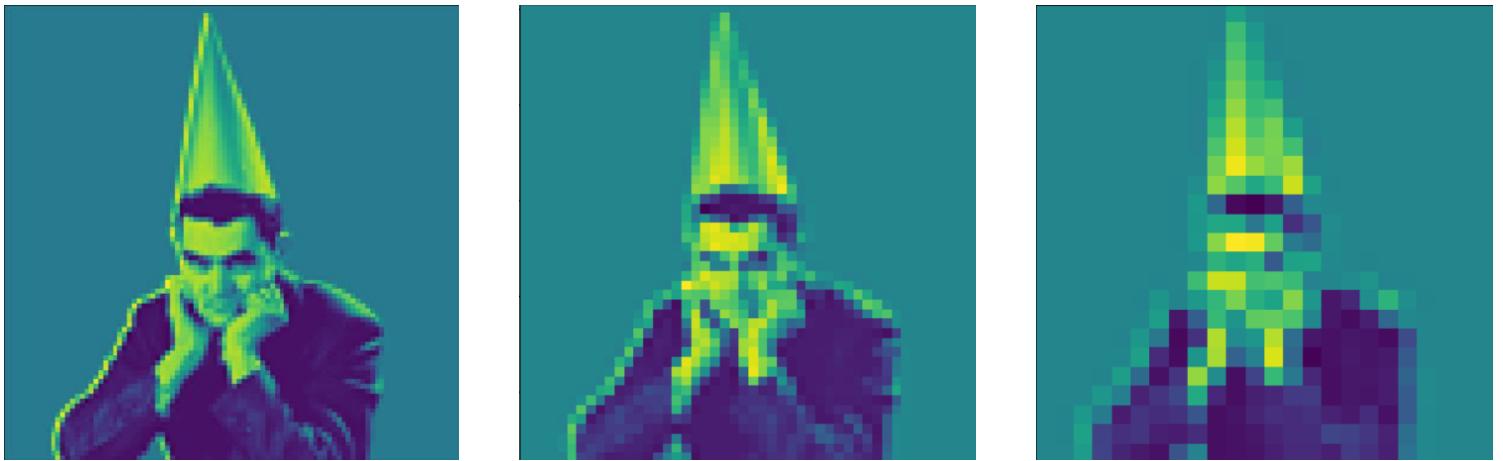


Ilustración 12.- Visualización de las imágenes del conjunto de datos inicial tras redimensionar. De izquierda a derecha y de arriba a abajo: (i)(1152x809) (ii)(768x539) (iii)(384x269) (iv)(96x96) (v)(48x48) (vi)(24x24)
Fuente: Propio

Dado que todas las imágenes del conjunto de datos final tienen el mismo tamaño, no es necesario normalizar los tamaños de las imágenes. Por otro lado, como se pretende realizar un estudio para determinar qué características conducen un modelo a lograr una mejor precisión, se van a probar distintos tamaños para los datos de entrada del modelo, probando así con el doble del tamaño original (96x96), el tamaño original (48x48), y la mitad del tamaño original (24x24).

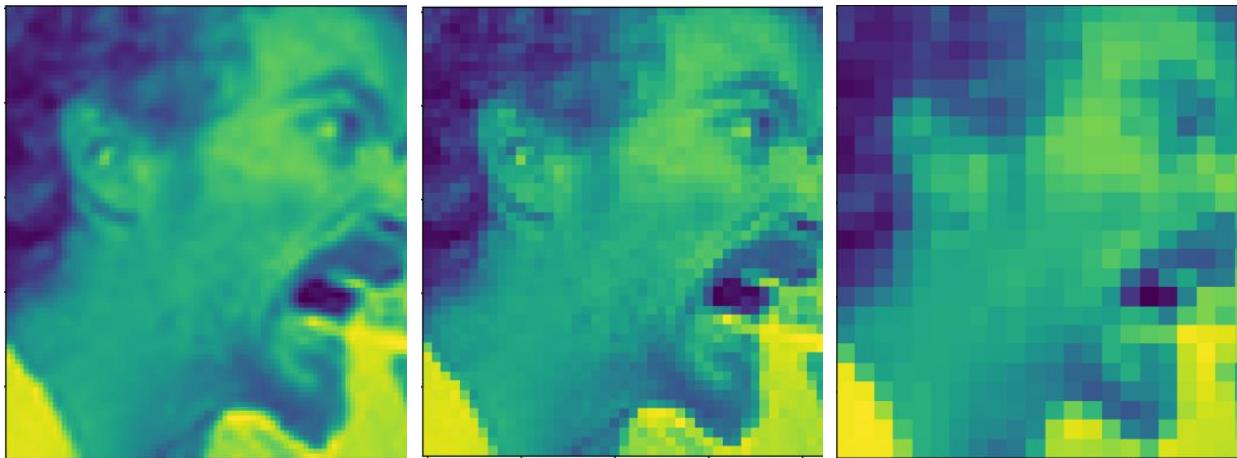


Ilustración 13.- Visualización de las imágenes del conjunto de datos final tras redimensionar. De izquierda a derecha: (i)(96x96) (ii)(48x48) (iii)(24x24) Fuente: Propio

3.2.2 Color

El color de las imágenes con las que trabaja un modelo es también un factor importante que se debe tener en cuenta. Un modelo tardará menos en procesar una imagen en escala de grises que una imagen a color porque la primera solamente tiene un canal, y la segunda tiene tres. Por este motivo, en el conjunto de datos inicial es necesario un tratamiento de las imágenes para convertirlas a escala de grises. Con respecto al conjunto de datos final, como todas las imágenes están en escala de grises, no es necesario realizar ningún tratamiento.

3.3 Estructura de los modelos

Se van a crear varios modelos con distintas características, con el objetivo de determinar qué combinación de características son las que llevan a lograr una mayor precisión sobre el conjunto de imágenes de entrada.

3.3.1 Modelos MLP

Aunque este tipo de modelo no esté orientado al tratamiento de imágenes, se ha decidido analizar su comportamiento y precisión para imágenes en forma de vectores. Se va a crear un modelo de perceptrón multicapa distinto para cada tamaño de las imágenes, función de activación, fuerza del término L2 de regularización¹¹ y número de capas ocultas, usando la red neuronal MLPClassifier de la librería sklearn. La estructura general de los modelos MLP creados es la siguiente:

Tabla 3.- Capas y tamaños generales de los modelos MLP usados

Capa	Tamaño
input (InputLayer)	(input_size_h*input_size_w)
HiddenLayer	(n_hidden_layers)
OutputLayer	(num_classes)

¹¹ También llamado alpha

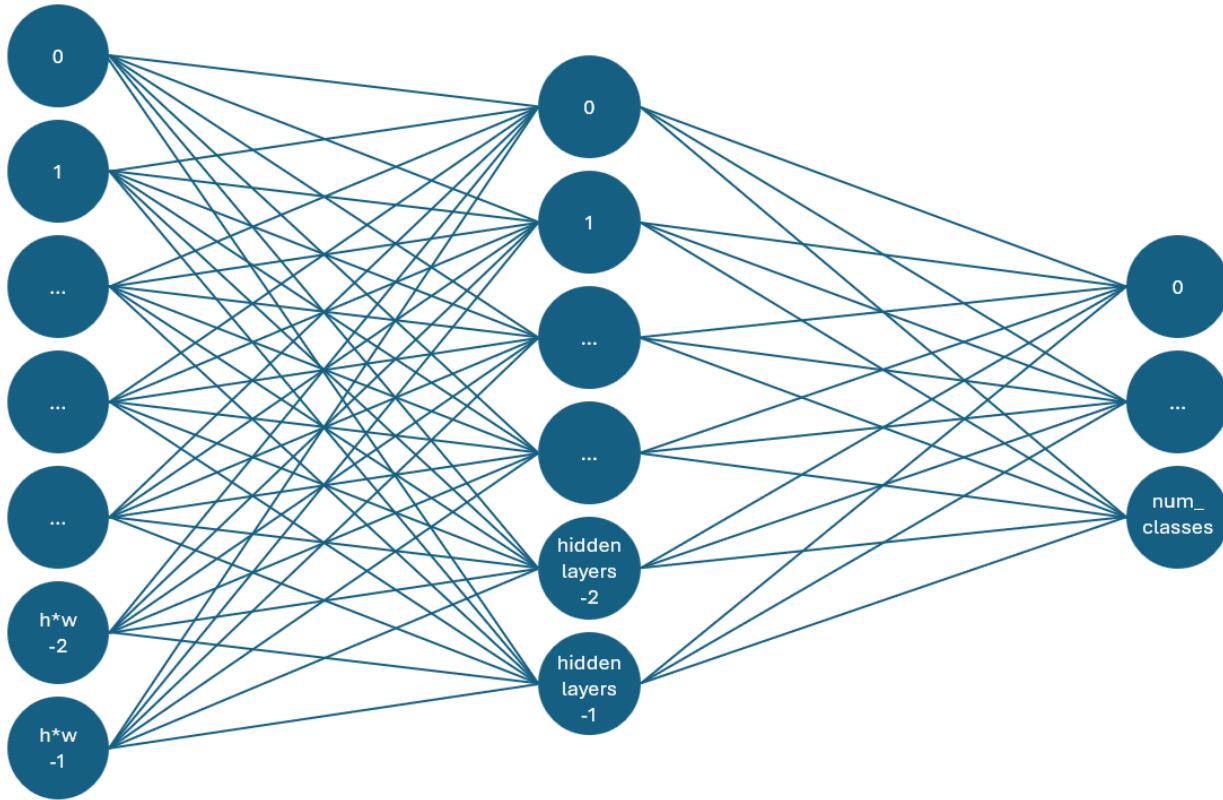


Ilustración 14.- Estructura de los modelos MLP usados. Fuente: Propio

La primera capa (inputLayer) se corresponde con los píxeles que dependen del tamaño de la imagen. El número de capas intermedias (hiddenLayers) variará entre 150 y 200. La última capa (outputLayer) corresponde con las diferentes clases que se pretenden clasificar (3 o 7)

La función de activación podrá ser: “ReLU” o “logistic” y el valor de alpha podrá tomar los valores: 0.01, 0.51, 1.01 o 1.51.

3.3.2 Modelos CNN

Para crear este tipo de modelo, se usan las capas disponibles en keras de la librería tensorflow. Se va a crear una red neuronal convolucional distinta para cada tamaño de imagen, función de activación, tamaño de los filtros y tamaño de agrupación, partiendo de la estructura general siguiente:

Tabla 4.- Estructura general de los modelos de redes neuronales convolucionales usados.

Capa	Tamaño de salida
input (InputLayer)	[(None, input_size, input_size, 1)]
conv2d (Conv2D)	(None, input_size, input_size, 64)
max_pooling2d (MaxPooling2D)	(None, input_size/pool_size, input_size/pool_size, 64)
dropout (Dropout)	(None, input_size/pool_size, input_size/pool_size, 64)
conv2d (Conv2D)	new_input_size = input_size/pool_size (None, new_input_size, new_input_size, 32)
max_pooling2d (MaxPooling2D)	(None, new_input_size/pool_size, new_input_size/pool_size, 32)
dropout (Dropout)	(None, new_input_size/pool_size, new_input_size/pool_size, 32)
flatten (Flatten)	(None, new_input_size/pool_size* new_input_size/pool_size*32)
dense (Dense)	(None, 256)
dropout (Dropout)	(None, 256)
logits (Dense)	(None, num_classes)
softmax (Activation)	(None, num_classes)

Como se puede observar, el modelo consta de una capa inicial que se encarga de recibir las imágenes de entrada. Después, una capa convolucional aplica un filtro sobre las regiones de las imágenes. El resultado de la capa anterior se usa de entrada para una capa de agrupación que se encarga de partir la imagen en varios grupos para después seleccionar el mayor elemento de cada grupo. Esto hace que el tamaño de la imagen resultante sea mucho menor, evitando que la imagen crezca en el proceso. A la imagen reducida se le aplica una capa que desactiva neuronas con cierta probabilidad para evitar que el modelo sobreaprenda. Tras acabar este primer ciclo, se realiza una nueva vuelta para después aplicar las últimas capas. En la última

parte se aplica una capa que convierte la imagen en un vector unidimensional para que pueda ser tratado por la siguiente capa, y que combina las características extraídas por las capas anteriores para clasificar la entrada. A continuación, se aplica otra capa de desactivación de neuronas, seguida de una capa que da puntuaciones a cada clase para cada imagen de entrada. Por último, se aplica una capa softmax que devuelve la predicción de la imagen de entrada. Para el problema de clasificación de siete tipos de emociones no es necesario añadir más capas convolucionales porque el modelo consigue extraer las características importantes con solamente dos capas.

La función de activación tendrá que ser una función no lineal: "ReLU" o "sigmoid"; el tamaño del filtro podrá ser: 2 o 6; y el tamaño de agrupación (pool_size) podrá ser: 2 o 3.

3.4 Elección del conjunto de datos de entrenamiento y prueba

Para usar el conjunto de datos en el modelo, es necesario determinar el conjunto de entrenamiento (X_{train}) y el conjunto de prueba (X_{test}), con sus etiquetas (y_{train} e y_{test} respectivamente). Se ha usado la herramienta `train_test_split` de la biblioteca `sklearn` para separar con aleatoriedad el conjunto de datos original en 80% entrenamiento y 20% prueba.

3.5 Procesamiento de los datos para los modelos

Habiendo determinado los conjuntos de datos con los que se van a entrenar y probar los modelos, el siguiente paso es llevar a cabo un procesamiento de las imágenes y sus clases para que el modelo pueda extraer características de las imágenes y asociarlas a ciertas clases.

Para ello, en lo que respecta a las imágenes, primero se normalizan los datos de los píxeles para que todos estén dentro del rango [0, 1]. De esta forma, además de hacer que el modelo converja más rápido en el proceso de entrenamiento, se consigue evitar que ciertas características presentes en las imágenes prevalezcan sobre otras debido a una gran diferencia entre los datos de los píxeles.

En lo que respecta a las clases, el modelo no es capaz de procesar las clases directamente como los humanos lo haríamos, es decir, en palabras; pero sí es capaz de

procesarlas si se consiguen representar en formato numérico. Para ello, se ha establecido una relación numérica para relacionar un valor numérico a cada clase. Tras realizar esta conversión, se procede a convertir las clases a formato categórico, representándolas como vectores unitarios.

La relación numérica para la clasificación de tres tipos de emociones es la siguiente: "angry" (0), "happy" (1) y "sad" (2) Por ejemplo, para procesar la etiqueta "happy" primero se convertirá a un 1, y después a (0,1,0).

La relación numérica para la clasificación de siete de emociones es: "angry" (0), "disgust" (1), "fear" (2), "happy" (3), "neutral" (4), "sad" (5) o "surprise" (6) Por ejemplo, para procesar la clase "surprise" primero se convertirá a un 6, y después a (0,0,0,0,0,0,1).

Capítulo 4 - Clasificación de emociones en imágenes

El conjunto de imágenes con el que se va a trabajar para la primera aproximación ha sido obtenido de la página web de Kaggle¹², y está disponible para su descarga¹³. El conjunto de datos consta de cerca de 250 imágenes a color de diferentes tamaños. En las imágenes se pueden visualizar diferentes personas con tres tipos de emociones, entre ellas: felicidad, tristeza y enfado. Las imágenes son muy diferentes entre sí porque se pueden ver, tanto personas individuales como grupos de personas, y también cuerpos completos y caras. El objetivo de la primera aproximación es determinar si el conjunto de datos elegido es útil, es decir, comprobar si el modelo consigue extraer las características suficientes para poder aplicar la explicabilidad. A continuación, se muestran el número de ejemplos y la frecuencia de cada clase del conjunto de datos inicial para verificar si está equilibrado:

Tabla 5.- Comprobación del equilibrado del conjunto de datos inicial

Clase	Número de ejemplos	Frecuencia
“happy”	100	37.735849
“sad”	86	32.452830
“angry”	79	29.811321

Se puede observar que hay más muestras de la clase “happy” que de las demás, lo que podrá afectar negativamente a la precisión del modelo.

El conjunto de imágenes con el que se va a trabajar para realizar la aproximación final también ha sido obtenido de la página web de Kaggle y está disponible para su

¹² www.kaggle.com

¹³ Se puede descargar en el siguiente enlace:

<https://www.kaggle.com/datasets/sanidhyak/human-face-emotions>

descarga¹⁴. FER2013 contiene alrededor de 35.000 imágenes en escala de grises de tamaño 48x48. Las imágenes constan de rostros de personas de diferentes edades que expresan diferentes emociones, entre ellas: enfado, asco, miedo, felicidad, tristeza, sorpresa y neutralidad.

Para comenzar con el análisis final, se ha decidido en primer lugar trabajar sobre un subconjunto de imágenes reducido para determinar si el conjunto original puede ofrecer resultados prometedores. De esta forma, solamente se seleccionan del conjunto de datos original aquellas imágenes en las que aparecen rostros de personas que expresan tres tipos de emociones: felicidad, tristeza y enfado. Con estas tres emociones, se creará un clasificador y se evaluará su precisión. A continuación, se muestran el número de ejemplos y la frecuencia de cada clase del subconjunto de datos usado para verificar si está equilibrado:

Tabla 6.- Comprobación del equilibrado del subconjunto extraído del conjunto de datos final

Clase	Número de ejemplos	Frecuencia
“happy”	8989	44.902343
“sad”	6077	30.356162
“angry”	4953	24.741496

Se puede observar que hay muchas más muestras de la clase “happy” que de las demás, pero dependiendo de las características del modelo usado, el impacto negativo a la hora de clasificar variará.

Para realizar un estudio más aproximado a la realidad, se ha decidido trabajar sobre un conjunto de imágenes más complejo que el inicial, con siete tipos de emociones. A continuación, se muestran el número de ejemplos y la frecuencia de cada clase del conjunto de datos final para verificar si está equilibrado:

¹⁴ Se puede descargar en el siguiente enlace:

<https://www.kaggle.com/datasets/msambare/fer2013>

Tabla 7.- Comprobación del equilibrado del conjunto de datos final

Clase	Número de ejemplos	Frecuencia
“happy”	8989	25.048068
“neutral”	6198	17.270878
“sad”	6077	16.933709
“fear”	5121	14.269791
“angry”	4953	13.801655
“surprise”	4002	11.151671
“disgust”	547	1.524229

Se puede observar, al igual que en el conjunto de datos inicial, sigue habiendo una gran diferencia de ejemplares de la clase “happy” con respecto a las demás. Cabe destacar que la clase “disgust” tiene muy pocas muestras, por lo que afectará significativamente a la precisión de los distintos modelos usados.

4.1 Modelos MLP

Dependiendo de su estructura y del tamaño de las imágenes de entrada, algunos modelos tendrán más fácil detectar patrones de las imágenes de entrada que otros. El tiempo de ejecución de cada modelo dependerá significativamente del número de capas ocultas y del tamaño de los datos con los que se esté trabajando, por lo que es importante establecer una relación entre la precisión obtenida por el modelo, y su tiempo de ejecución para elegir el mejor modelo.

Los tamaños de las imágenes de entrada para los modelos MLP que se van a usar en el proceso de entrenamiento para la primera aproximación son: (1152, 809), (768, 539), (384, 269), (96, 96), (48, 48) y (24, 24) En cambio, los tamaños que se van a usar para la aproximación final son: (96, 96), (48, 48) y (24, 24)

A continuación, se muestra una gráfica con los tiempos y las precisiones obtenidas sobre el conjunto de entrenamiento obtenido del conjunto de datos inicial.

Cada punto representa un modelo MLP con una configuración concreta de los parámetros usados:

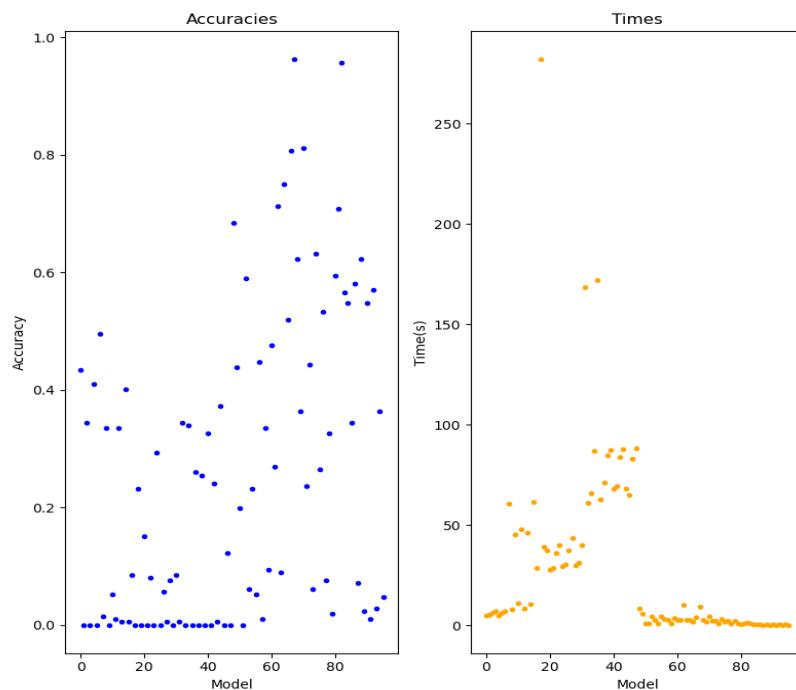


Ilustración 15.- Tiempos y rendimientos de los modelos MLP del conjunto de datos inicial tras el proceso de entrenamiento

Se puede observar que las altas precisiones obtenidas muestran signos de que los modelos han sobreaprendido de los datos de entrenamiento. Curiosamente, la mayoría de los modelos que mayor precisión obtienen son los que tardan menos en entrenar, porque trabajan con imágenes de tamaño bastante reducido (24x24) Es interesante observar cómo casi la mitad de los modelos no consiguen obtener una precisión mayor que 0.33 (que se corresponde a clasificar aleatoriamente los tres tipos de emociones) Una causa de esto puede ser una combinación extraña de características usadas para crear el modelo, la combinación de imágenes que constituyen el conjunto de entrenamiento, o simplemente porque este tipo de modelo no está destinado a procesar imágenes.

A continuación, se muestra una gráfica con los tiempos y las precisiones obtenidas sobre el conjunto de entrenamiento para el problema de clasificación de tres clases del conjunto de datos final. Cada punto representa un modelo MLP con una configuración concreta de los parámetros usados:

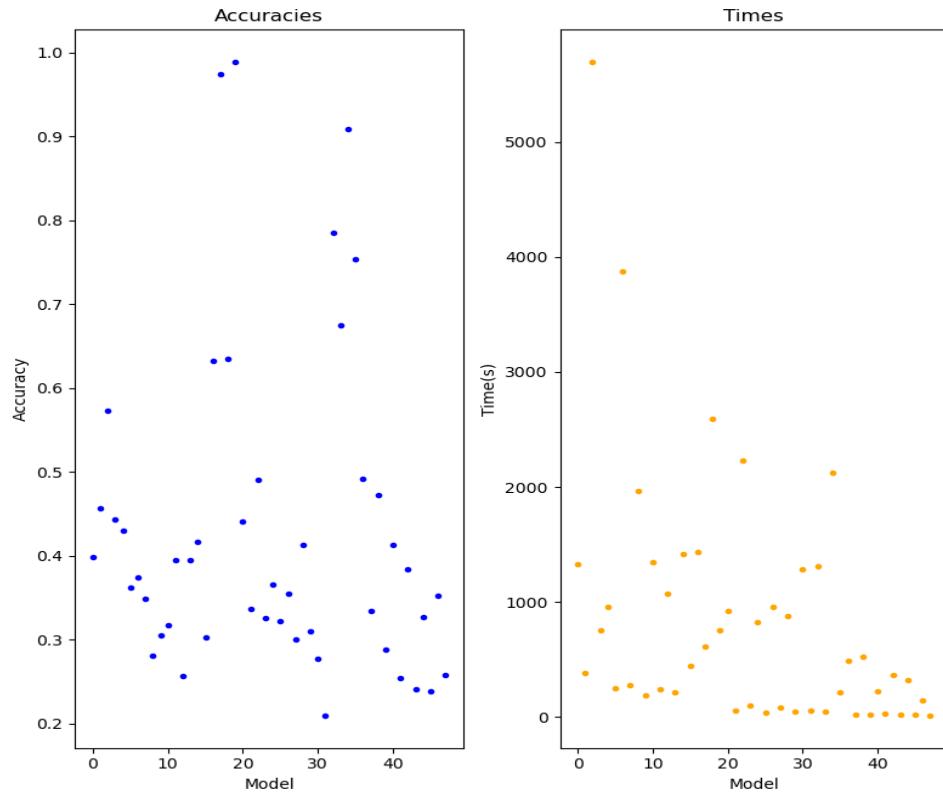


Ilustración 16.- Tiempos y rendimientos de los modelos MLP de clasificación de tres tipos de emociones tras el proceso de entrenamiento

Se puede observar que no se consiguen resultados aceptables. Según indica la gráfica, algunos modelos sobreaprenden de los datos de entrenamiento al obtener un rendimiento cercano al 100%, por lo que su precisión sobre el conjunto de prueba no se espera que sea muy bueno. Hay un modelo que obtiene una precisión de 0.78 en poco tiempo, en comparación con otros modelos que obtienen precisiones ligeramente más altas en tiempos mucho mayores. Esto sugiere que a veces es interesante sacrificar precisión a cambio de ganar tiempo. Llama la atención que alrededor de 1/3 de los modelos no consiguen una precisión mayor que 0.33, que se corresponde a la precisión de realizar una clasificación aleatoria en tres clases. Esto puede deberse a que los modelos MLP no son idóneos para el procesamiento de imágenes.

Debido a que los modelos MLP de clasificación de tres tipos de emociones del conjunto de datos final no ofrecen resultados aceptables, se ha decidido no realizar más pruebas con este tipo de modelo para clasificar siete tipos de emociones, porque en el conjunto de datos completo la diferencia de imágenes entre clases es mucho mayor y se esperan obtener datos aún peores que los previamente obtenidos.

4.1.1 Mejores modelos MLP obtenidos

A continuación, se muestran las características y los resultados del mejor modelo MLP obtenido para el problema de clasificación de tres clases del conjunto de datos inicial:

- Precisión sobre el conjunto de entrenamiento: 0.81
- Precisión sobre el conjunto de prueba: 0.49
- Tamaño de las imágenes de entrada: (48, 48)
- Función de activación: "ReLU"
- Tamaño de las capas ocultas: (200,)
- Alpha: 0.51

Este modelo sobreaprende de los datos de entrenamiento, pero no afecta significativamente a la precisión sobre los datos de prueba porque sigue siendo el modelo que mayor precisión obtiene sobre este conjunto, aunque sea insuficiente.

En la matriz de confusión se puede observar que la clase que mejor clasifica es "angry"; y la que peor "sad", que obtiene un porcentaje de acierto del 50%. Las clases que menos confunde son "sad" y "happy", por lo que se puede deducir que el modelo ha aprendido a diferenciar bocas cóncavas hacia arriba (contento) de bocas cóncavas hacia abajo (triste). Como dato curioso, la clase que mejor clasifica no es la clase mayoritaria, sino la que menos. Esto se puede deber a que las imágenes elegidas encajan con los patrones obtenidos, o que son muy parecidas entre sí.

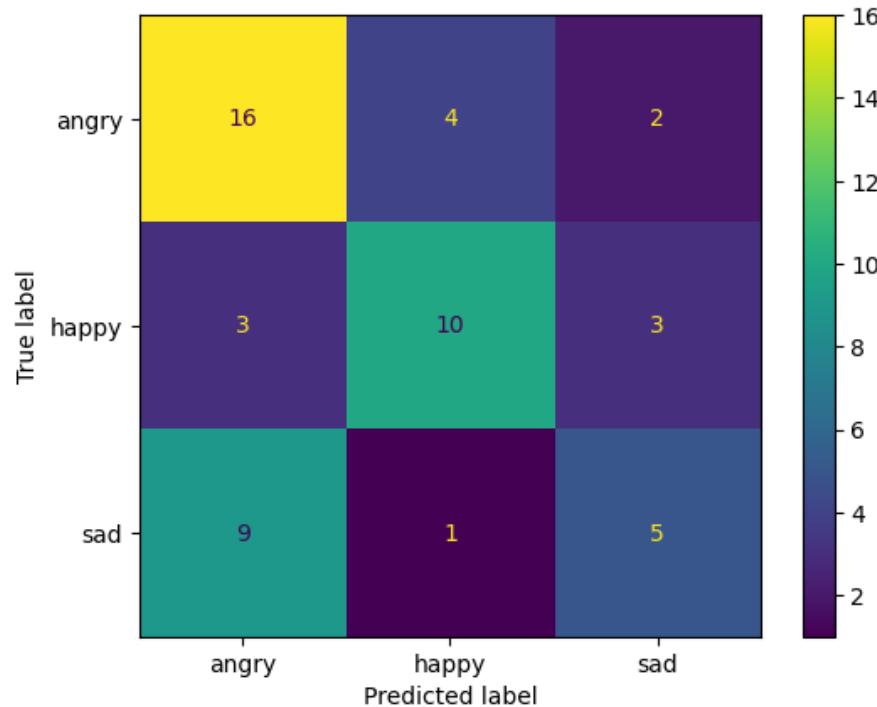


Ilustración 17.- Matriz de confusión del mejor modelo MLP del conjunto de datos inicial

A continuación, se muestran las características y los resultados del mejor modelo obtenido para el problema de clasificación de tres tipos de emociones del conjunto de datos final:

- Precisión sobre el conjunto de entrenamiento: 0.98
- Precisión sobre el conjunto de prueba: 0.51
- Tamaño de las imágenes de entrada: (48, 48)
- Función de activación: "logistic"
- Tamaño de las capas ocultas: (200,)
- Alpha: 0.01

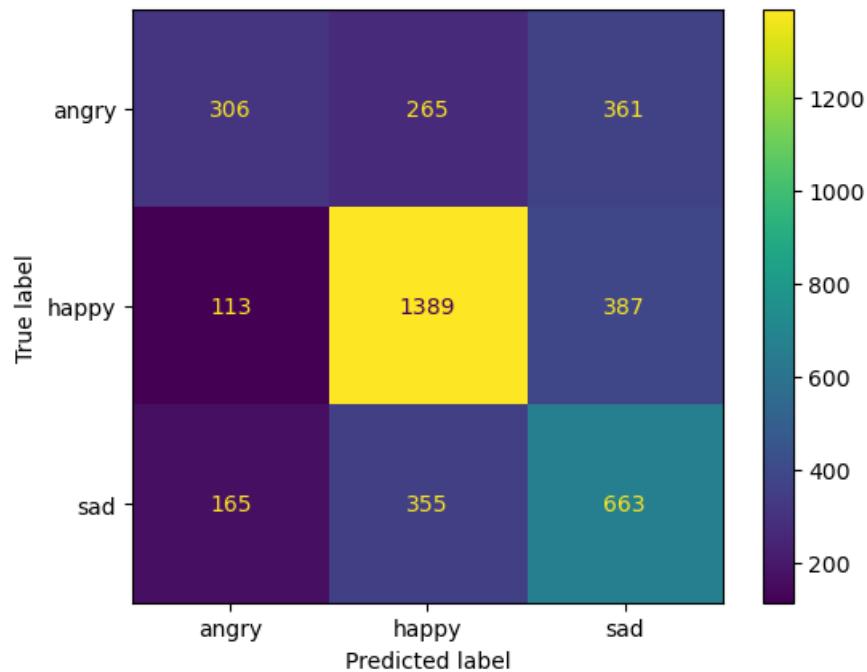


Ilustración 18.- Matriz de confusión del mejor modelo MLP de clasificación de tres tipos de emociones

El mejor modelo sobreaprende de los datos de entrenamiento. Se puede observar que la clase que mejor clasifica es “happy”; y la que peor, “angry”. Esto se puede deber al equilibrado de los datos, ya que la primera clase es la que más imágenes tiene, y la segunda, la que menos. Aunque la clasificación que realiza es bastante acertada, las clases que más confunde entre sí son “happy” y “sad”, aunque no de forma general; las clases que menos confunde son “angry” con “happy”. Esto significa que el modelo ha conseguido extraer las características suficientes como para saber que, por ejemplo, si detecta la presencia de dientes en una imagen, no necesariamente la clase a la que pertenece es “happy”, sino que también puede ser “angry”.

4.2 Modelos CNN

Las redes neuronales convolucionales son idóneas para problemas de clasificación de imágenes, porque son capaces de analizar las características entre conjuntos de píxeles contiguos de las imágenes de entrada en forma de vectores que contienen información de los píxeles. Durante el proceso de entrenamiento de los modelos, se registran los tiempos que tarda cada modelo en aprender – que dependerá

significativamente del tamaño de las imágenes de entrada y del número de épocas; y la precisión obtenida sobre el conjunto de entrenamiento. Para realizar una mejor evaluación de los modelos, se ha usado el mismo conjunto de imágenes de entrenamiento y prueba para entrenar y probar todos los modelos.

Los tamaños de las imágenes de entrada para los modelos CNN que se van a usar en ambos conjuntos de datos son: (96, 96), (48, 48) y (24, 24)

El proceso de entrenamiento para los modelos de clasificación en tres clases de la primera aproximación se va a realizar en 20 épocas,¹⁵ suficientes para el reducido número de imágenes que constituyen el conjunto de datos inicial. Con más de 20 épocas, el modelo se ajusta demasiado a los datos de entrenamiento, pero sin obtener mejora sobre imágenes con las que no ha trabajado.

Más adelante, se muestra una gráfica con los tiempos y las precisiones obtenidas sobre el conjunto de entrenamiento obtenido del conjunto de datos inicial. Cada punto representa un modelo CNN con una configuración concreta de los parámetros usados. Las gráficas muestran precisiones insuficientes para el problema de clasificación de tres clases que se pretende resolver. Aunque los modelos tarden poco en entrenar y algunos consigan obtener precisiones mayores que 0.7, la tendencia es obtener modelos que no consiguen extraer las regiones relevantes de las imágenes de entrada. Se podría decir que se “ pierden” o “ confunden” con las imágenes usadas para entrenar los modelos.

¹⁵ También llamadas “epoch”

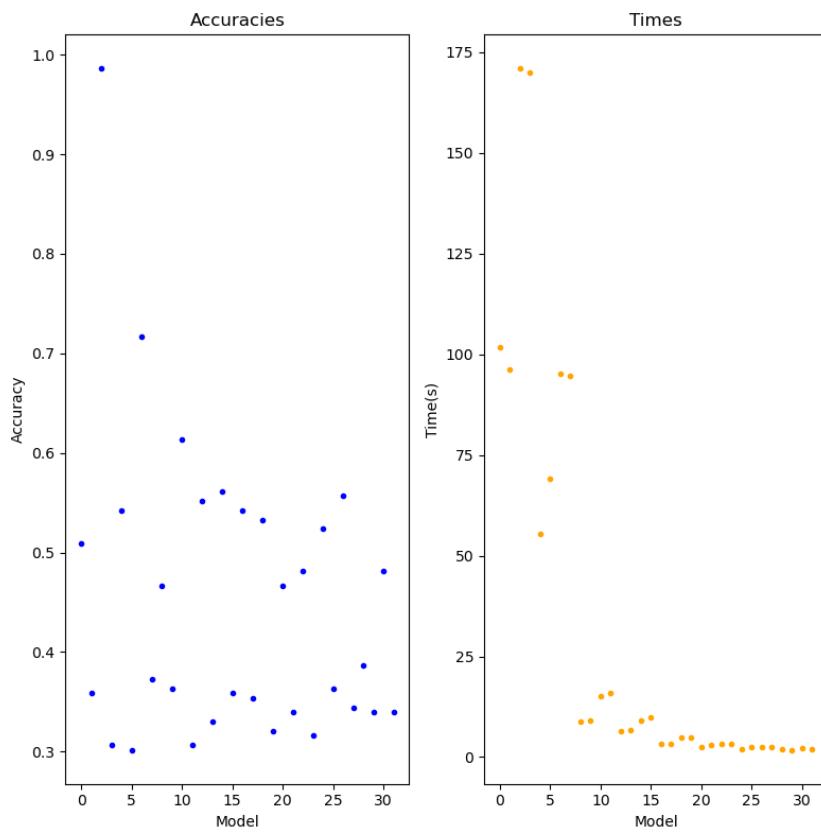


Ilustración 19.- Tiempos y rendimientos de los modelos CNN del conjunto de datos inicial tras el proceso de entrenamiento

El proceso de entrenamiento para los modelos de clasificación en tres clases del conjunto de datos final se va a realizar en 20 épocas y el entrenamiento para los modelos de clasificación en siete clases se va a realizar en 40 épocas. El número de épocas considerado es suficiente para que los modelos obtengan las características necesarias de cada clase. Un número mayor de épocas puede conducir a los modelos a un sobreaprendizaje, consiguiendo una muy buena precisión sobre los datos de entrenamiento, pero muy mala sobre los datos de prueba.

A continuación, se muestra una gráfica con los tiempos y las precisiones obtenidas sobre el conjunto de entrenamiento del conjunto de datos final para el problema de clasificación de tres clases. Cada punto representa un modelo CNN con una configuración concreta de los parámetros usados:

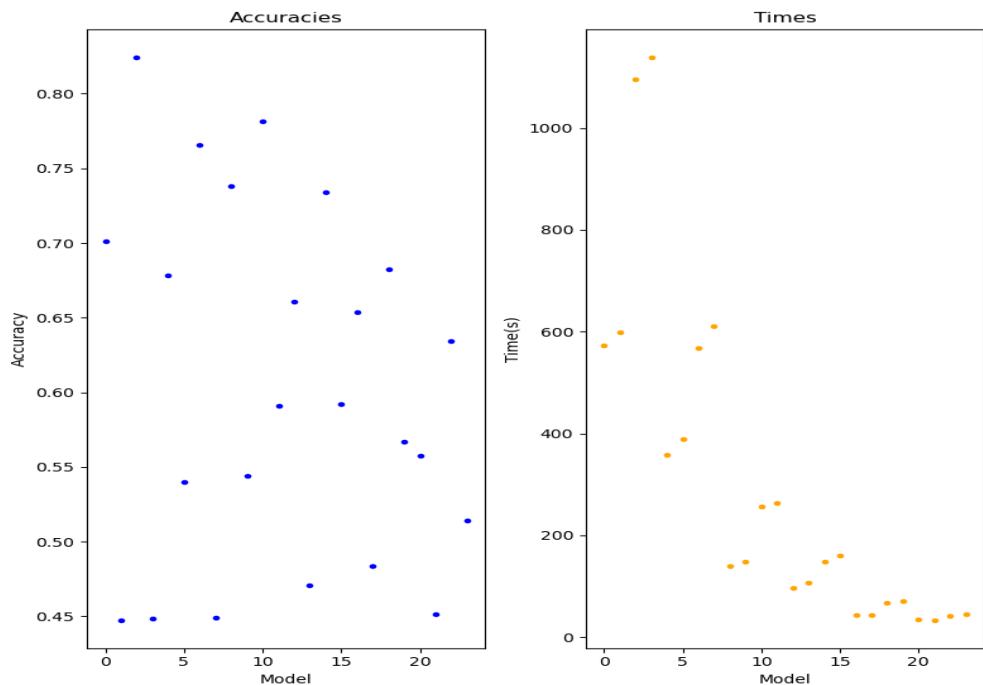


Ilustración 20.- Tiempos y rendimientos de los modelos CNN de clasificación de tres tipos de emociones tras el proceso de entrenamiento

Se puede observar que se consiguen resultados bastante aceptables en tiempos relativamente pequeños, como es el caso del modelo número 11, que a pesar de no ser el que mejor precisión consigue, obtiene la segunda mejor precisión (0,78) en un tiempo mucho menor (4 minutos frente a 18 minutos del mejor modelo) Esto sugiere que no necesariamente un mayor tamaño de los datos de entrada implica una mayor precisión, pudiendo usar menos información en las imágenes para que sean más rápidos a la hora de entrenar y predecir nuevas muestras. Llama la atención que la mayoría de los modelos no consiguen una precisión aceptable, es decir, mayor del 60%.

A continuación, se muestra una gráfica con los tiempos y las precisiones obtenidas sobre el conjunto de entrenamiento para el problema de clasificación de siete clases de la aproximación final. Cada punto representa un modelo CNN con una configuración concreta de los parámetros usados:

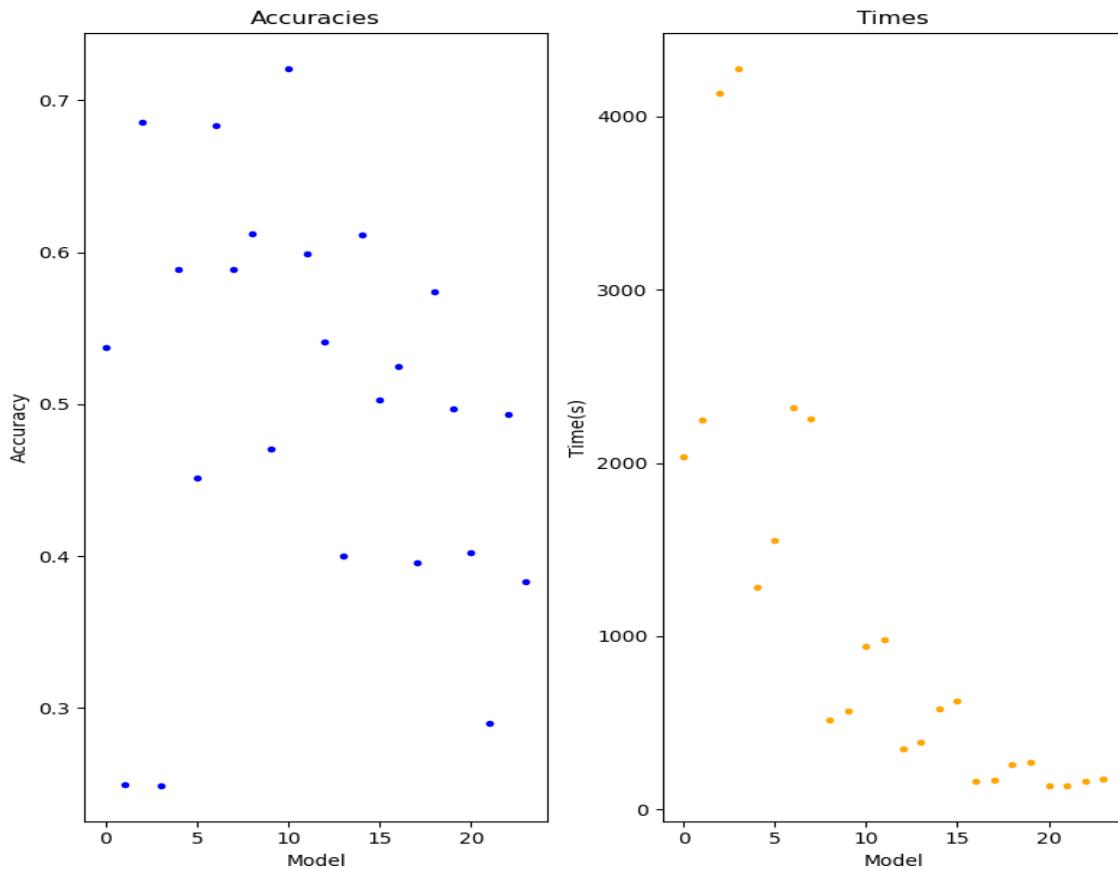


Ilustración 21.- Tiempos y rendimientos de los modelos CNN de clasificación de siete tipos de emociones tras el proceso de entrenamiento

Se puede observar que se consiguen resultados bastante aceptables en tiempos considerables, teniendo en cuenta la gran cantidad de imágenes que forman parte del conjunto de entrenamiento. El mejor modelo obtiene más precisión en mucho menos tiempo que el segundo mejor. Como se comentó anteriormente, trabajar con imágenes más grandes no implica obtener una mayor precisión, porque el modelo puede no detectar correctamente los patrones. Es interesante ver que la mayoría de los modelos consiguen una precisión mayor del 50%. Esto puede deberse a que las características usadas potencian la detección de patrones.

4.2.1 Mejores modelos CNN obtenidos

A continuación, se muestran las características y los resultados del mejor modelo CNN obtenido para el conjunto de datos inicial:

- Precisión sobre el conjunto de entrenamiento: 0.55
- Precisión sobre el conjunto de prueba: 0.54
- Tamaño de las imágenes de entrada: (96, 96)
- Función de activación: “ReLU”
- Tamaño de los filtros: (2, 2)
- Tamaño de agrupación: (3, 3)

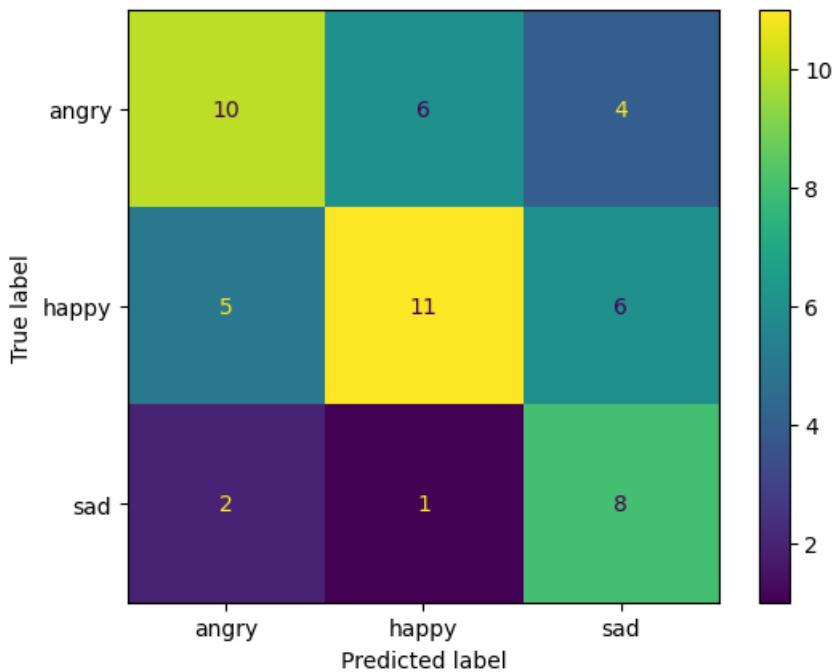


Ilustración 22.- Matriz de confusión del mejor modelo CNN del conjunto de datos inicial

En la matriz de confusión se puede observar que la clase que mejor clasifica es “happy”; y la que peor “sad”, que falla más veces de las que acierta en la predicción. Las clases que menos confunde son “sad” y “happy”, por lo que es capaz de extraer alguna característica diferente de cada clase.

A continuación, se muestran las características y los resultados del mejor modelo obtenido para el problema de clasificación de tres tipos de emociones del conjunto de datos final:

- Precisión sobre el conjunto de entrenamiento: 0.73
- Precisión sobre el conjunto de prueba: 0.73

- Tamaño de las imágenes de entrada: (48, 48)
- Función de activación: “ReLU”
- Tamaño de los filtros: (6, 6)
- Tamaño de agrupación: (3, 3)

En la matriz de confusión se puede observar que la clase que mejor clasifica es “happy”; y la que peor, “angry” al igual que el modelo MLP. También coincide que las clases que menos confunde son “angry” con “happy”, pero ahora las confunde ligeramente menos. Este modelo, a diferencia del MLP comentado anteriormente, las clases que más confunde entre sí son “angry” y “sad”, aunque no de forma general. Esto lleva a pensar que el modelo ha conseguido extraer las características importantes de las imágenes para realizar la clasificación.

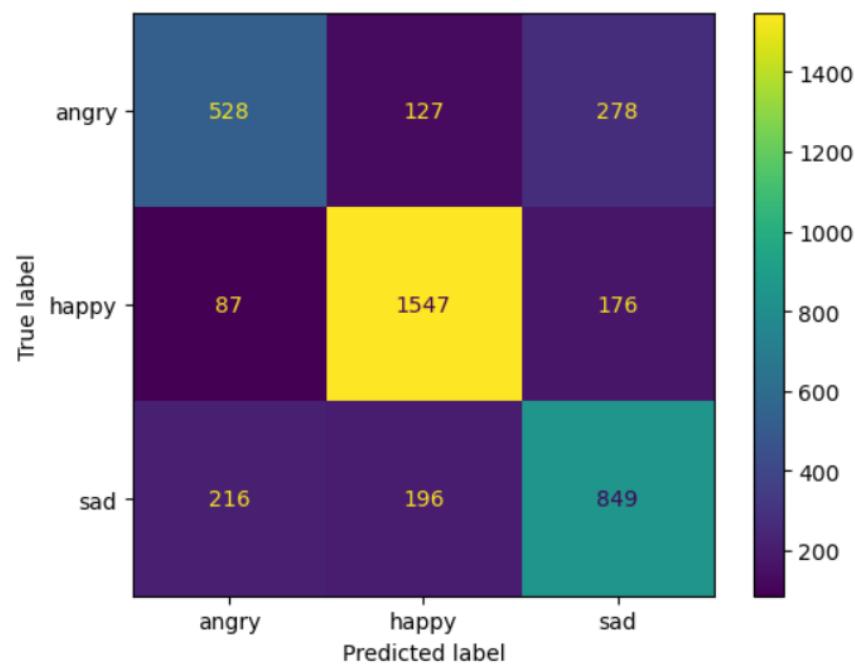


Ilustración 23.- Matriz de confusión del mejor modelo CNN de clasificación de tres tipos de emociones

A continuación, se muestran las características y los resultados del mejor modelo obtenido para el problema de clasificación de siete tipos de emociones del conjunto de datos final:

- Precisión sobre el conjunto de entrenamiento: 0.6118
- Precisión sobre el conjunto de prueba: 0.56
- Tamaño de las imágenes de entrada: (48, 48)
- Función de activación: “ReLU”
- Tamaño de los filtros: (6, 6)
- Tamaño de agrupación: (3, 3)

En comparación con el problema anterior de clasificación de tres emociones, la precisión no ha disminuido tanto, teniendo en cuenta que se han añadido cuatro clases más. En la matriz de confusión se puede observar que la clase que mejor clasifica es “happy”; y la que peor, “disgust”, aunque el porcentaje de acierto sea mayor que el de fallo. Las clases que más confunde entre sí son “sad” con “fear”; y las que menos, “disgust” con “happy”, “neutral” y “surprise”. Esto se puede deber a que el número de imágenes para la clase “disgust” es insuficiente para poder extraer las características relativas a esa clase, pero en general el modelo ha extraído características para poder discriminar cada una de las siete clases de manera aceptable.

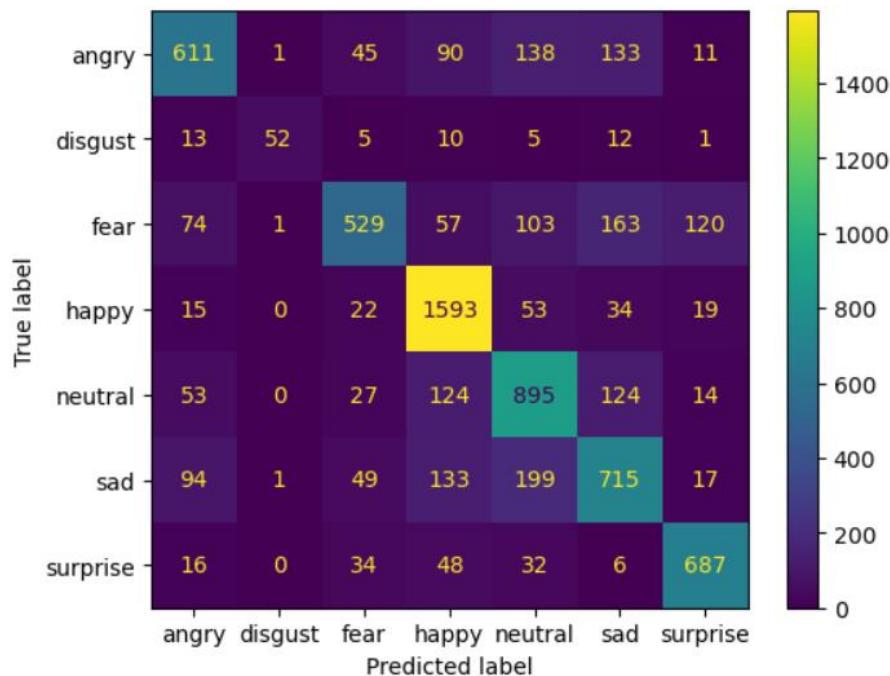


Ilustración 24.- Matriz de confusión del mejor modelo CNN de clasificación de siete tipos de emociones

Capítulo 5 - Explicabilidad

En este capítulo se van a aplicar distintos métodos de explicabilidad sobre imágenes de la librería ALIBI explain, con el objetivo de mostrar las regiones de la imagen que resultan determinantes a la hora de realizar la predicción por parte del modelo. El principal objetivo no es solamente determinar qué método de explicabilidad es más apto para este conjunto de datos, sino también cuál es el más entendible por los humanos.

En primer lugar, se cargan los mejores modelos CNN de clasificación de tres y siete tipos de emociones obtenidos en el apartado anterior. Para obtener la información del tamaño de las imágenes de entrada, se consulta la primera capa del modelo¹⁶. Una vez cargado, se procede a cargar el conjunto de datos original (FER2013) redimensionando las imágenes al tamaño determinado por el modelo. Se ha dividido el conjunto de datos en dos subconjuntos: uno mayoritario que contiene casi todas las muestras de las imágenes, las cuales serán usadas por algunos métodos como GradientSimilarity para mostrar imágenes similares y así explicar la imagen de entrada; y otro minoritario, que contiene alrededor de 100 imágenes en las cuales se visualizarán las explicaciones. Estos subconjuntos serán usados por todos los métodos para poder realizar un estudio más preciso. A continuación, se detallan en subapartados los distintos métodos de explicabilidad aplicados.

5.1 Anchorage

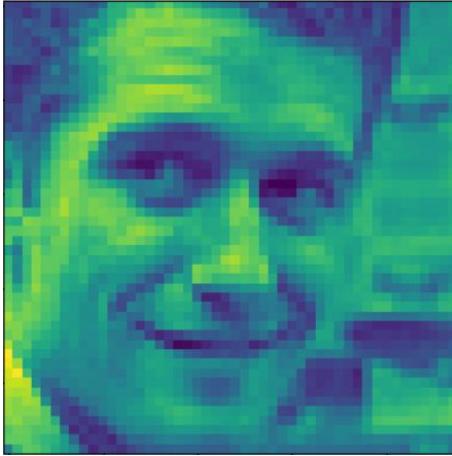
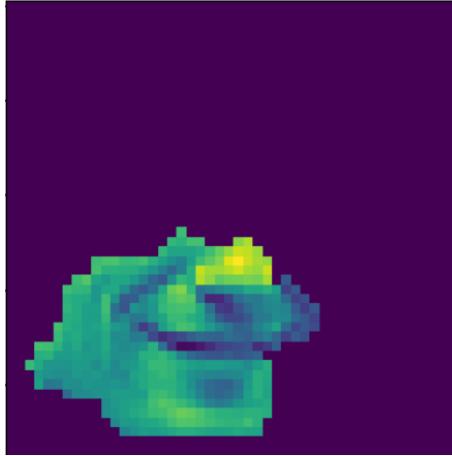
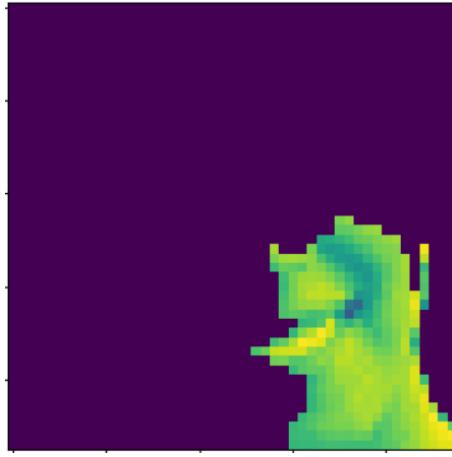
Este método muestra, dada una imagen de entrada, una máscara que contiene las regiones de la imagen que constituyen el ancla. Este método de explicabilidad es el más fácil de interpretar por los humanos, porque simplemente muestra las regiones importantes de la imagen.

¹⁶ También llamada InputLayer

A continuación, se van a mostrar algunos ejemplos representativos del método AnchorImage para el problema de clasificación de tres tipos de emociones aplicado a imágenes del conjunto de datos final.

En las imágenes posteriores se puede observar que el modelo ha conseguido reconocer que en la clase “happy” es importante la forma de la boca, así como las arrugas de su alrededor. Para distinguir de otras clases parecidas como “angry”, en algunas imágenes también enfoca la atención en la forma de los ojos. Cabe destacar que el modelo no ha detectado correctamente la emoción de la última imagen, en gran medida, porque no ha determinado que la información de los ojos era importante al haber encontrado la presencia de dientes, característica que suponemos que considera representativa de la emoción “happy”

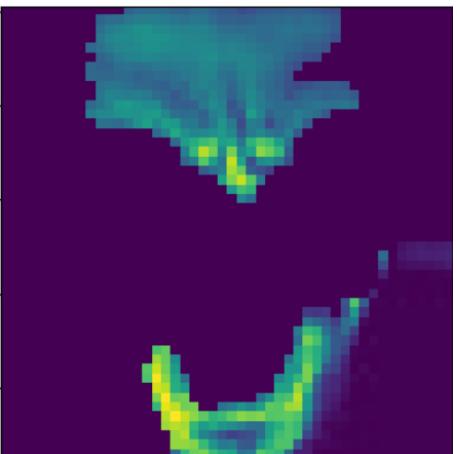
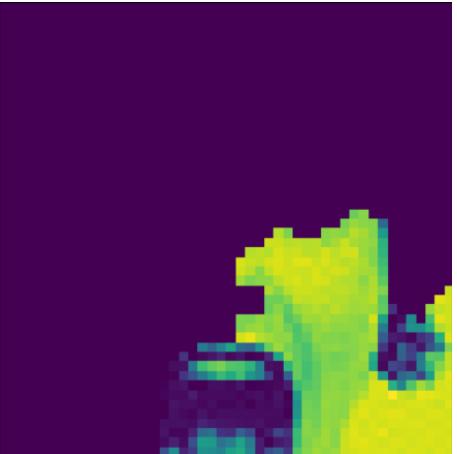
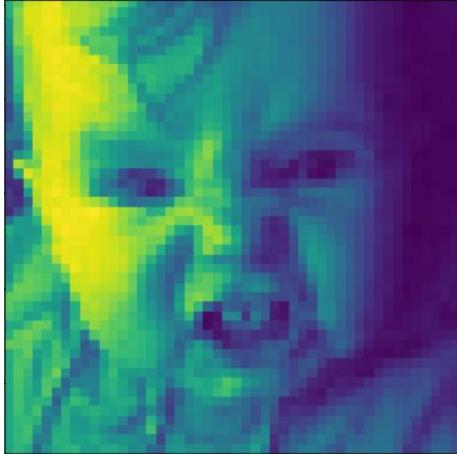
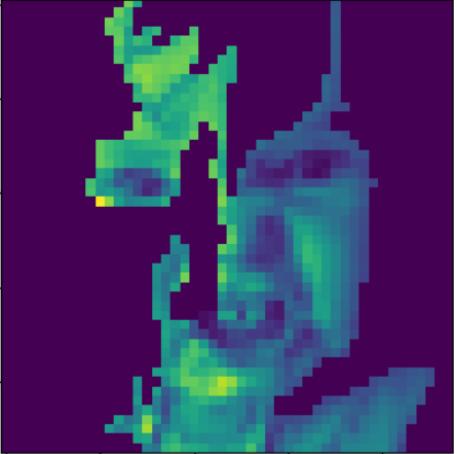
Tabla 8.- Explicabilidad por el método de AnchorImage de imágenes correspondientes a la clase “happy” para el problema de clasificación de tres emociones

		Clase real: “happy” Clase predicha: “happy”
		Clase real: “happy” Clase predicha: “happy”

		Clase real: "happy" Clase predicha: "happy"
		Clase real: "angry" Clase predicha: "happy"

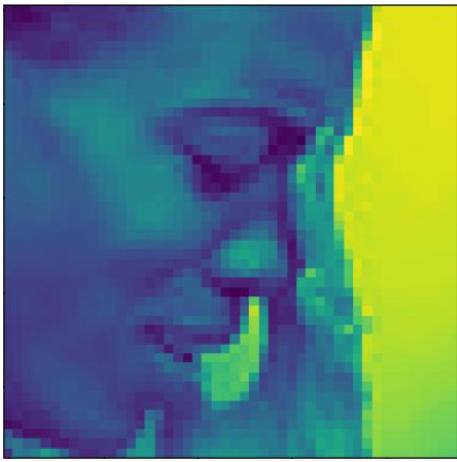
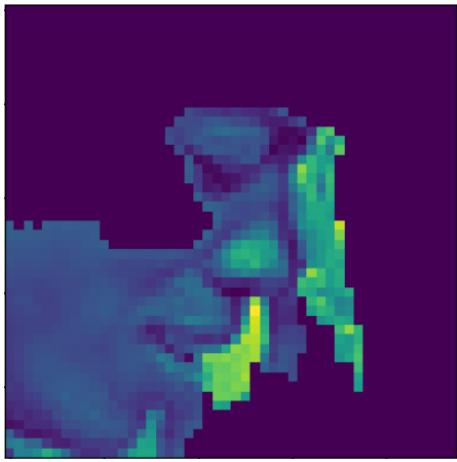
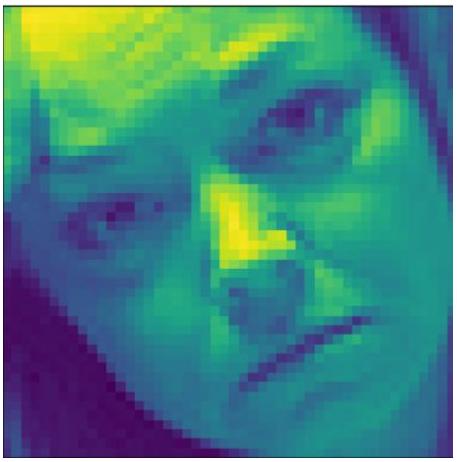
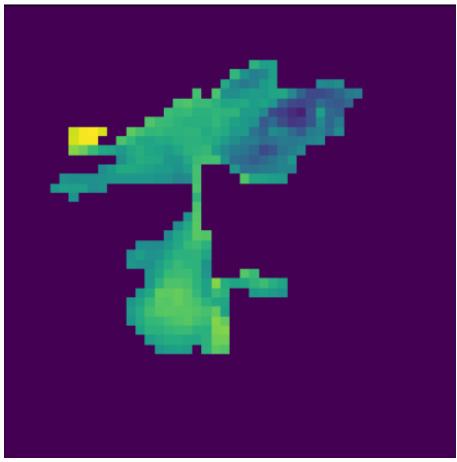
En las imágenes posteriores se puede observar que el modelo ha conseguido reconocer que en la emoción "angry" son importantes las arrugas de la frente y las de alrededor de la boca. Además, cuando estas características son confusas para el modelo y tiene que distinguir "angry" de "happy", suele enfocar la atención en la forma de los ojos y de la boca, como se indica en las últimas dos imágenes.

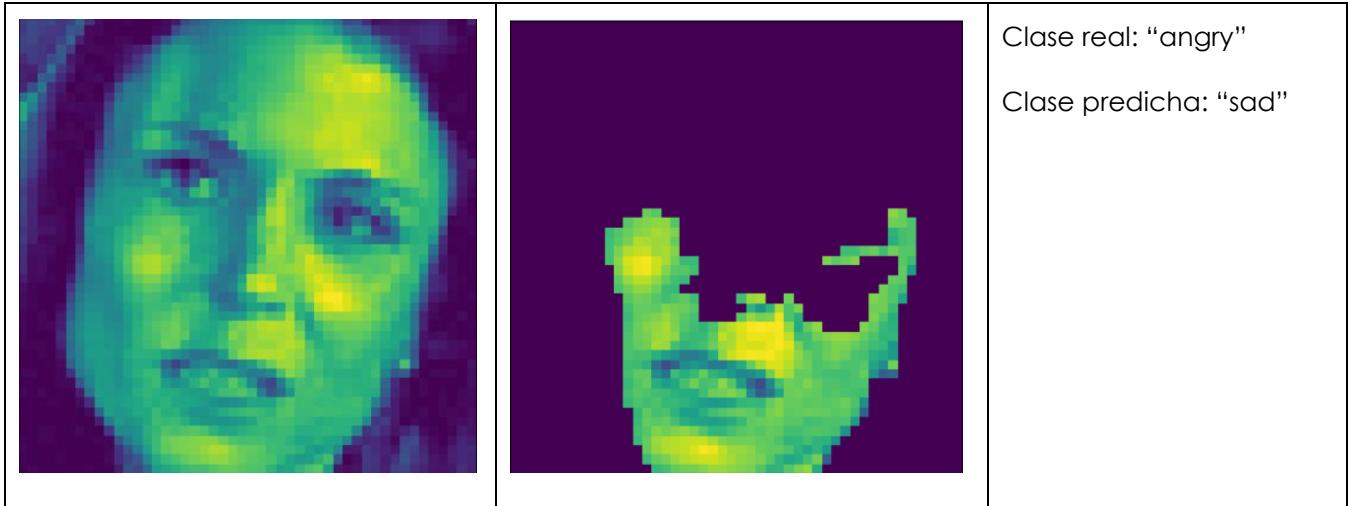
Tabla 9.- Explicabilidad por el método de AnchorImage de imágenes correspondientes a la clase "angry" para el problema de clasificación de tres emociones

		Clase real: "angry" Clase predicha: "angry"
		Clase real: "angry" Clase predicha: "angry"
		Clase real: "angry" Clase predicha: "angry"

En las imágenes que se presentan a continuación, se puede observar que el modelo ha conseguido reconocer que para la emoción "sad" la forma de las cejas juega un papel fundamental, así como la postura de la cara, que tiende a estar más torcida que en otras emociones consideradas en este problema. De la misma manera que las anteriores emociones, a veces necesita obtener información acerca de la forma de la boca y ojos para discriminar de otras clases. Cabe destacar que en la última imagen el modelo ha predicho erróneamente "sad", pudiéndose deber a que no ha determinado que la información de las cejas era importante.

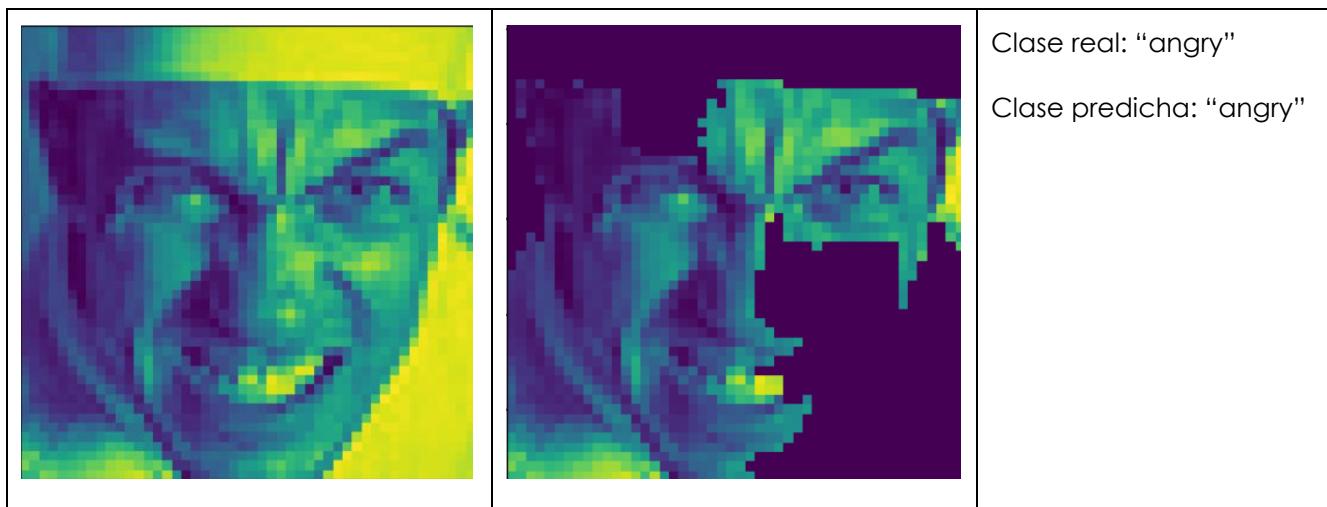
Tabla 10.- Explicabilidad por el método de Anchormage de imágenes correspondientes a la clase "sad" para el problema de clasificación de tres emociones

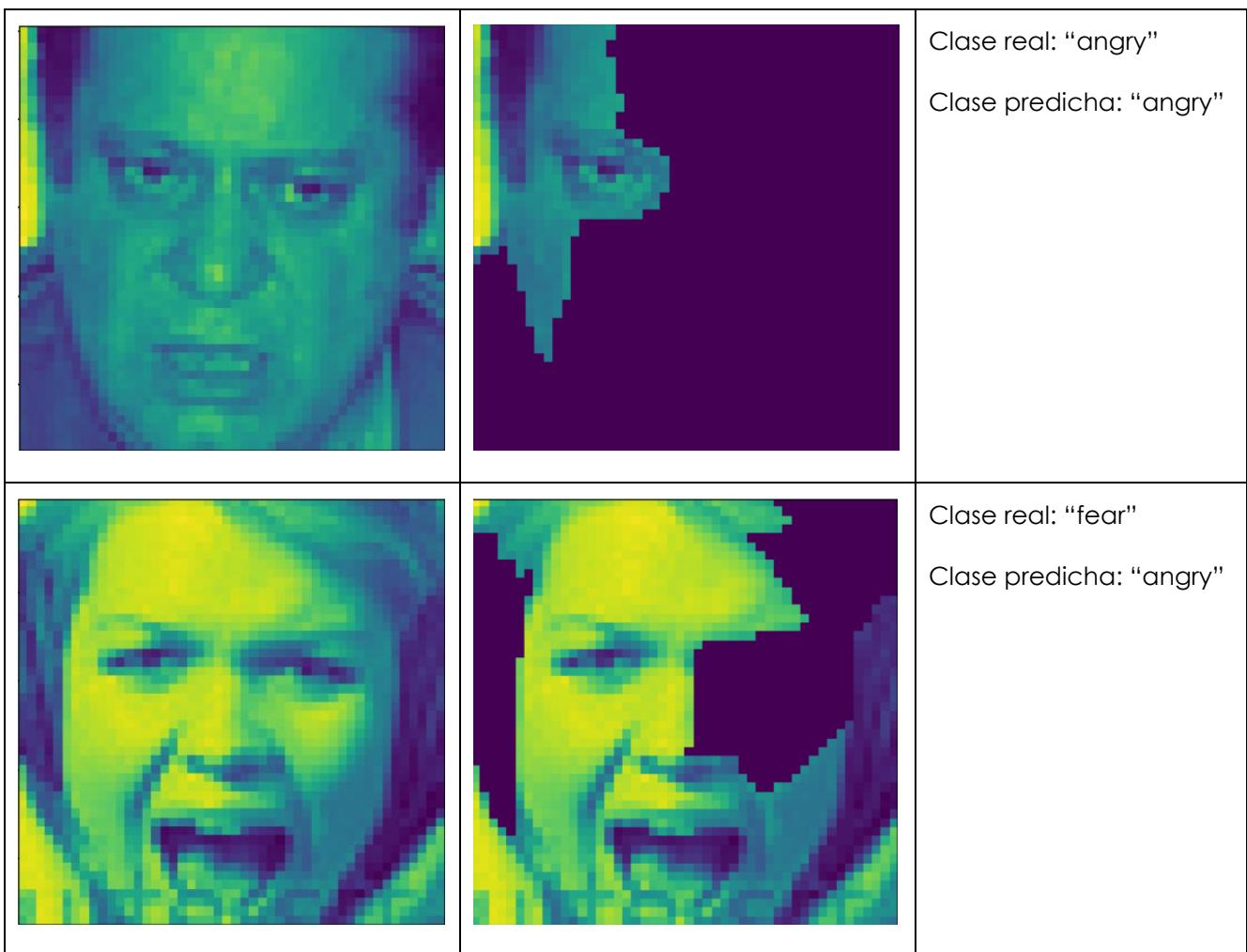
		Clase real: "sad" Clase predicha: "sad"
		Clase real: "sad" Clase predicha: "sad"



A continuación, se van a mostrar algunos ejemplos representativos del método AnchorImage para el problema de clasificación de siete tipos de emociones aplicado a imágenes del conjunto de datos final.

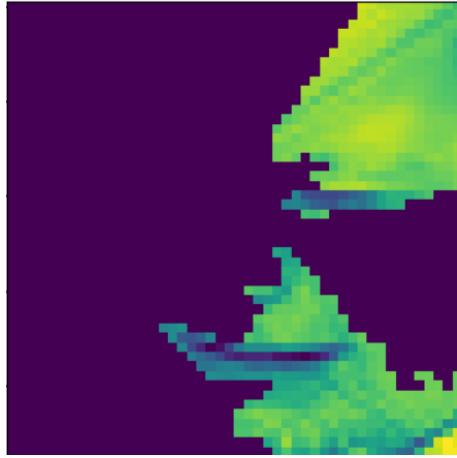
Tabla 11.- Explicabilidad por el método de AnchorImage de imágenes correspondientes a la clase "angry" para el problema de clasificación de siete emociones





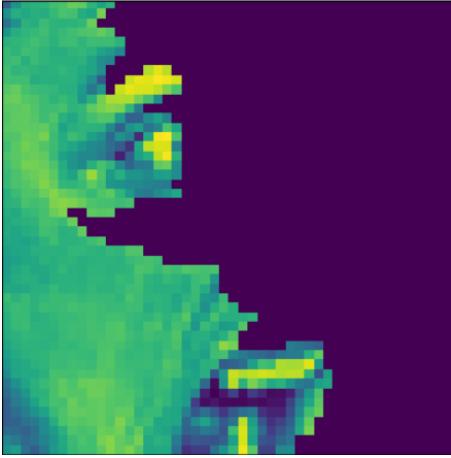
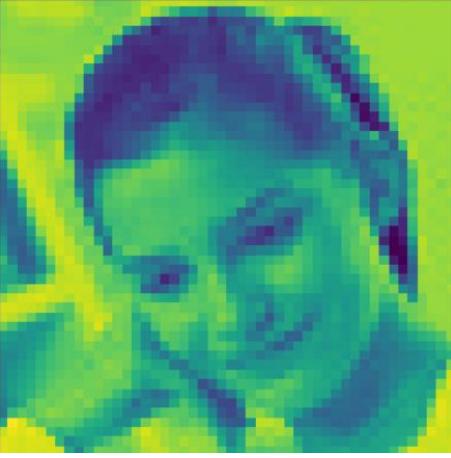
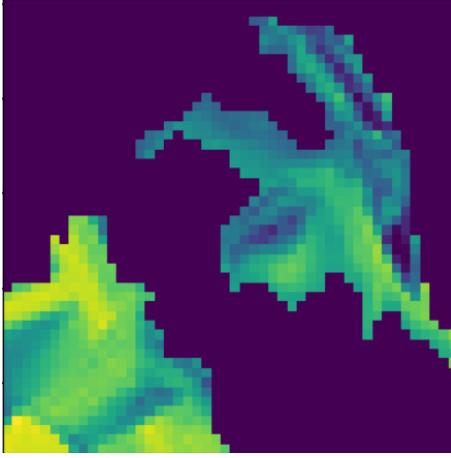
Lo que diferencia la emoción “angry” de “surprise” es la forma de los ojos. En ambas emociones la boca está abierta, pero los ojos de la clase “angry” no están totalmente abiertos. La forma de las cejas también juega un papel importante a la hora de predecir esta emoción. Se puede observar que en la última imagen el modelo ha tenido en cuenta los ojos y la boca a la hora de predecir, pero no ha acertado y ha predicho “angry” porque ha determinado que la boca está demasiado abierta.

Tabla 12.- Explicabilidad por el método de AnchorImage de imágenes correspondientes a la clase "disgust" para el problema de clasificación de siete emociones

		Clase real: "disgust" Clase predicha: "disgust"
		Clase real: "disgust" Clase predicha: "happy"

Esta emoción tiene características comunes a "sad". Ambas tienen bocas planas o ligeramente abiertas, además de ojos y cejas muy similares. La gran diferencia con la emoción "sad" reside en la forma de la boca. Por lo general, la boca en personas disgustadas no suele estar cerrada y curva hacia abajo, como es natural en personas tristes. En la última imagen, a pesar de que el modelo ha tenido en consideración la forma del ojo, la engañosa forma de la boca ha influido a la hora de clasificar la imagen como "happy"

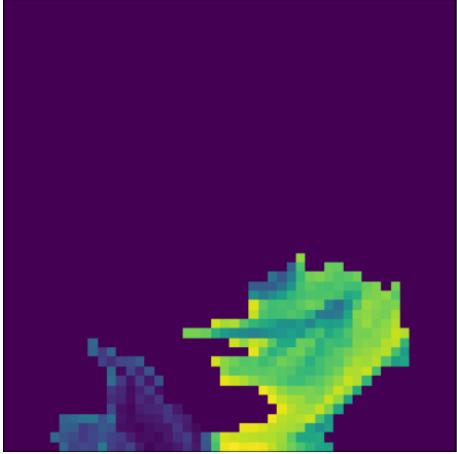
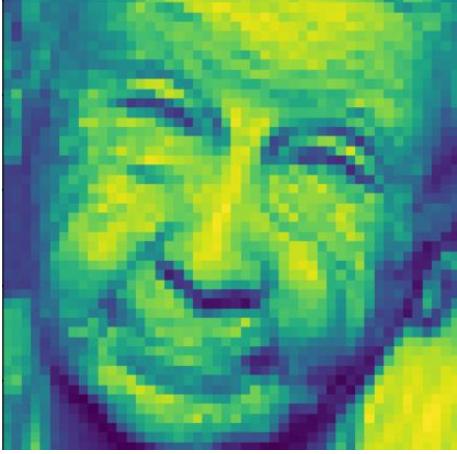
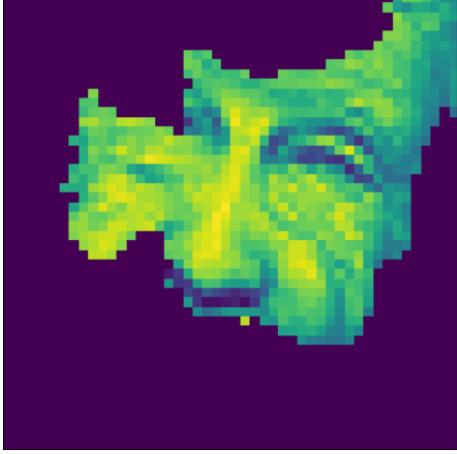
Tabla 13.- Explicabilidad por el método de Anchormage de imágenes correspondientes a la clase "fear" para el problema de clasificación de siete emociones

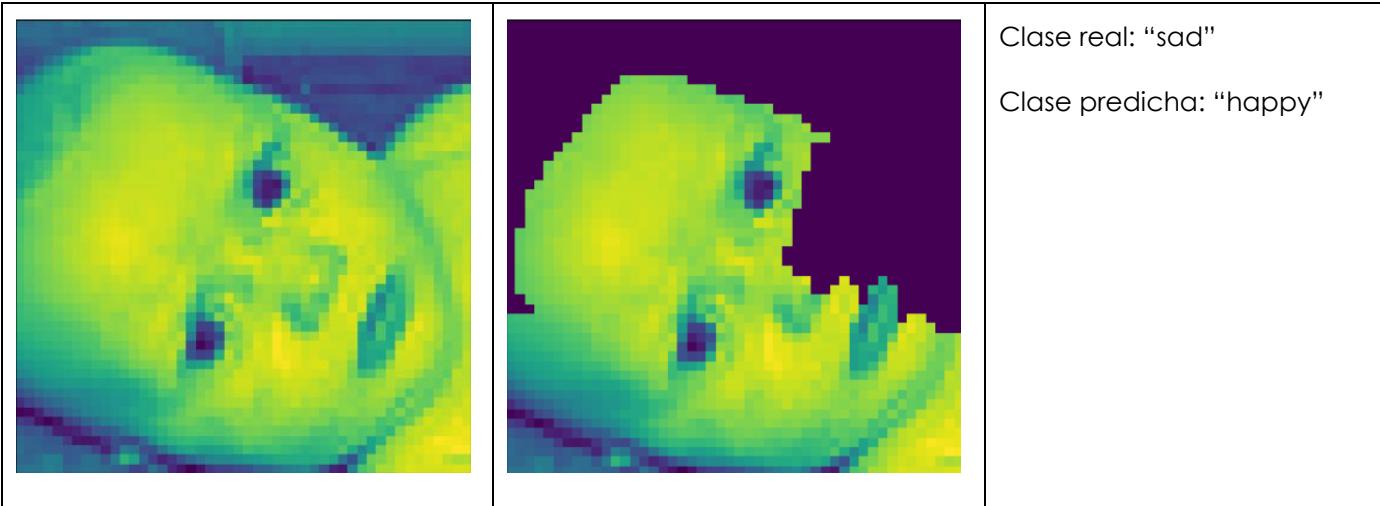
		Clase real: "fear" Clase predicha: "fear"
		Clase real: "fear" Clase predicha: "fear"
		Clase real: "happy" Clase predicha: "fear"

La gran diferencia entre "fear" y "angry" es la forma de la boca. En ambas emociones es común observar ojos y cejas con formas muy parecidas, pero las personas

con miedo suelen tener la boca más cerrada. En la última imagen, el clasificador se ha confundido de emoción. La inclinación de la cabeza ha afectado a la ceja y el ojo, ambas regiones relevantes para el modelo. Se puede pensar que, si hubiera tenido en cuenta la boca, habría predicho correctamente la emoción.

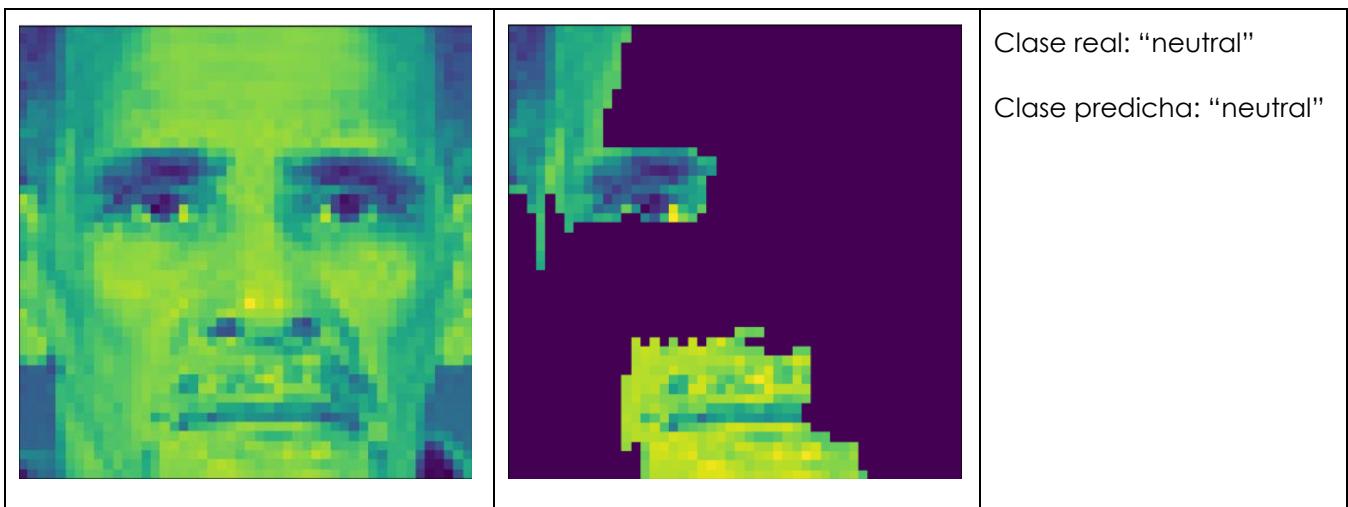
Tabla 14.- Explicabilidad por el método de AnchorImage de imágenes correspondientes a la clase "happy" para el problema de clasificación de siete emociones

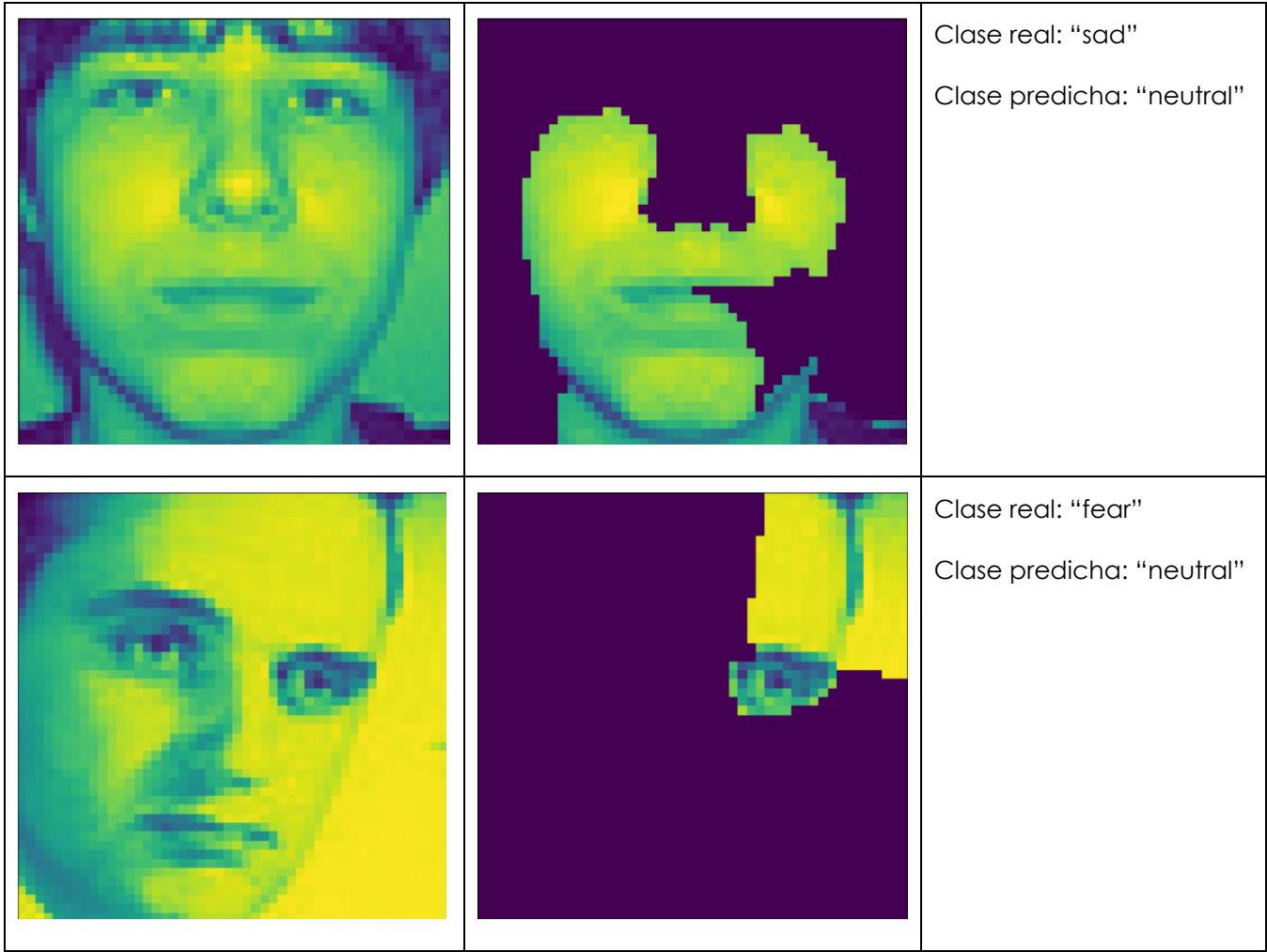
		Clase real: "happy" Clase predicha: "happy"
		Clase real: "happy" Clase predicha: "happy"



Para clasificar la emoción "happy", el modelo tiene en cuenta la forma de la boca, que tiene que ser cóncava hacia arriba, pudiendo contener o no dientes. Si resulta que no puede extraer características suficientes de la boca, presta atención a los ojos, que tienen forma curva hacia abajo, tal y como se puede ver en la segunda imagen. En la última imagen se puede observar una mala predicción por parte del modelo, y se puede deber a que solamente ha tenido en cuenta la mitad de la boca que, confusamente, parece estar curva por la inclinación de la cabeza.

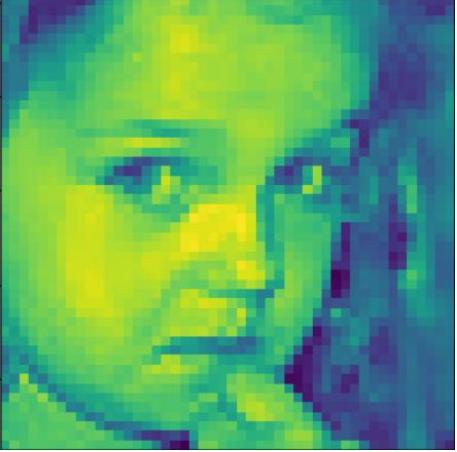
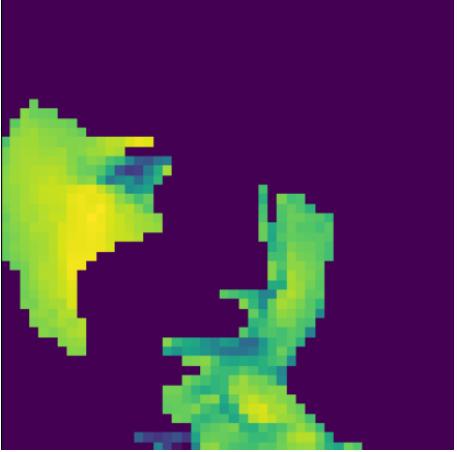
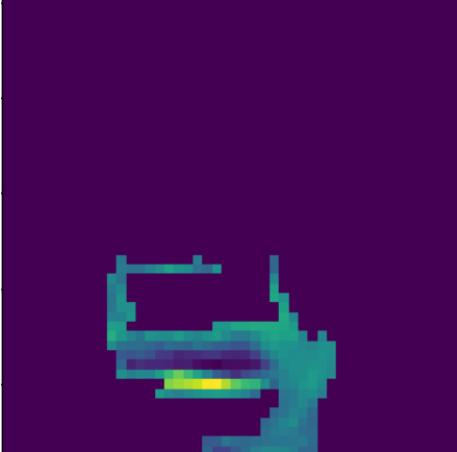
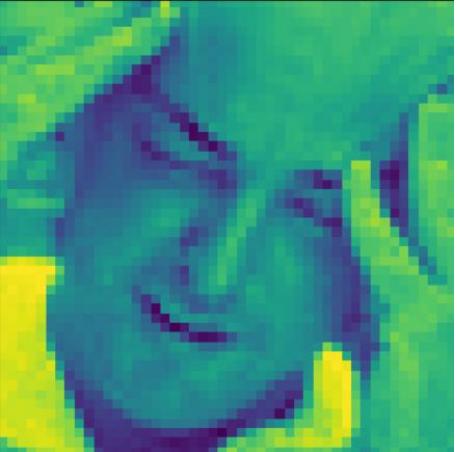
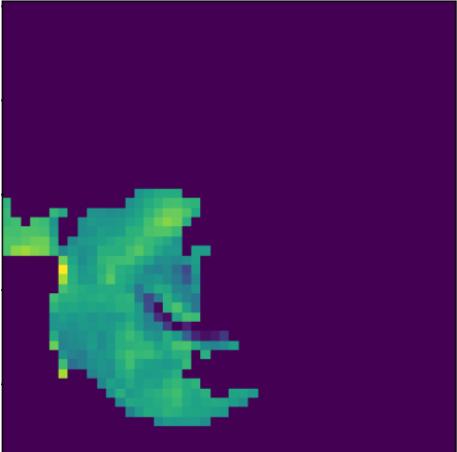
Tabla 15.- Explicabilidad por el método de Anchormage de imágenes correspondientes a la clase "neutral" para el problema de clasificación de siete emociones





Para predecir "neutral", el modelo tiene que detectar una boca cerrada y totalmente recta, al igual que las cejas. El modelo ha fallado la clasificación de la segunda imagen porque ha tenido en cuenta la forma recta de la boca, pero no los ojos. La última imagen resulta engañosa al clasificador, porque muestra a una persona con miedo que no tiene la boca abierta. El modelo se ha fijado en la forma de una ceja y de un ojo, y ha predicho "neutral"

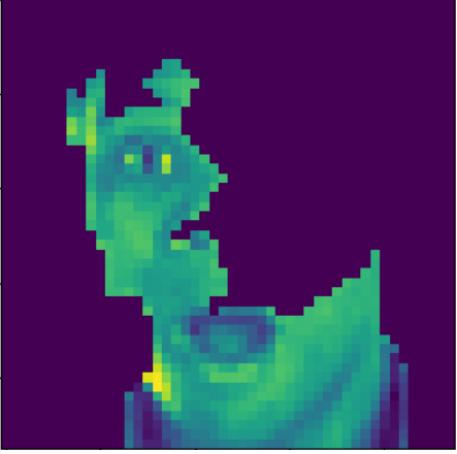
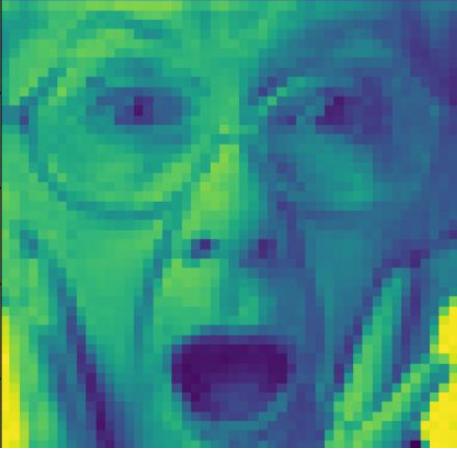
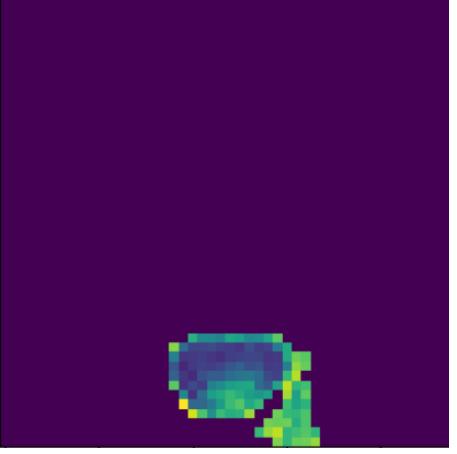
Tabla 16.- Explicabilidad por el método de AnchorImage de imágenes correspondientes a la clase “sad” para el problema de clasificación de siete emociones

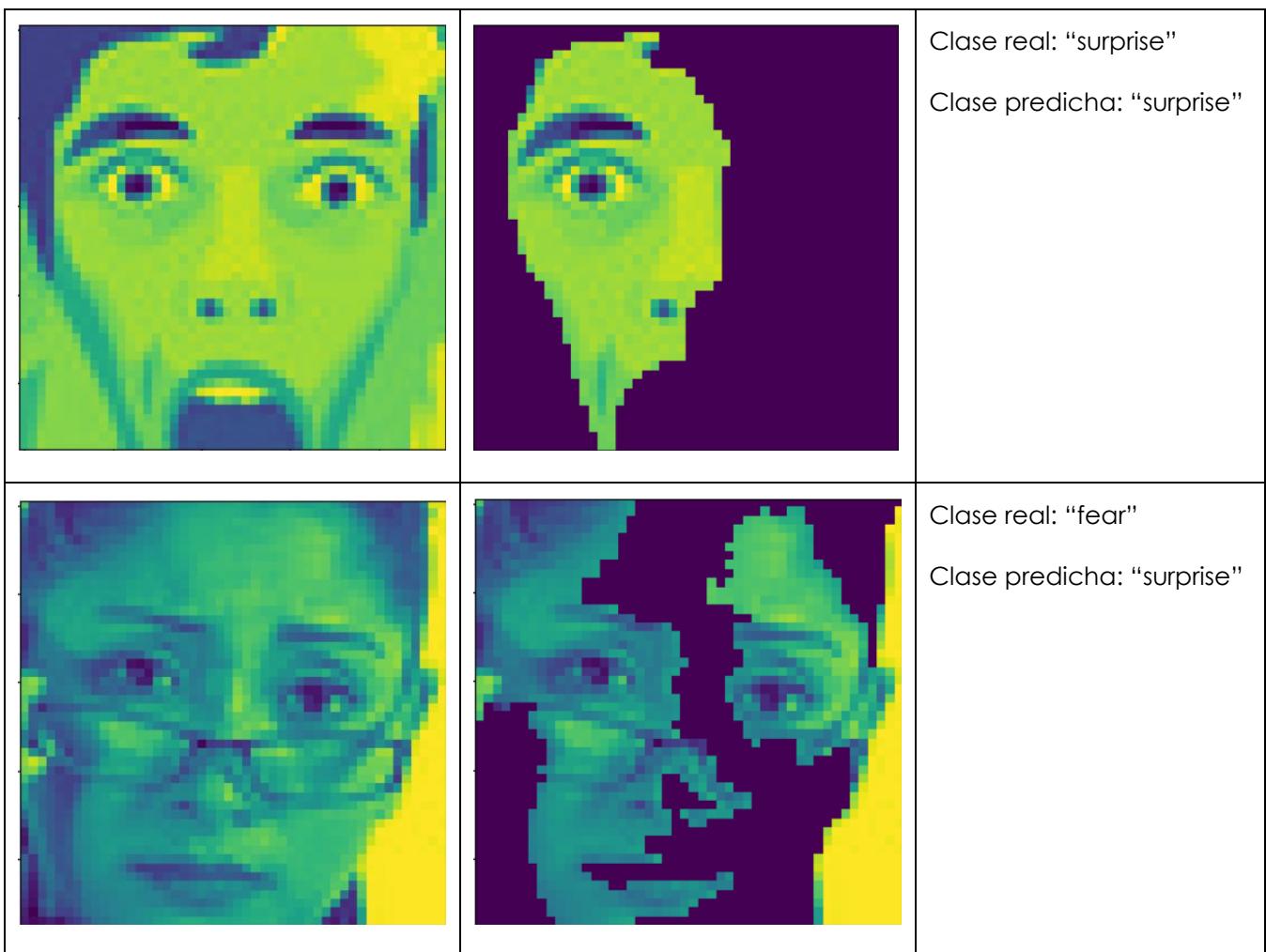
		Clase real: “sad” Clase predicha: “sad”
		Clase real: “sad” Clase predicha: “sad”
		Clase real: “sad” Clase predicha: “happy”

Para determinar si una imagen representa la emoción “sad”, el clasificador se basa en la forma de la boca, que suele ser cerrada y cóncava hacia abajo o plana y

un poco abierta; en la forma de las cejas, cóncavas hacia abajo de forma pronunciada; y en los ojos. Para la última imagen, el modelo se ha fijado solamente en la forma de la boca, que realmente encaja con la clase "happy" debido a la inclinación de la cabeza; pero se puede llegar a pensar que, si hubiera fijado el ancla en los ojos, tal vez habría predicho otra emoción por el simple motivo de que su patrón no encaja con la emoción "sad"

Tabla 17.- Explicabilidad por el método de AnchorImage de imágenes correspondientes a la clase "surprise" para el problema de clasificación de siete emociones

		Clase real: "surprise" Clase predicha: "surprise"
		Clase real: "surprise" Clase predicha: "surprise"



Se puede observar que el modelo tiende a reconocer la emoción "surprise" en una imagen cuando fija el ancla en una boca redonda abierta, o cuando lo fija en un ojo, al igual que la boca, muy abierto y redondo. En la última imagen cabe destacar que el modelo no ha predicho bien la emoción. Esto se debe a que ha fijado el ancla en los ojos y en la forma de la cara, excluyendo la boca. Si hubiera considerado la boca cerrada, probablemente hubiera acertado la emoción.

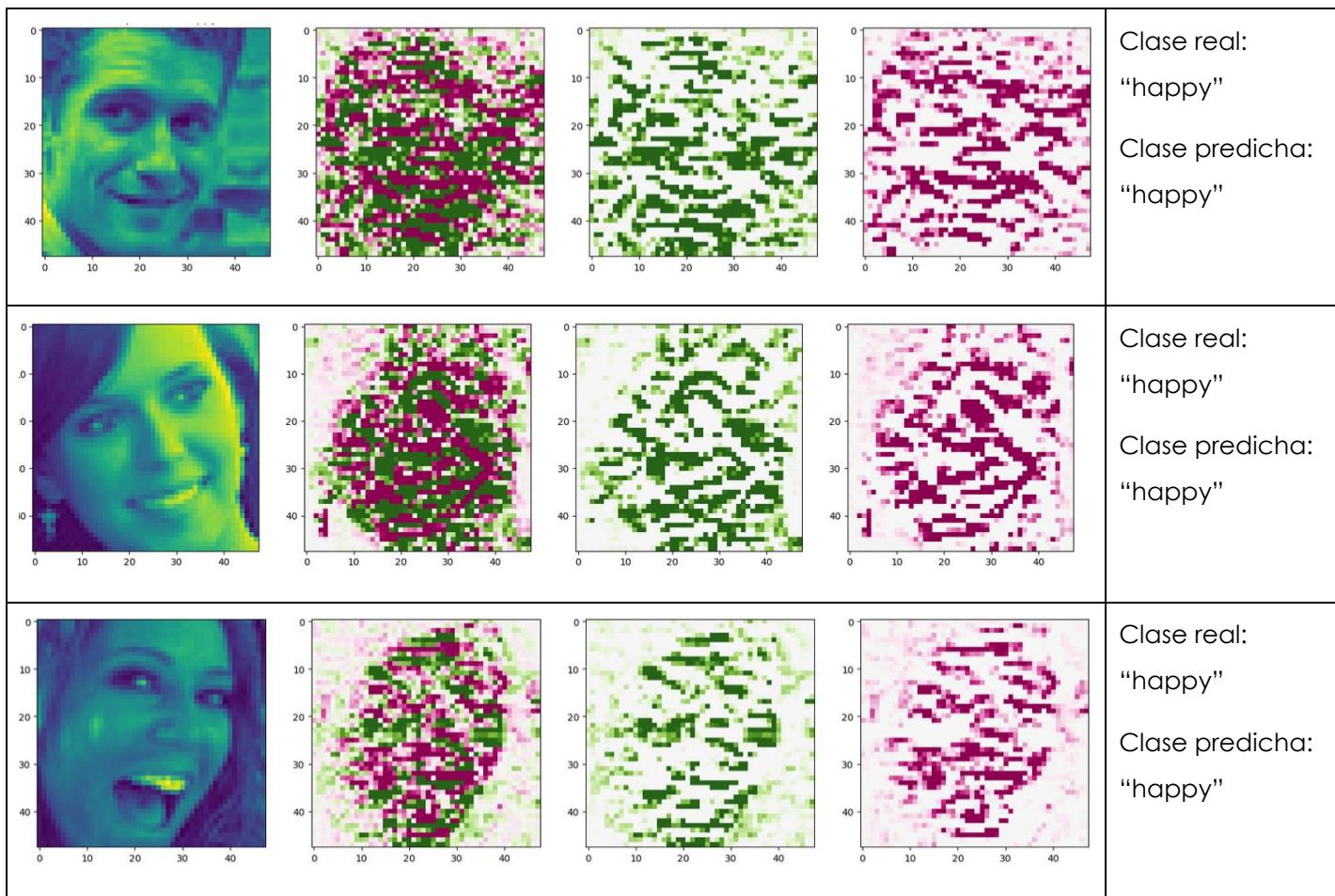
5.2 IntegratedGradients

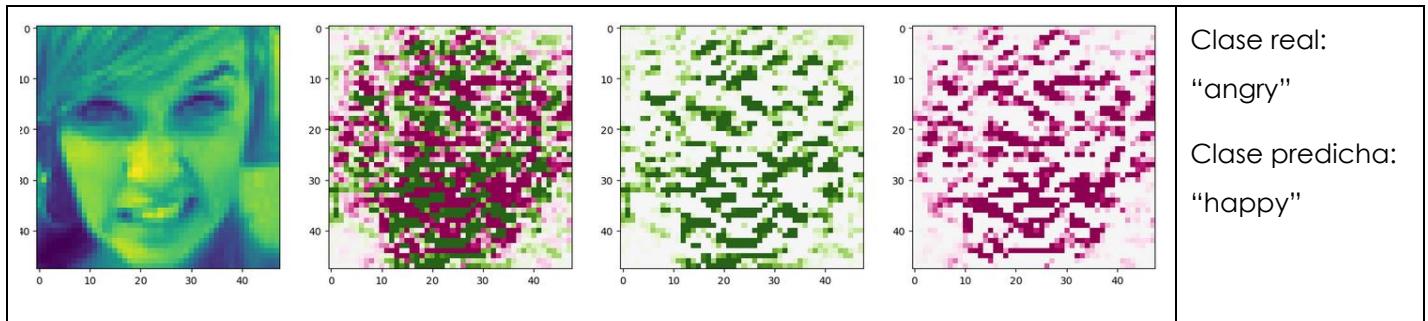
Este método muestra, dada una imagen de entrada, un conjunto de píxeles favorables a la clase predicha, y otros en contra. Este método de explicabilidad es el difícil de interpretar por los humanos, porque lleva mucho tiempo determinar qué regiones favorables está detectando para la imagen a explicar.

A continuación, se van a mostrar algunos ejemplos representativos del método IntegratedGradients para el problema de clasificación de tres tipos de emociones aplicado a imágenes del conjunto de datos final.

En las imágenes posteriores se puede observar que el modelo ha conseguido reconocer que para la emoción “happy” son importantes las regiones cercanas a la boca, por lo que marca si los píxeles de las arrugas de su alrededor son positivas o negativas. En algunas imágenes que representan la emoción “happy” también marca píxeles favorables de la frente. Cabe destacar que el modelo no valora positivamente los píxeles cercanos a la boca en la última imagen, por lo que realiza una predicción incorrecta.

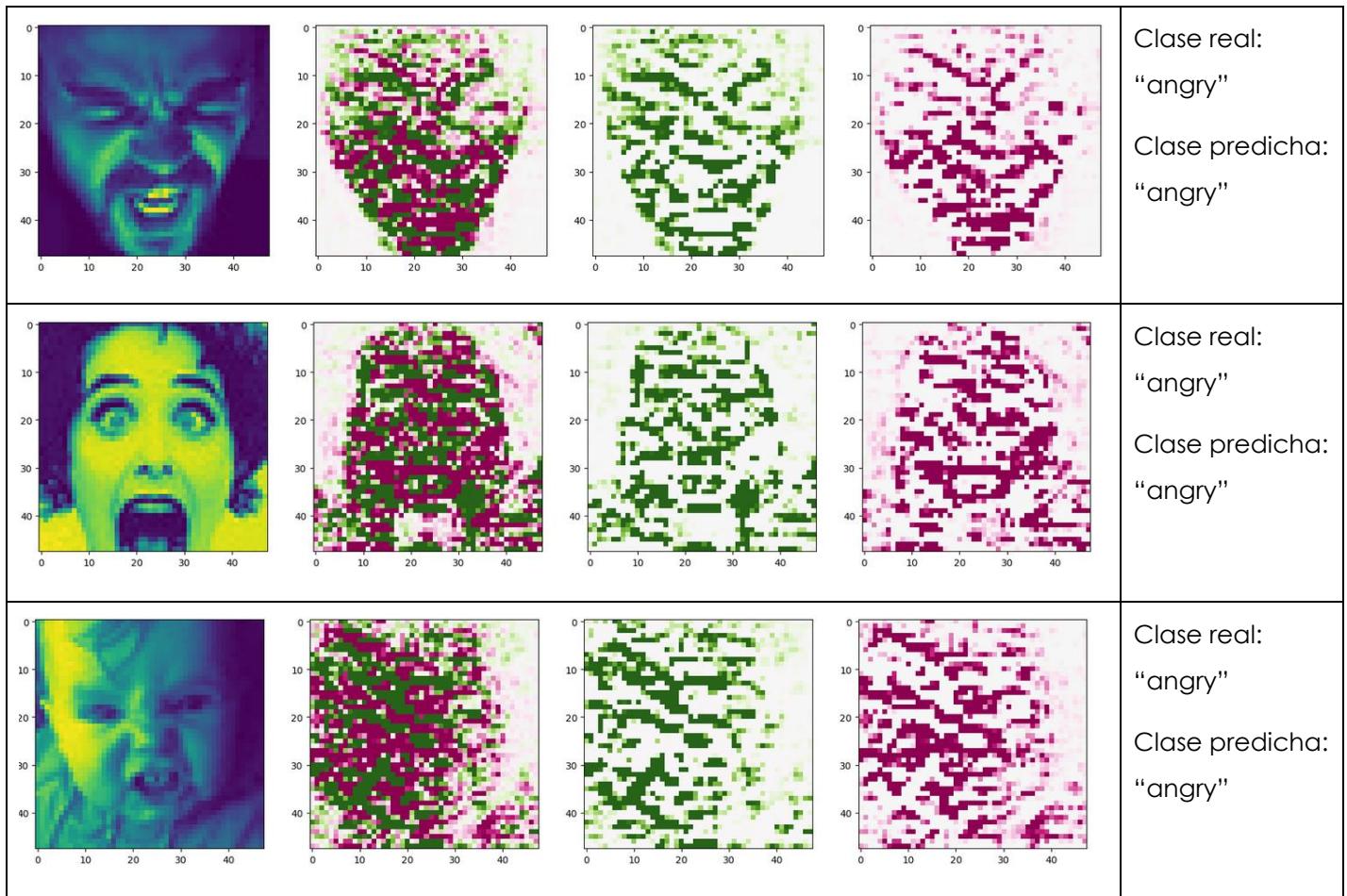
Tabla 18.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase “happy” para el problema de clasificación de tres emociones





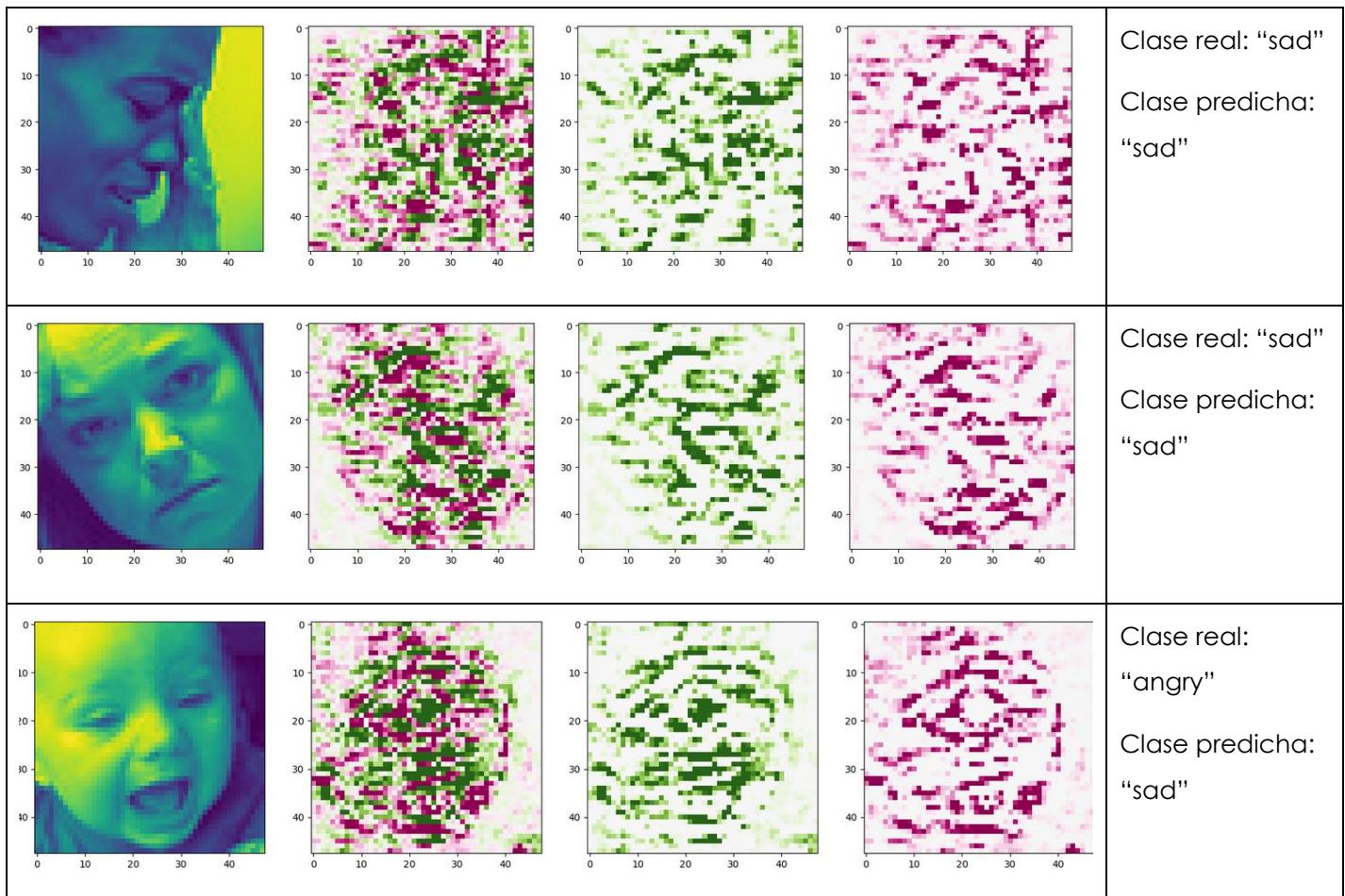
En las imágenes que se presentan a continuación se puede observar que el modelo ha conseguido reconocer que para la emoción "angry" son importantes los píxeles que determinan la forma de las cejas, en concreto, si son cóncavas. Dependiendo de la imagen, valora positivamente la presencia de dientes.

Tabla 19.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase "angry" para el problema de clasificación de tres emociones



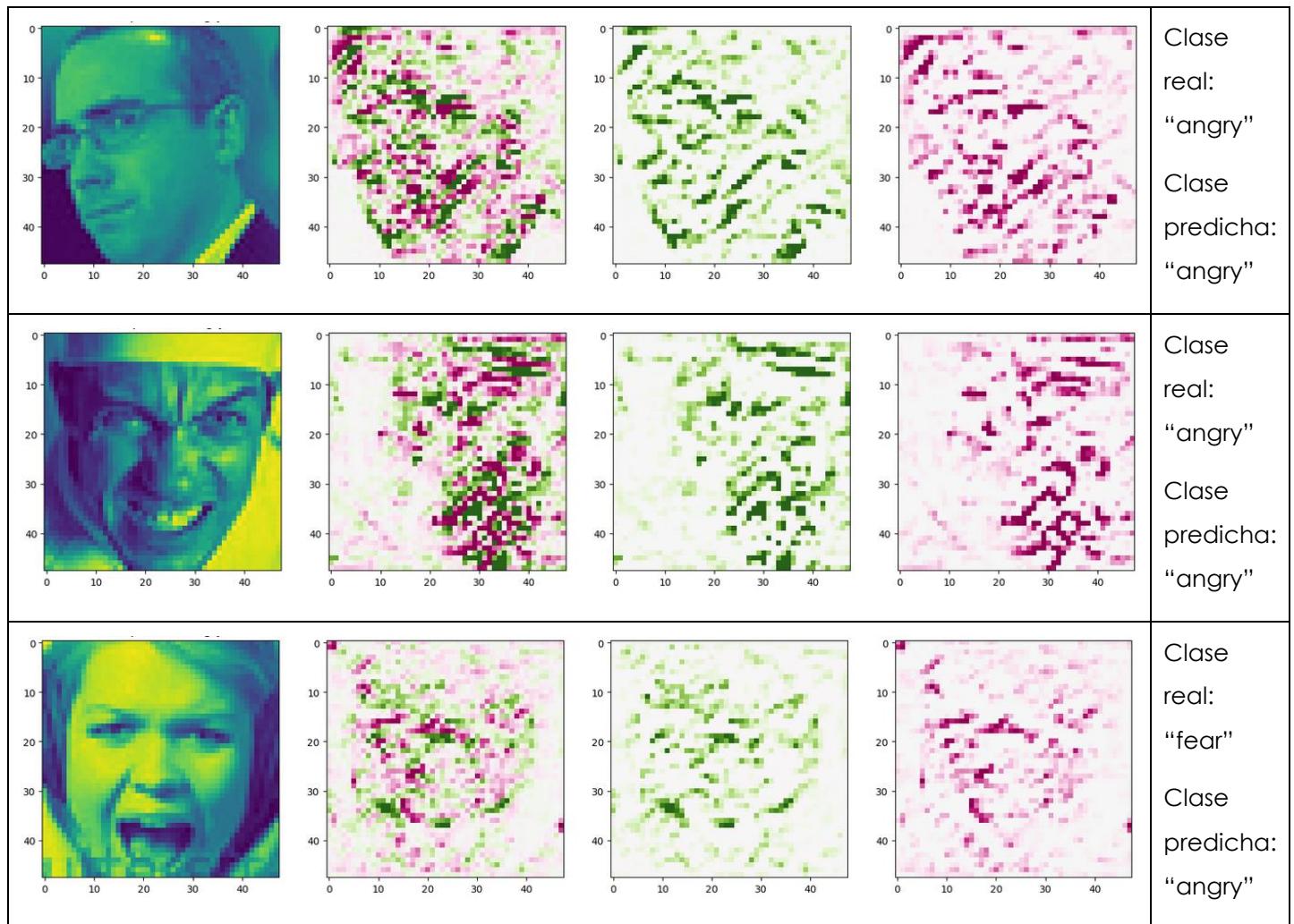
En las imágenes de abajo se puede observar que el modelo ha valorado positivamente aquellos píxeles cercanos a las cejas y ojos para la emoción "sad". De esta forma, determinadas formas de ojos y cejas cobran peso a la hora de determinar la emoción a clasificar. El modelo ha asignado valores negativos a los píxeles que trazan el contorno de la cara en la segunda imagen. Cabe destacar que el modelo no ha asignado valores positivos a la forma de las cejas en la última imagen, por lo que ha realizado una predicción incorrecta.

Tabla 20.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase "sad" para el problema de clasificación de tres emociones



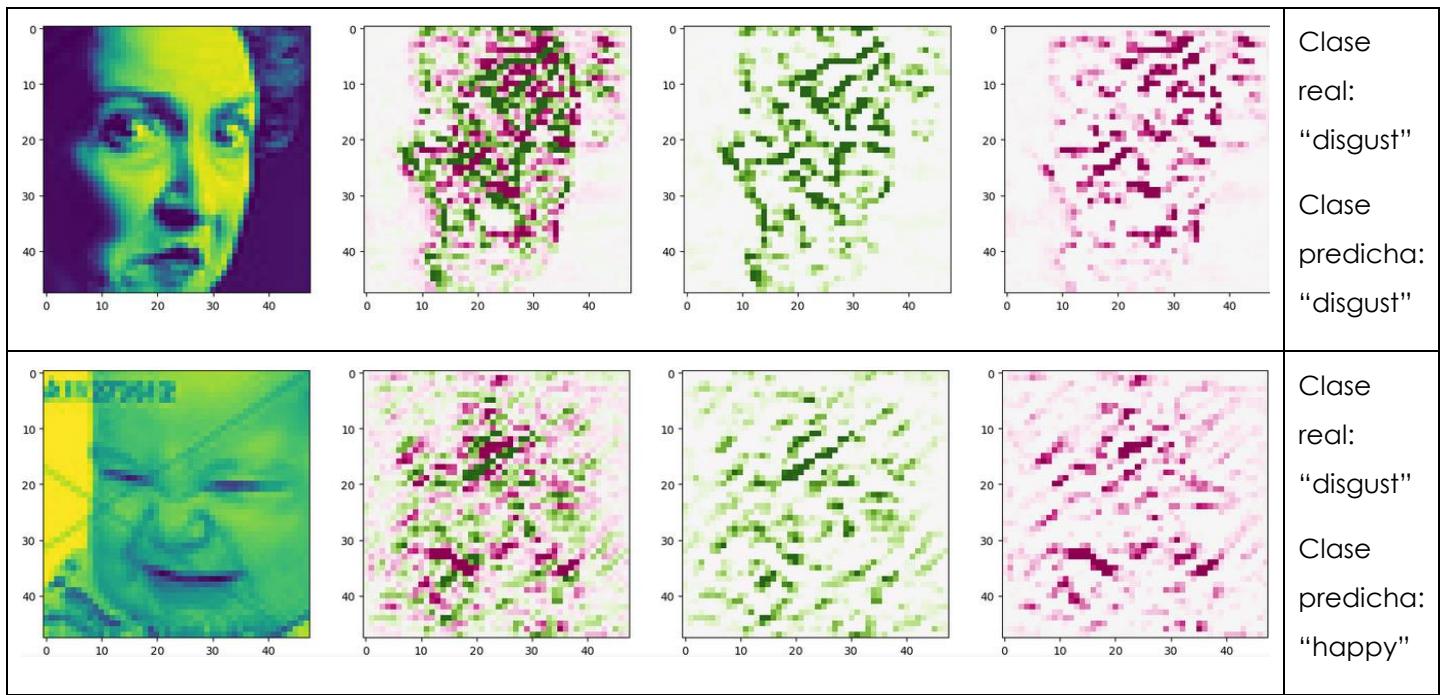
A continuación, se van a mostrar algunos ejemplos representativos del método IntegratedGradients para el problema de clasificación de siete tipos de emociones aplicado a imágenes del conjunto de datos final.

Tabla 21.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase "angry" para el problema de clasificación de siete emociones



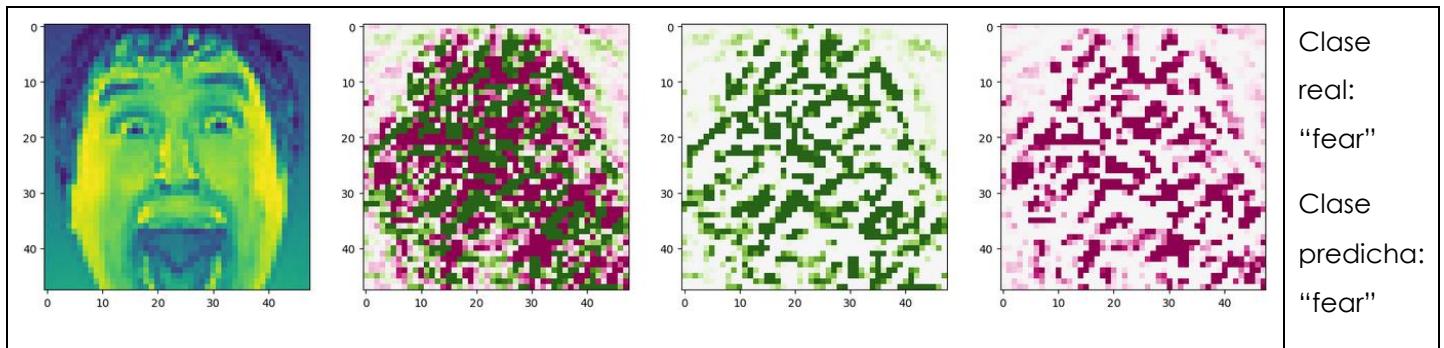
En las imágenes anteriores se puede observar que el modelo ha conseguido reconocer que en la clase "angry" son importantes aquellos píxeles relativos a los ojos, así como las cejas que presentan una forma curva. En la segunda imagen se han valorado negativamente los píxeles de las cejas porque considera que la curva es demasiado pronunciada para una persona enfadada. Tal y como se aprecia en la última imagen, a veces valora positivamente los píxeles de los dientes si no encuentra las características comentadas anteriormente.

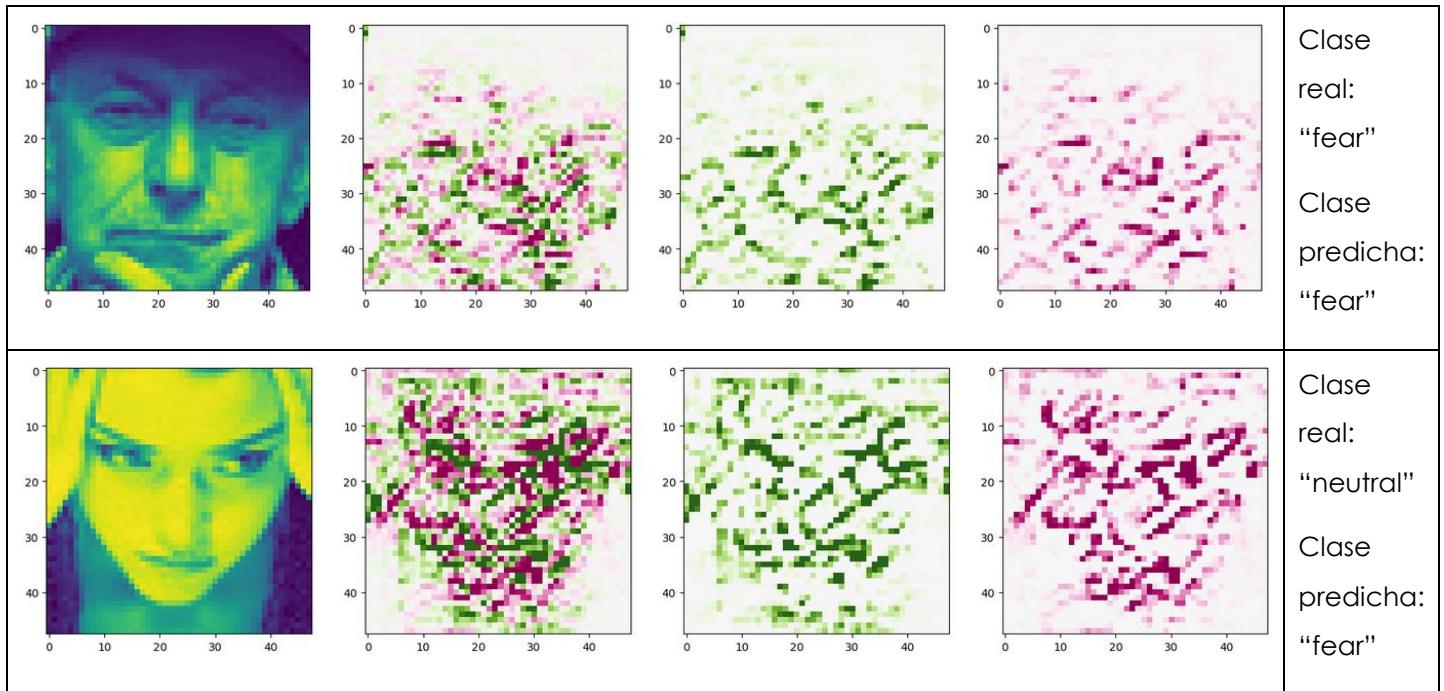
Tabla 22.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase "disgust" para el problema de clasificación de siete emociones



En las imágenes anteriores se puede ver cómo actúa el modelo a la hora de clasificar la emoción "disgust". Valora positivamente los píxeles de las cejas si son cóncavas. Por otro lado, ha aprendido a valorar negativamente las arrugas de alrededor de la boca en ciertas ocasiones, como es el caso de la primera imagen, pero no en la segunda.

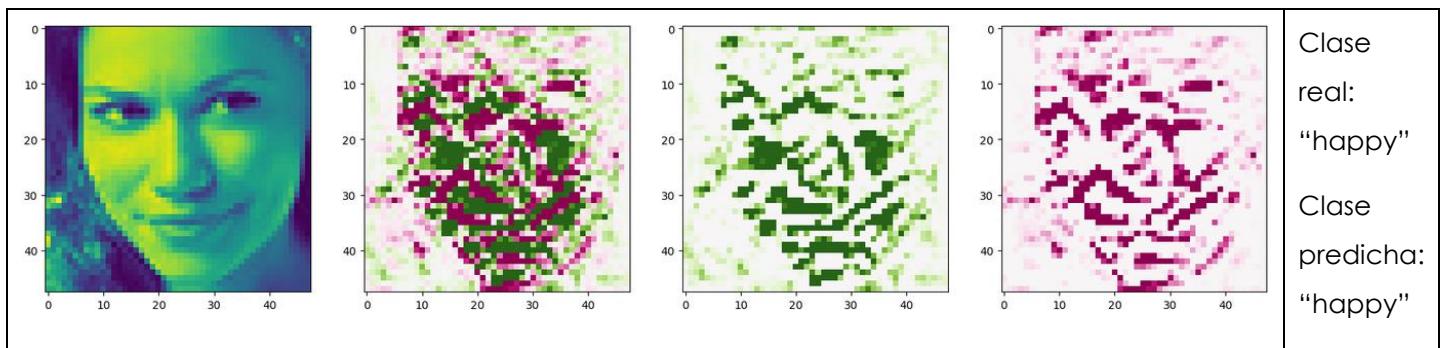
Tabla 23.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase "fear" para el problema de clasificación de siete emociones

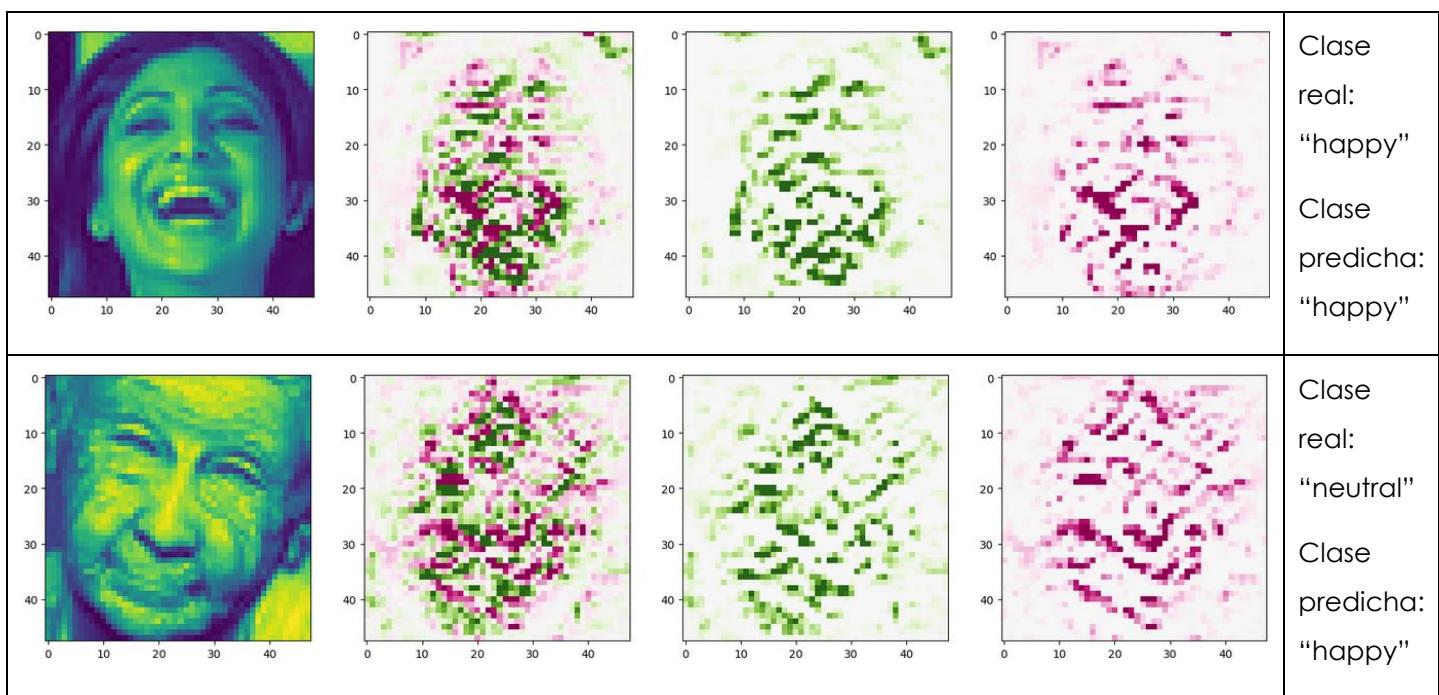




En lo que respecta a la emoción "fear" los píxeles que el modelo marca positivamente son los relativos a los ojos y las cejas, dejando a un lado las regiones de la boca y aledañas. Para verificar que el modelo no siempre acierta, se presenta la última imagen a modo de ejemplo. El modelo valora positivamente los píxeles correspondientes a una ceja y un lateral de la boca, y por ese motivo no ha predicho la clase correcta.

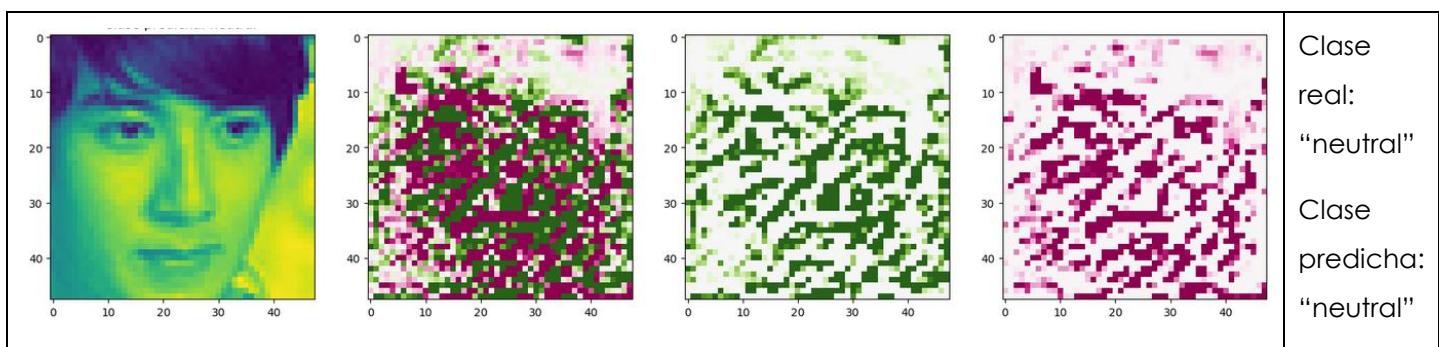
Tabla 24.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase "happy" para el problema de clasificación de siete emociones

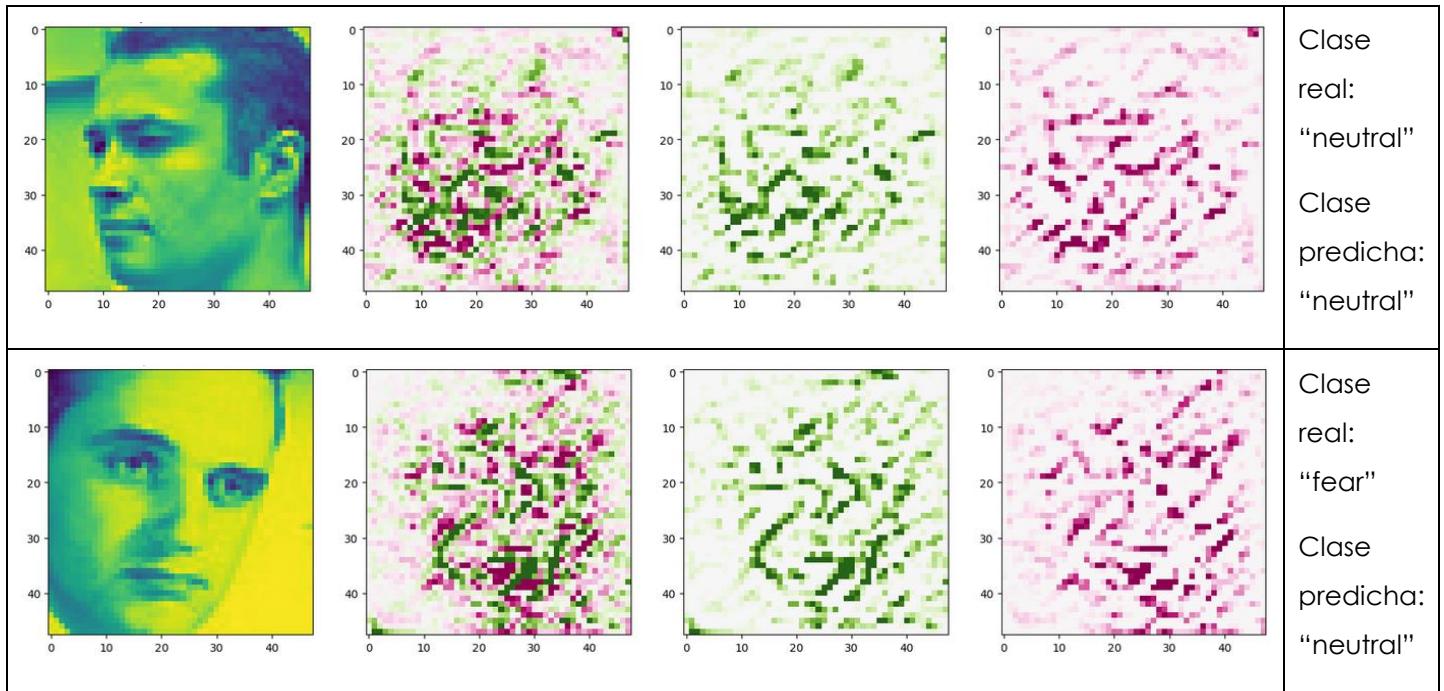




Cuando aparece una boca que encaja con los patrones del clasificador para la emoción "happy", el modelo valora positivamente los píxeles que modelan esa forma, como se puede observar en las dos primeras imágenes. En la imagen final, como no tiene una forma definida para la boca, se centra en valorar positivamente otros píxeles que encuentra relevantes, como las arrugas de la frente y la mitad de un ojo. Debido a que centra su atención en otras características, predice una clase parecida a la real: "happy"

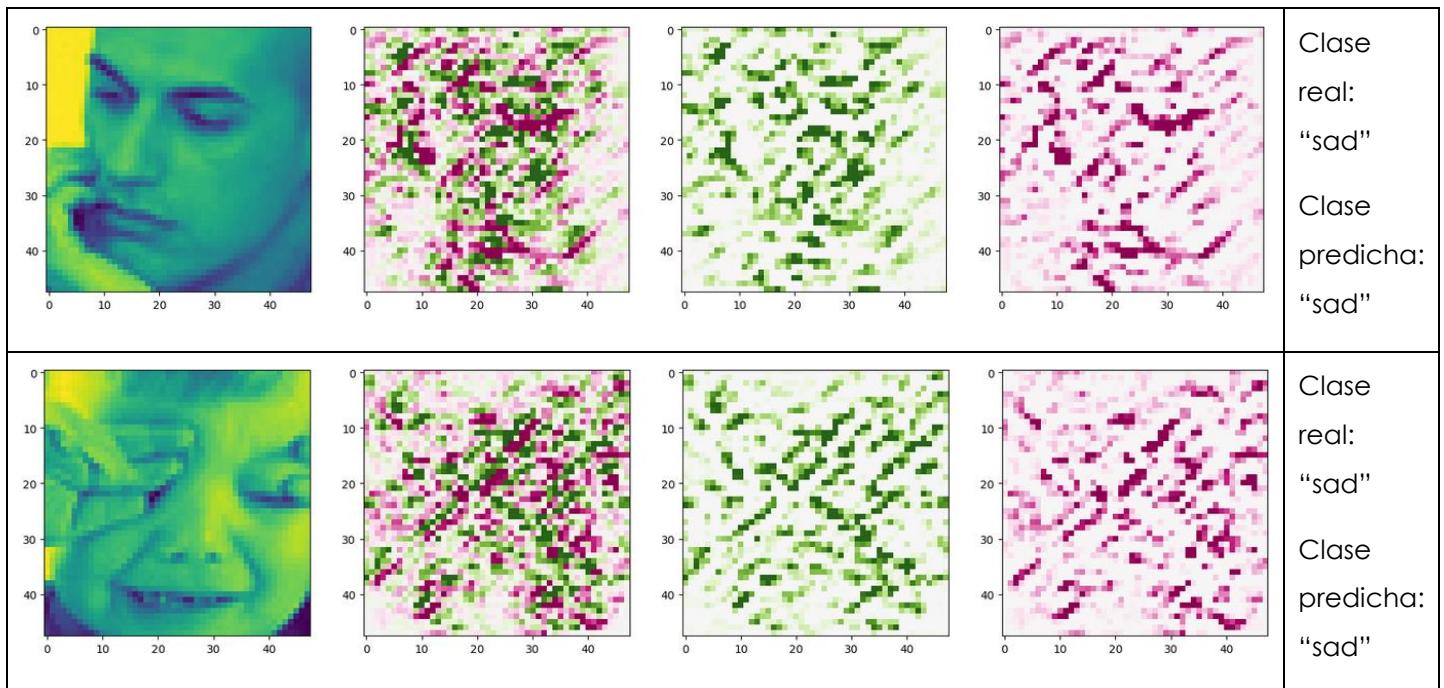
Tabla 25.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase "neutral" para el problema de clasificación de siete emociones

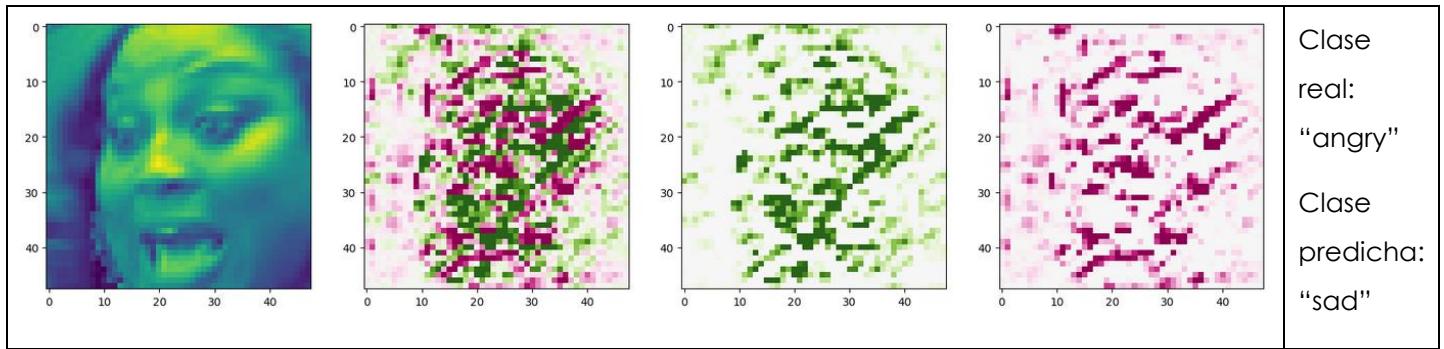




El modelo solamente tiende a valorar positivamente píxeles relativos a la boca y alrededores para la emoción "neutral". Se puede observar en la última imagen que se han valorado positivamente los píxeles de la arruga cercana a la boca y los que trazan el contorno de un ojo, por lo que no se predice la emoción correcta.

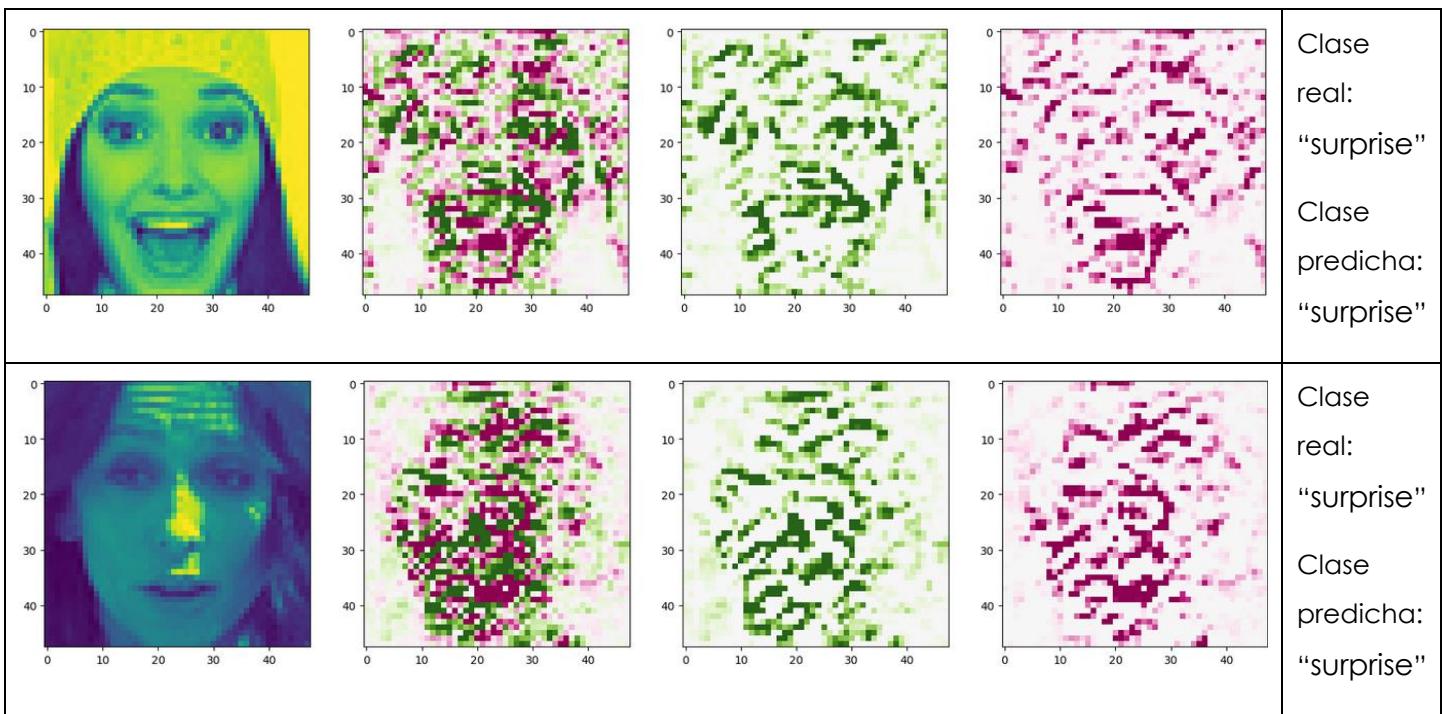
Tabla 26.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase "sad" para el problema de clasificación de siete emociones

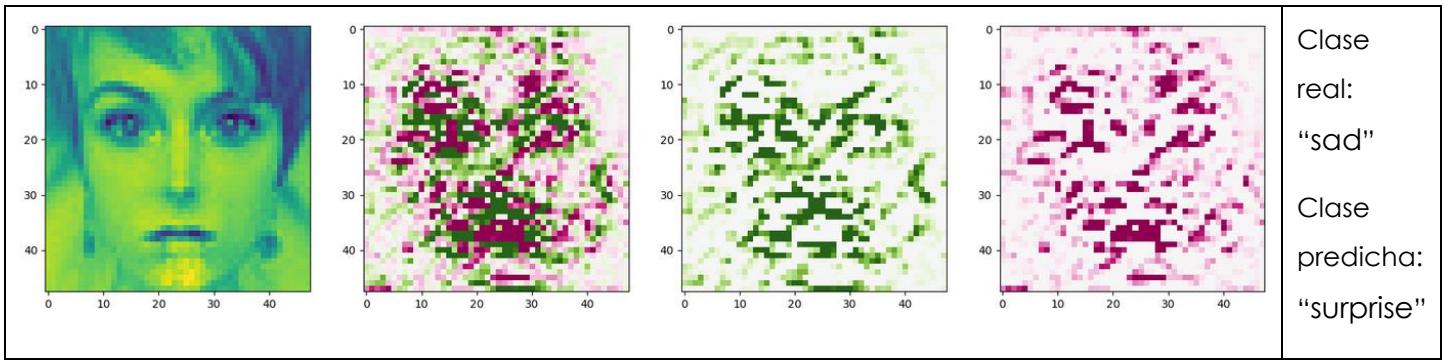




Para "angry" se puede observar que el modelo ha conseguido valorar positivamente los píxeles aledaños a los ojos y a las cejas de las personas de las imágenes. Por el contrario, valora negativamente los píxeles del contorno de la cara y a veces, aquellos cercanos a la boca. Como se puede apreciar en la última imagen, el modelo ha asignado valores positivos bajos a las cejas, haciendo que no tengan gran relevancia en la clasificación, y es por eso por lo que la emoción predicha es errónea.

Tabla 27.- Explicabilidad por el método de IntegratedGradients de imágenes correspondientes a la clase "surprise" para el problema de clasificación de siete emociones





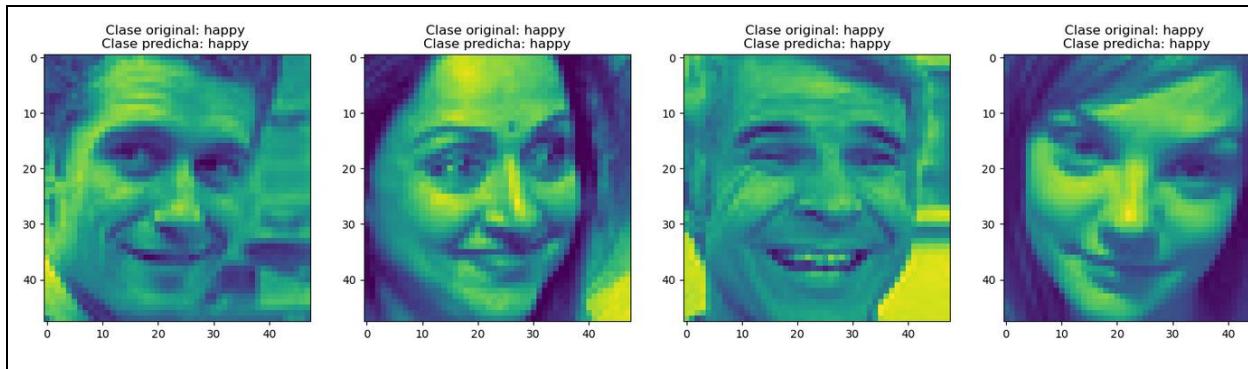
El modelo asigna valores positivos a los píxeles relativos a los dientes y al contorno de la boca cuando está abierta, pero valora negativamente los píxeles del interior de la boca, porque no ofrecen información de interés. Como se puede ver en la última imagen, se valoran a favor los píxeles que marcan el contorno del ojo derecho, y se valoran en contra los correspondientes a la boca.

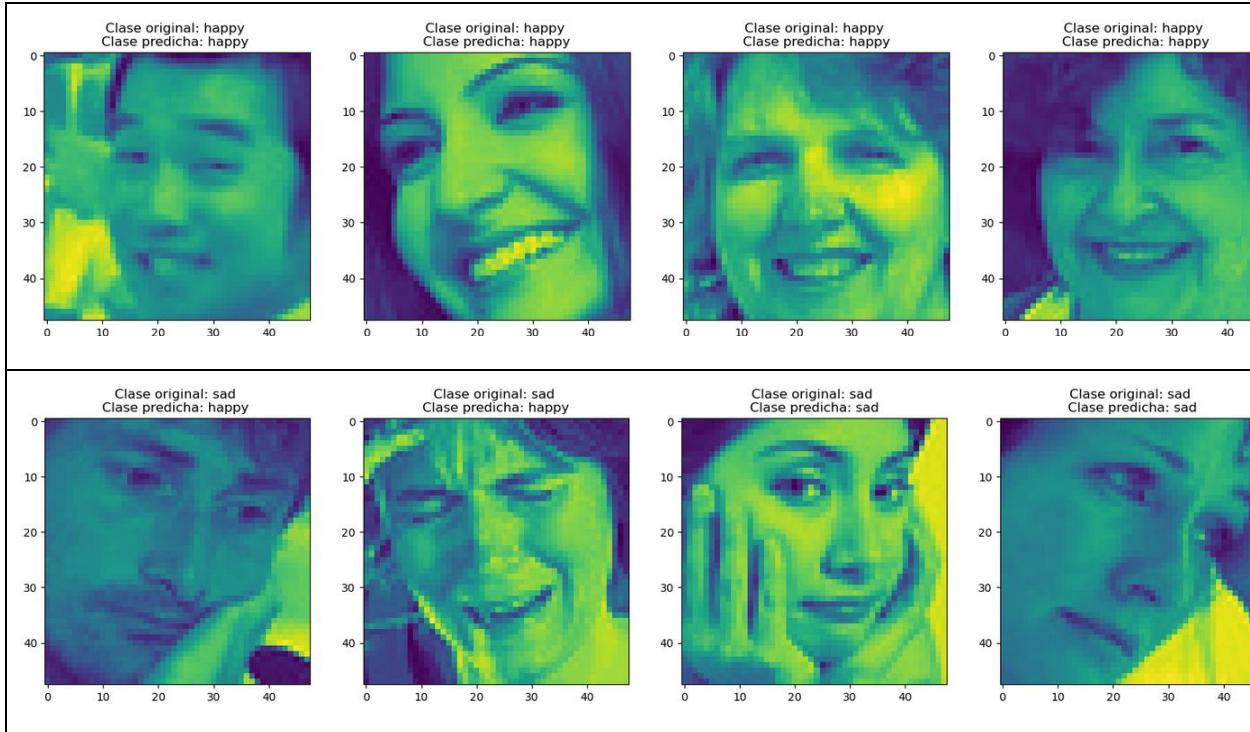
5.3 GradientSimilarity

Este método muestra, dada una imagen de entrada, tres imágenes con rasgos parecidos. Este método de explicabilidad es el que más se aproxima a la forma de clasificar humana, porque basa las explicaciones en la proximidad y comparación con referencias.

A continuación, se van a mostrar algunos ejemplos representativos del método GradientSimilarity para el problema de clasificación de tres tipos de emociones aplicado a imágenes del conjunto de datos final.

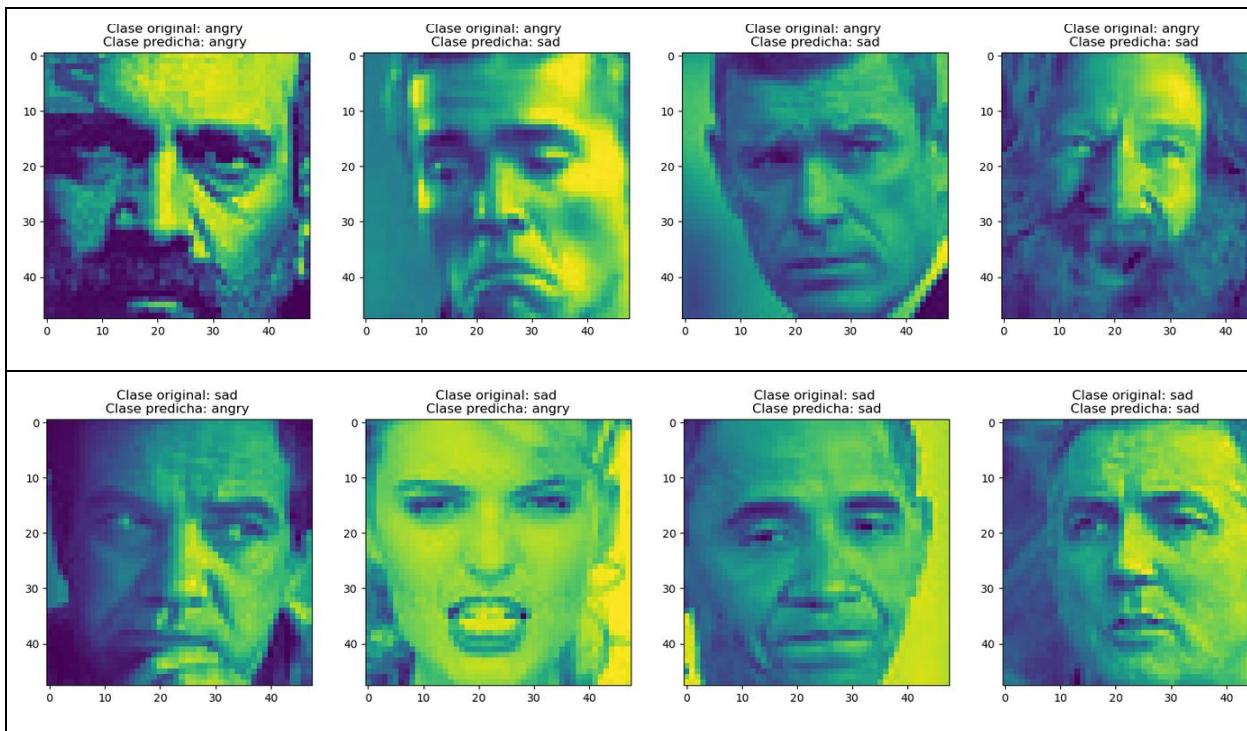
Tabla 28.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase "happy" para el problema de clasificación de tres emociones





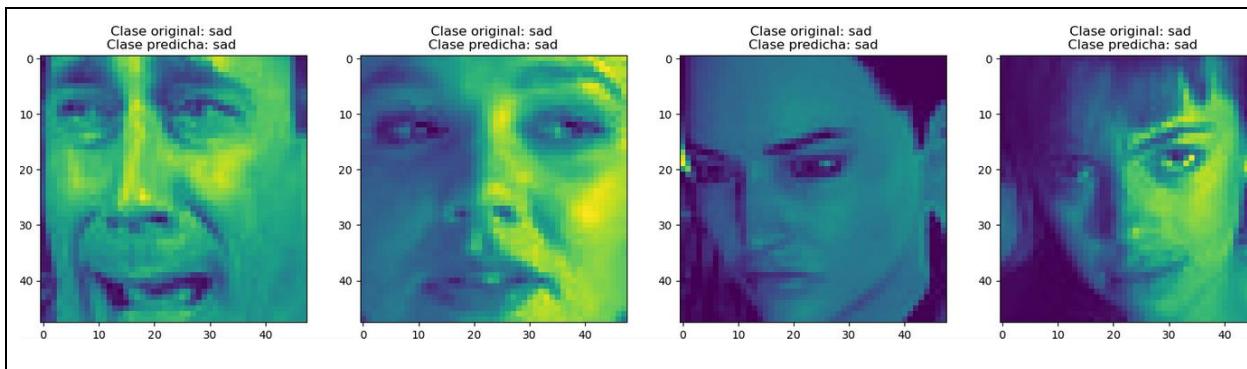
En las imágenes que se muestran anteriormente, se puede observar que el modelo, si recibe una imagen de una persona contenta, pero con boca cerrada, muestra por lo general imágenes de personas contentas, pero con bocas cerradas, por lo que se puede deducir que la boca es un factor importante que tiene en cuenta a la hora de detectar la emoción "happy". Como dato curioso, también se fija si en la cara hay presentes elementos extraños, para así buscar imágenes que también tienen elementos extraños parecidos. Un ejemplo de esto es la última imagen, que ha detectado que hay una mano, y muestra caras con manos. También, en relación con la última imagen, la emoción predicha ("happy") no se corresponde con la original ("sad"). Esto se puede deber a que hay otra imagen parecida que también ha clasificado de forma errónea, pudiendo hacernos pensar que el modelo arrastra errores.

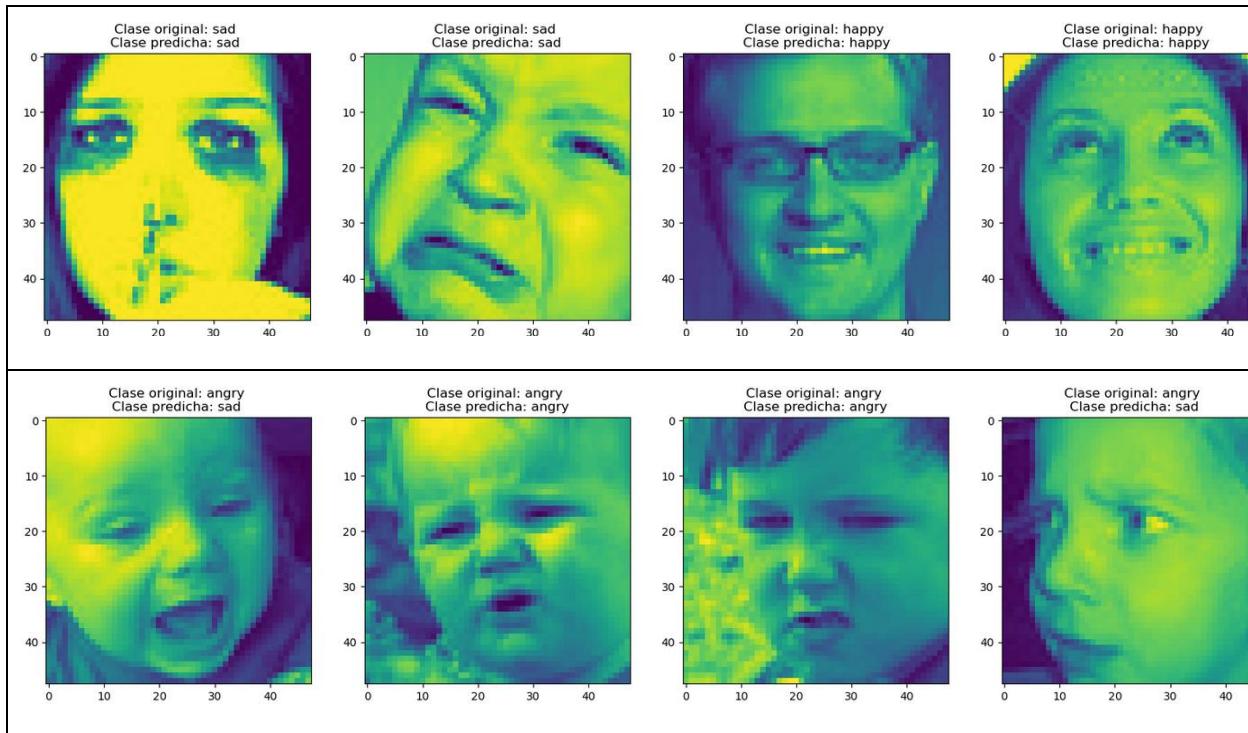
Tabla 29.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase "angry" para el problema de clasificación de tres emociones



Al igual que en "happy", se puede observar que el modelo, si recibe una imagen de una persona enfadada, pero con boca abierta, muestra imágenes de personas enfadadas, pero con boca abierta, lo que nos lleva a pensar que para discriminar la emoción "angry" se fija en la forma de la boca. También muestra formas de cejas parecidas.

Tabla 30.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase "sad" para el problema de clasificación de tres emociones

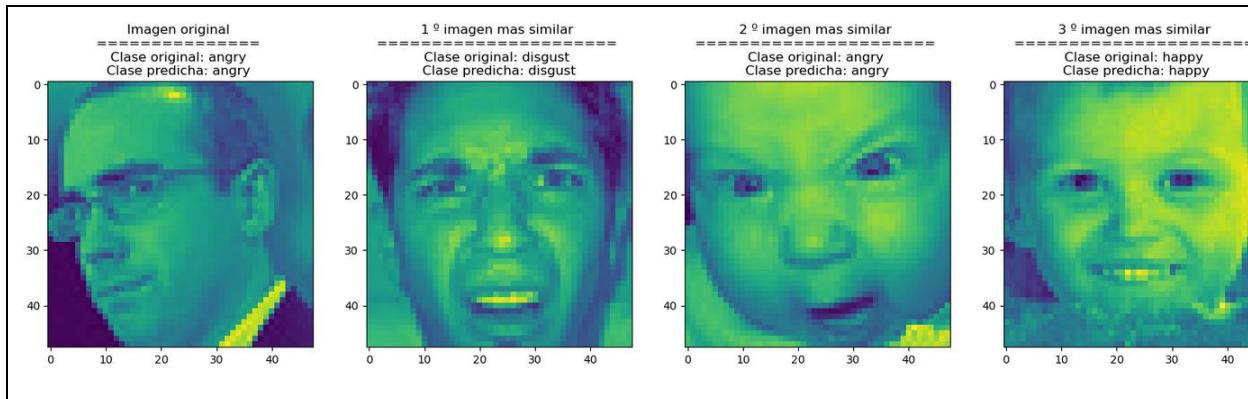


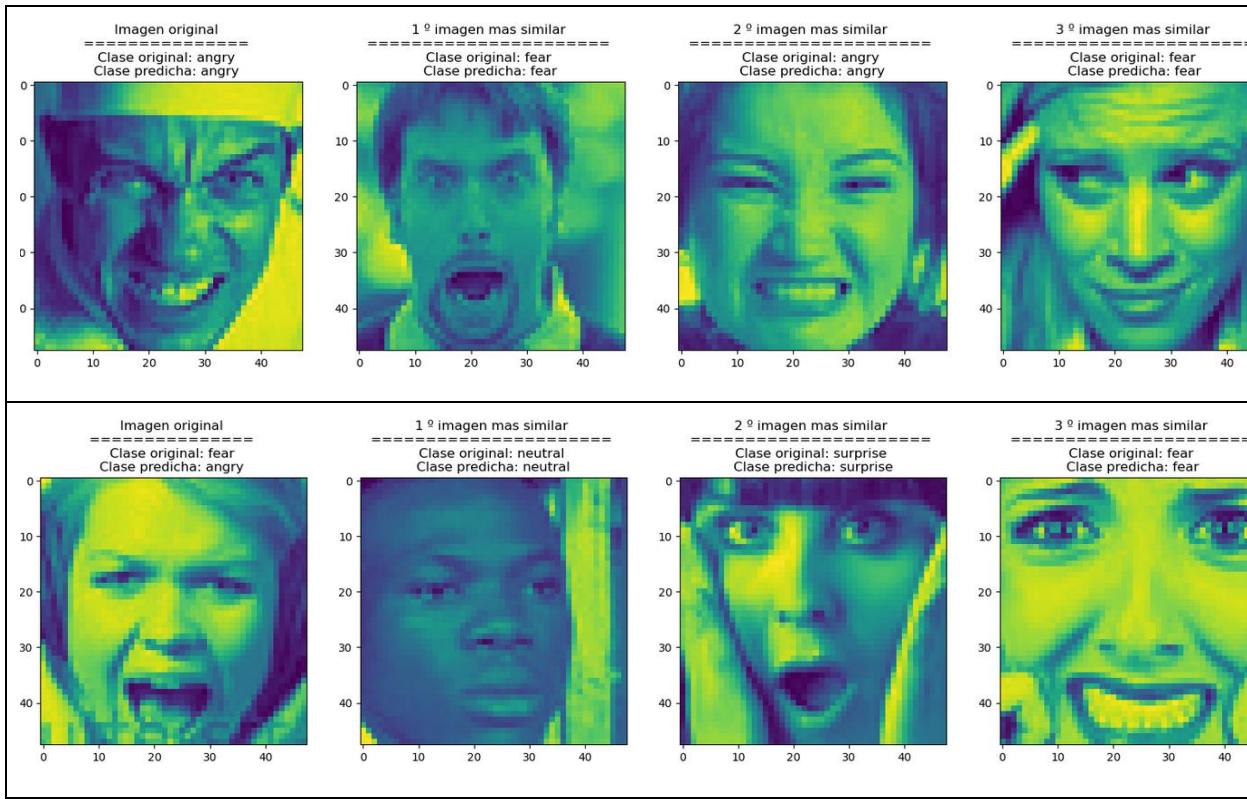


En esta emoción, no solamente se fija en la boca, sino también en las arrugas de la cara. Se puede observar como en la última imagen, al tener el bebé pocas arrugas, el explicador muestra imágenes de bebés. Siguiendo las conclusiones tomadas anteriormente, la razón por la que el modelo ha predicho "sad" y no "angry" en la última imagen se puede deber a la forma de los ojos.

A continuación, se van a mostrar algunos ejemplos representativos del método GradientSimilarity para el problema de clasificación de siete tipos de emociones aplicado a imágenes del conjunto de datos final.

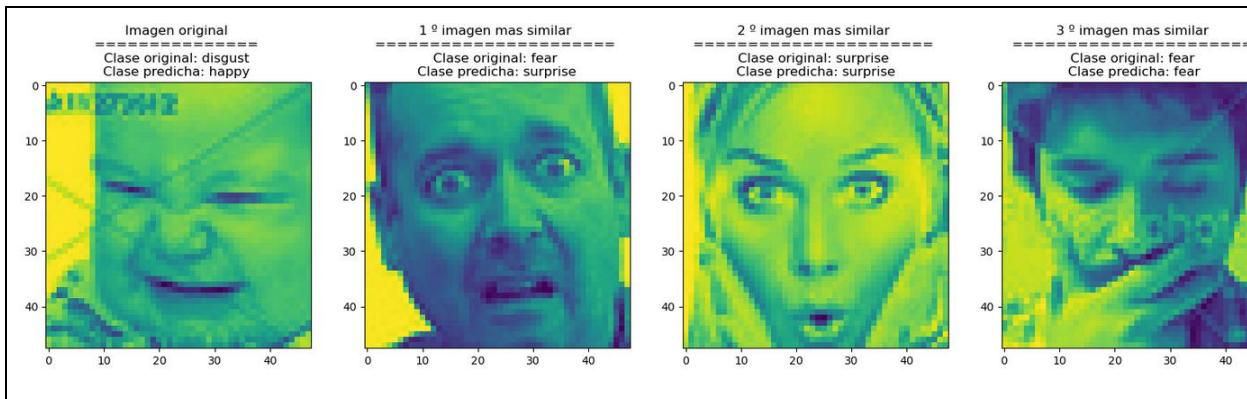
Tabla 31.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase "angry" para el problema de clasificación de siete emociones

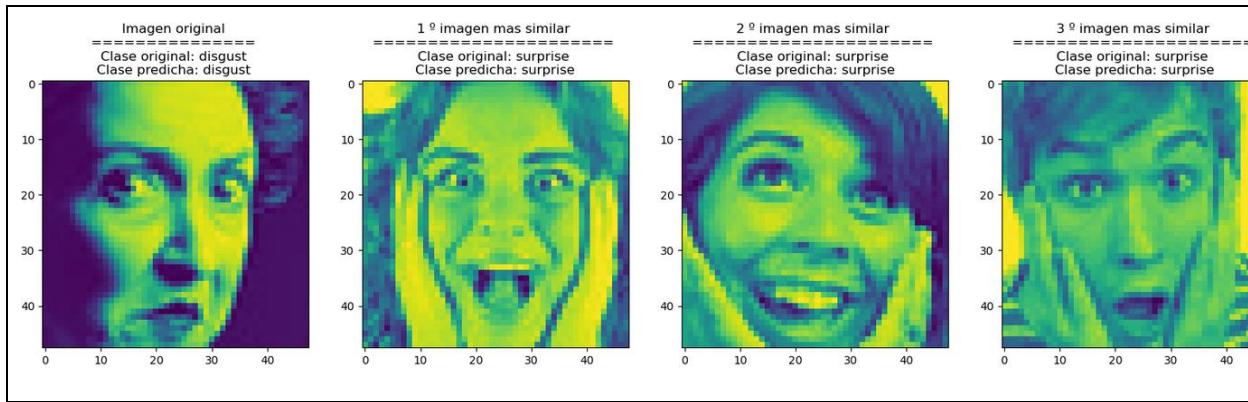




Se puede observar que para explicar imágenes de la emoción "angry" no necesariamente elige imágenes muy similares, sino que, en algunas como es el caso de la primera imagen (señor con gafas), muestra imágenes con ojos parecidos al original, pudiendo elegir imágenes de otras emociones totalmente diferentes a la que se ha predicho. En el caso de la primera imagen de la segunda fila, primero ha buscado una imagen con cejas muy similares, y después se ha centrado en buscar imágenes con la boca similar.

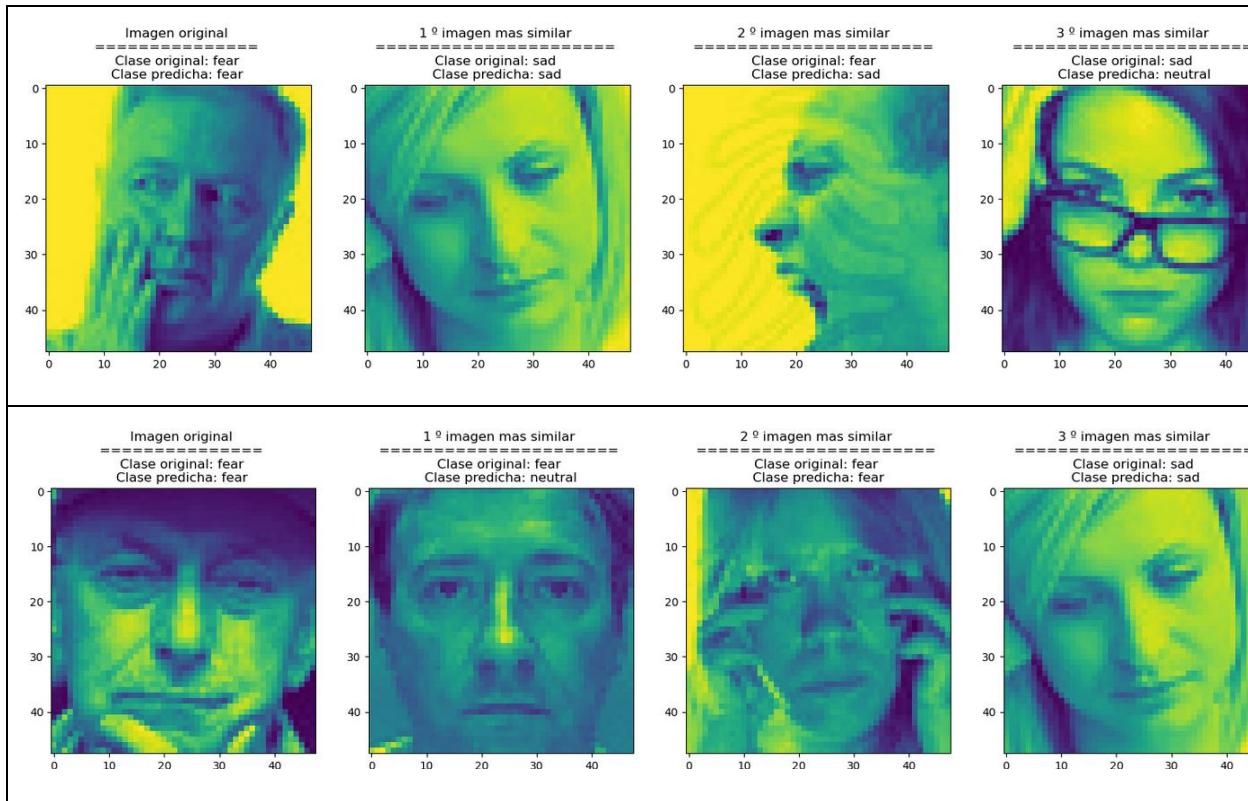
Tabla 32.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase "disgust" para el problema de clasificación de siete emociones

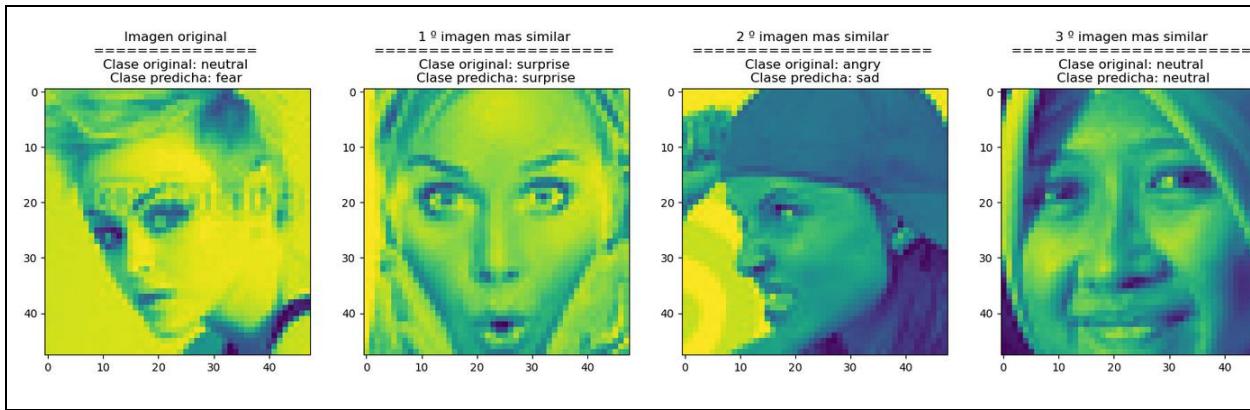




Para la emoción “disgust”, tal y como se puede ver en la primera imagen de la última fila, se buscan imágenes con ojos y cejas muy parecidos, aunque las demás imágenes buscadas no coincidan con la emoción de la imagen que se pretende explicar. Las características de las imágenes elegidas para explicar la primera imagen de la primera fila no tienen ninguna similitud con las del bebé.

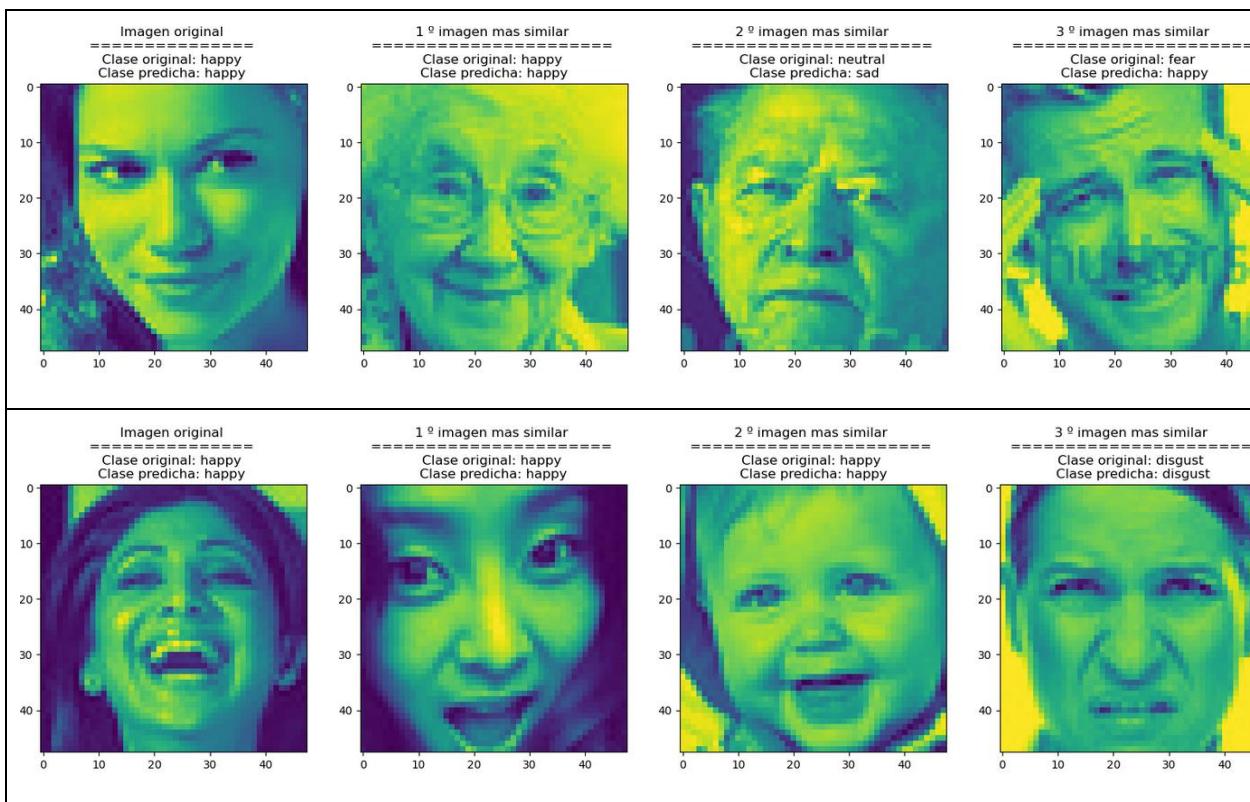
Tabla 33.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase “fear” para el problema de clasificación de siete emociones

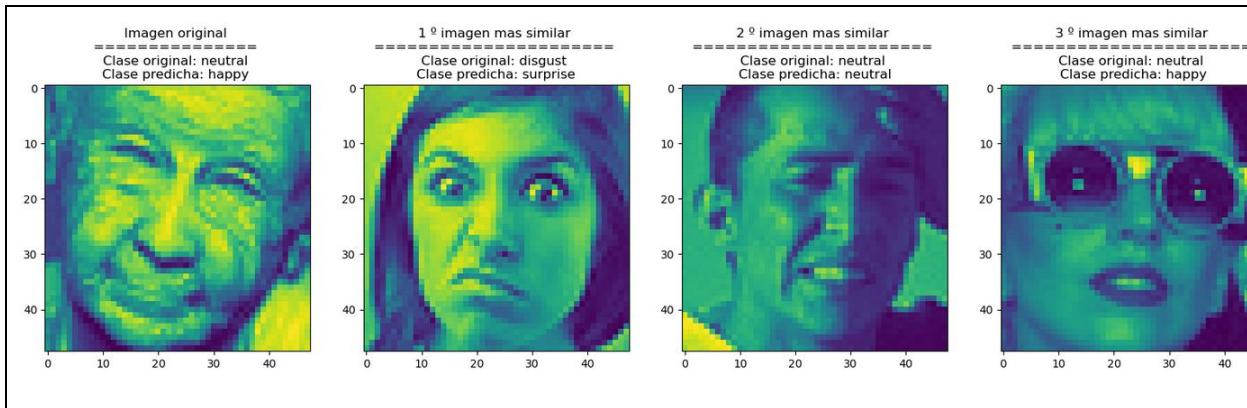




El clasificador muestra imágenes con ojos, boca y cejas parecidas para la emoción "fear". Este método se guía por similitud, y como se puede ver en la primera imagen de la última fila, busca imágenes con cejas parecidas a la original, sin importar si son las esperadas para la emoción que se pretende explicar.

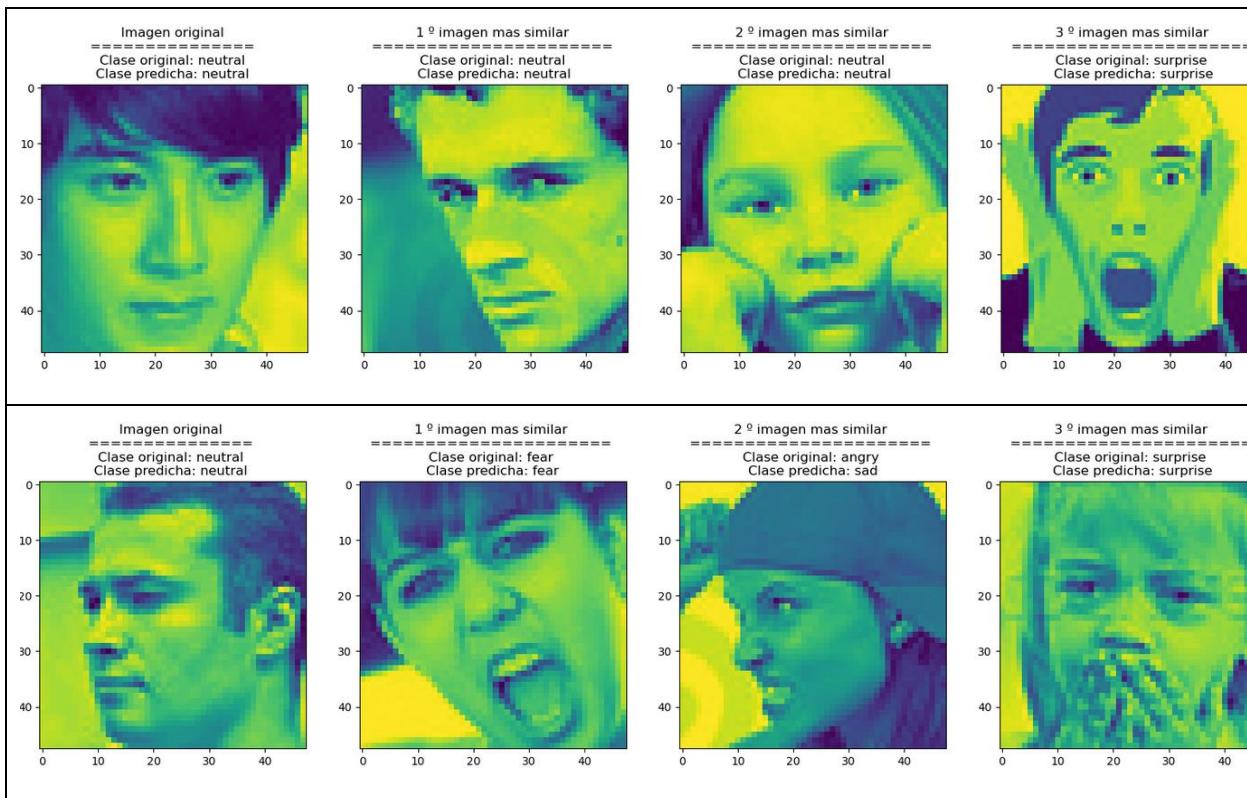
Tabla 34.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase "happy" para el problema de clasificación de siete emociones

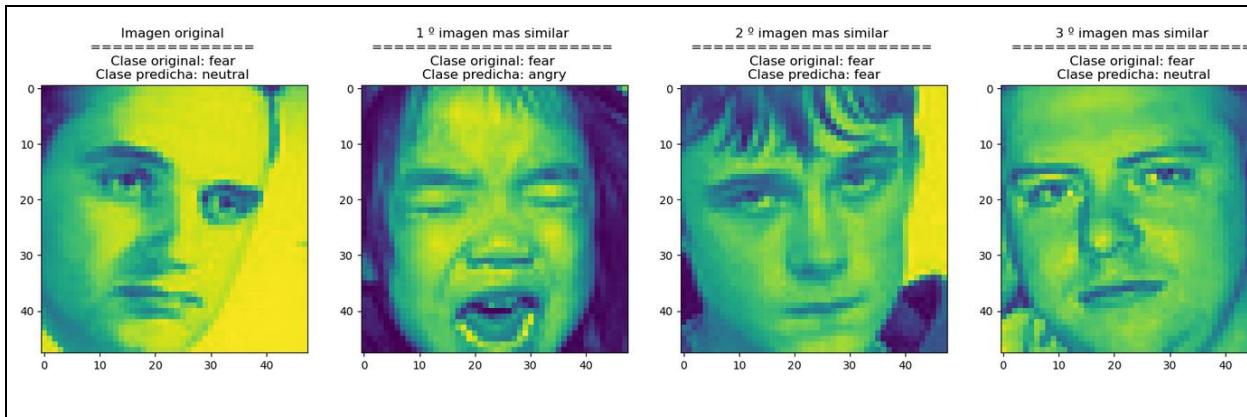




Se puede observar que el modelo se centra en la forma de la boca para imágenes clasificadas como "happy", por lo que el explicador muestra imágenes con bocas parecidas. Dado que no se ha podido fijar en la forma de la boca de la señora mayor de la última fila, además de no acertar la emoción a la que pertenece, ha mostrado imágenes con características muy distintas a la original.

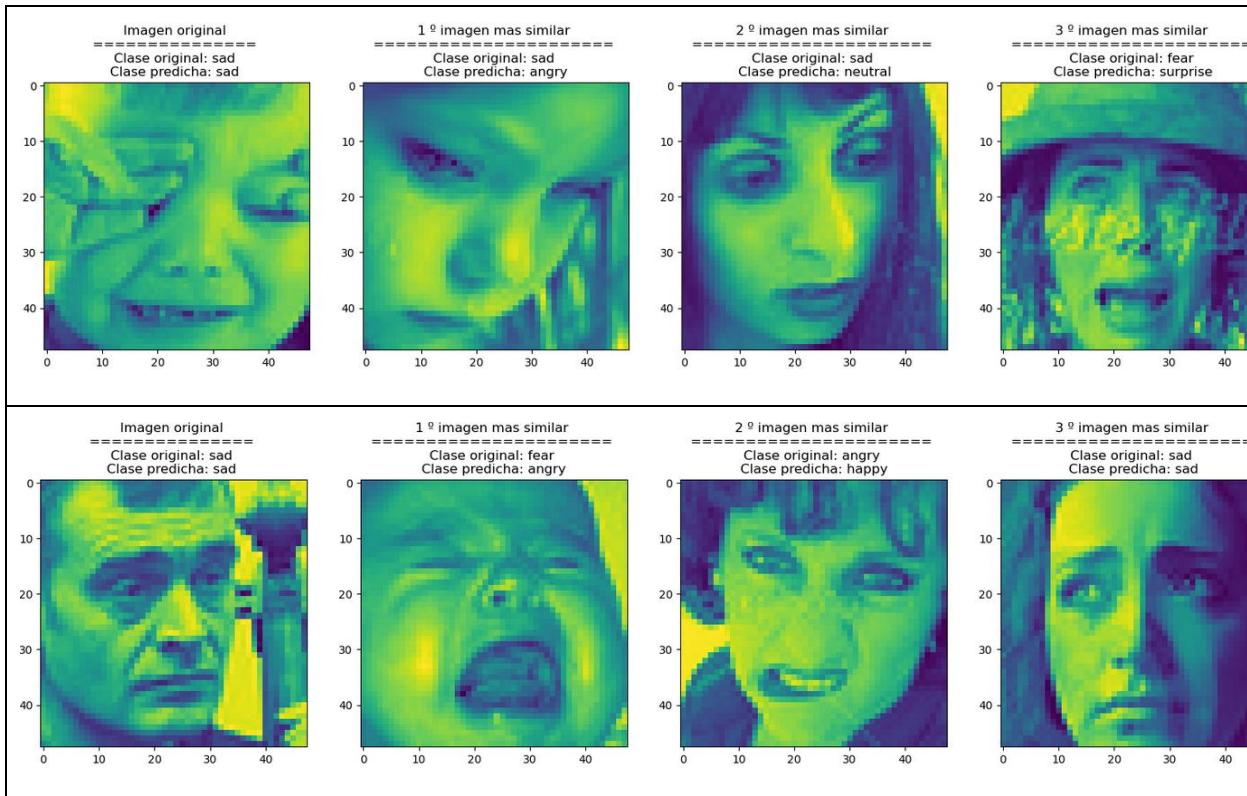
Tabla 35.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase "neutral" para el problema de clasificación de siete emociones

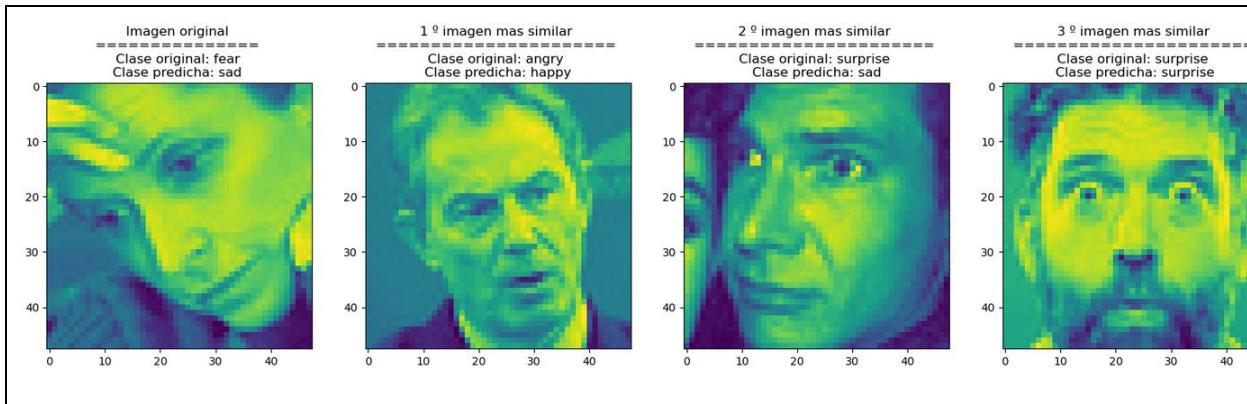




En el caso de la clase “neutral” el clasificador fija su atención mayoritariamente en las cejas. A veces, si no consigue extraer información relevante de la forma de las cejas, centra su atención en la boca, como es el caso de la primera imagen de la primera fila. Curiosamente, en la primera imagen de la última fila, que debería haber clasificado como “fear” pero ha clasificado erróneamente como “neutral”, muestra como imagen más similar una que resulta no tener relación alguna con la original.

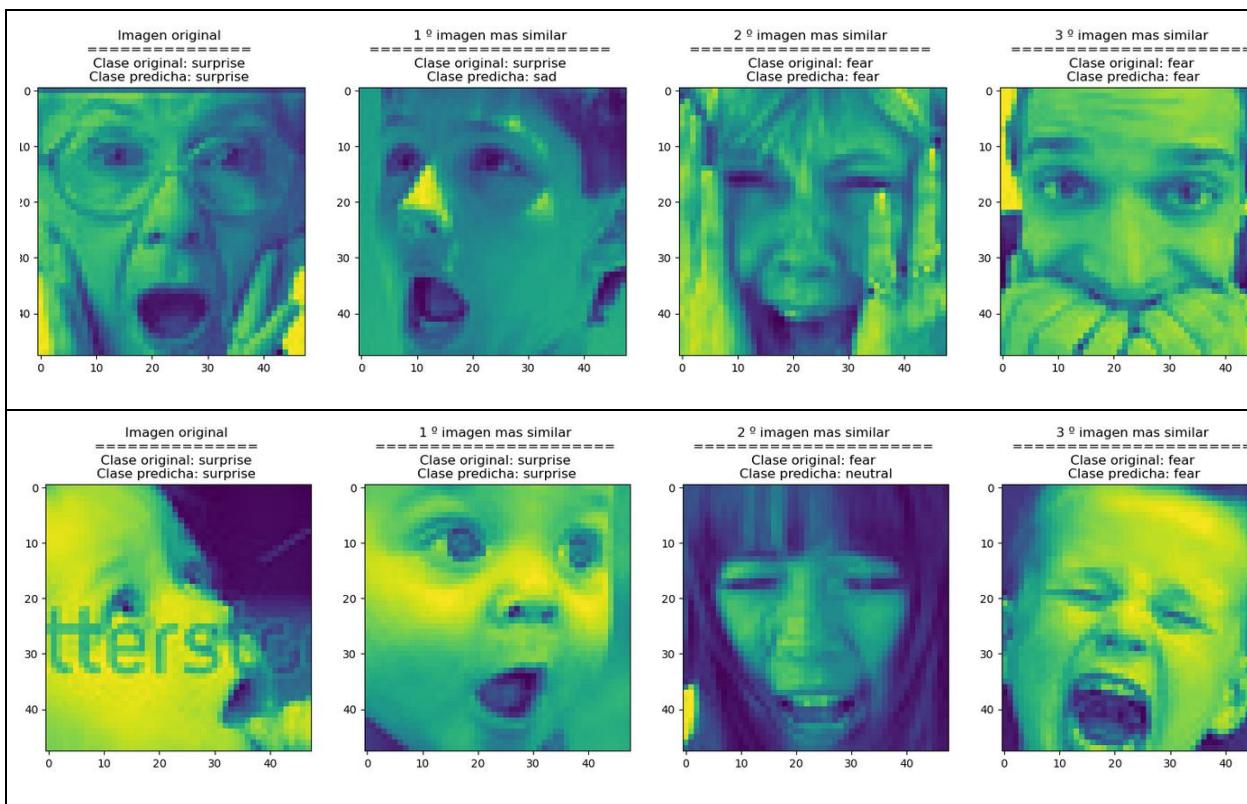
Tabla 36.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase “sad” para el problema de clasificación de siete emociones

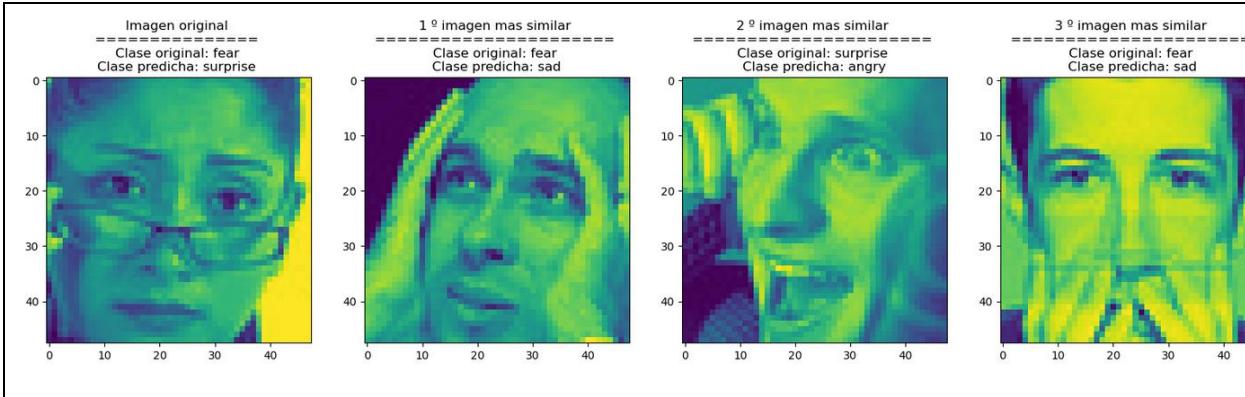




El modelo fija su atención mayoritariamente en la forma de la boca para la emoción "sad". También se fija algunas veces en las cejas. Por esto mismo, muestra imágenes con similitudes en estas características.

Tabla 37.- Explicabilidad por el método de GradientSimilarity de imágenes correspondientes a la clase "surprise" para el problema de clasificación de siete emociones





Claramente, el modelo muestra imágenes similares basándose en la forma de la boca para la emoción “surprise”. Aunque ese sea el factor predominante en la elección de las imágenes más similares, también se basa en la forma de las cejas. En relación con la primera imagen de la última fila, de clase “fear” pero clasificada como “surprise”, la mala predicción se puede deber a la presencia de elementos extraños en la cara, como gafas o pendientes que logran despistar al clasificador.

Capítulo 6 - Conclusiones y trabajo futuro

6.1 Conclusiones

En este apartado se comentarán las conclusiones de las diferentes aproximaciones que se han llevado a cabo.

Conclusiones de la primera aproximación a la clasificación de emociones en imágenes:

La alta resolución de las imágenes es un problema a la hora de entrenar los modelos, porque tardan más en entrenar y no consiguen mejores precisiones. Los modelos, se “ pierden ” al entrenar y no consiguen extraer características de muchas de las imágenes, en gran medida porque hay muy pocas imágenes disponibles, y porque las imágenes son muy distintas: hay tanto personas individuales como grupos de personas, y también cuerpos completos y caras.

Con respecto a los modelos MLP con los que se ha trabajado (en total 96), el 48% obtienen una precisión entre [0.0-0.09], el 16% entre [0.1-0.19], el 17% entre [0.2-0.29], el 17% entre [0.3-0.39] y el 2% entre [0.4-0.49]. La gran mayoría obtiene una precisión menor que 0.33 (clasificación aleatoria)

Con respecto a los modelos CNN con los que se ha trabajado (en total 32), el 59% obtienen una precisión entre [0.28-0.39], el 31% entre [0.40-0.49], y el 10% entre [0.50-0.54]. Estos modelos son mucho mejores que los MLP anteriores, pero siguen sin obtener resultados aceptables.

Aunque los modelos obtienen precisiones bajas para los datos de prueba, la clase que mejor clasifican es la mayoritaria: “ happy ”, y la que peor: “ sad ”. Las clases que menos confunden son “ sad ” con “ happy ”, por lo que ha extraído características que permiten discriminar entre clases, aunque no sean suficientes.

Conclusión: El conjunto de datos usado no es apto para el problema que se quiere resolver.

Conclusiones de la aproximación final a la clasificación de emociones en imágenes:

Clasificación y explicabilidad de tres tipos de emociones:

Debido a que los modelos MLP no están orientados al tratamiento de imágenes, muchos de los modelos obtienen precisiones sumamente bajas, menores de 0.33, y otros en cambio, sobreaprenden de los datos de entrenamiento. El mejor modelo no es el que más tarda ni el que mayores tamaños para las imágenes de entrada obtiene, por lo que a veces es necesario despreciar cierto porcentaje de precisión para lograr que el modelo sea más rápido. La clase que mejor clasifica es: "happy" y la que peor: "angry". Además, extrae alguna característica importante que permite diferenciar la clase "happy" de "angry", como la presencia de dientes.

Los modelos convolucionales consiguen resultados bastante aceptables en tiempos bastante pequeños, pero la mayoría no obtienen una precisión mayor que 0.6, pudiéndose deber a que la combinación de características usadas para crear el modelo no sean las idóneas, o a que los datos no están balanceados. El mejor modelo se comporta de forma muy parecida al MLP, pero confunde menos las clases, obteniendo una mayor precisión – concretamente: 0.73 frente a 0.51 del MLP.

En general, al ser un problema de clasificación de pocas clases, se puede observar mediante la explicabilidad que las características obtenidas están muy marcadas, existiendo suficientes variaciones entre éstas para poder discriminar con bastante acierto las clases.

Clasificación y explicabilidad de siete tipos de emociones:

Los modelos convolucionales consiguen resultados bastante aceptables en tiempos relativamente buenos para la gran cantidad de imágenes con las que trabaja. La mayoría de los modelos consiguen una buena precisión, es decir, mayor de 0.5, a pesar de que hay una clase con muy pocas instancias. Esto puede deberse a que las características usadas potencian la detección de patrones. En comparación con el mejor modelo CNN anterior, la precisión no es tan baja, ya que se han añadido más del doble de clases nuevas. Aunque la tasa de acierto sea mayor que la de fallo, la clase que peor clasifica es: "disgust" y la que mejor: "happy". Por otro lado, la clase que más

confunde es “sad” con “fear” y las que menos “disgust” con “happy”, “neutral” y “surprise”. Como hay pocas instancias de la clase “disgust”, el modelo no puede extraer características para discriminar esta clase de las demás. La precisión del mejor modelo CNN es 0.56.

Mientras hay clases muy reconocibles como: “happy” y “neutral”, hay otras muy parecidas entre sí: “angry”, “fear” y “surprise”, y “disgust” y “sad”. Se puede observar en la explicabilidad que, aunque tengan características comunes, hay pequeñas variaciones que hace que confunda “disgust” y “sad” poco entre sí. En cambio, al no haber encontrado una característica única para la clase “angry”, se confunde a menudo porque se basa en características que son también comunes a “fear” y “sad”. Ocurre lo mismo con la clase “fear”, que la confunde mucho con “angry” y “surprise”.

En relación con los métodos XAI usados, son muy útiles para mostrar las regiones importantes detectadas por el modelo al que se aplican. Las explicaciones son, en cierta medida, fácilmente interpretables, por lo que se pueden extraer conclusiones valiosas y de gran calidad para el estudio.

Conclusión: El conjunto de datos usado es apto para el problema que se quiere resolver, porque hay modelos que consiguen extraer muchas características de muy buena calidad. En general, la explicabilidad muestra regiones acordes con lo que se pretende que se fije el modelo, como bocas, ojos, cejas, arrugas, etc. Es por esto por lo que los mejores modelos obtenidos son muy buenos.

6.2 Trabajo futuro

En este apartado se detallan algunas propuestas de trabajo futuro que ayudarían a completar el trabajo realizado:

En primer lugar, un aspecto importante a realizar si se continúa con el trabajo es expandir el conjunto de datos final, completándolo hasta lograr un conjunto de datos equilibrado. Una vez realizadas pruebas sobre un conjunto de datos completo, se puede expandir también el conjunto de datos inicial y equilibrarlo para hacer pruebas con él. También se puede expandir el número de emociones, eligiendo emociones mucho más complejas.

Por otra parte, debido a que no ha sido posible ejecutar los modelos MLP con imágenes de gran resolución, en concreto con los tamaños: (1152, 809) y (768, 539), se propone usar dispositivos más potentes en la nube como Google Cloud para ejecutar el proceso de entrenamiento de estos modelos y ver si la tendencia actual de obtener mejores precisiones según disminuye el tamaño de la imagen, se mantiene. También se pueden usar otros frameworks de Inteligencia artificial para construir los modelos, como PyTorch.

Otro aspecto interesante que no se ha considerado en este trabajo, en gran medida para evitar tiempos de entrenamiento excesivamente elevados, es realizar el estudio sobre imágenes a color y después realizar una comparación de las nuevas características con las previamente obtenidas para verificar si coinciden y, por lo tanto, la información del color es un factor importante.

Introduction

Motivation

From a computational point of view, an image is a matrix of numbers which represent colors or gray tones. Thanks to the representation that pixels offer, in some cases it is possible to extract existing meaningful connections from a group of pixels that create relations between a group of images, as they share patterns. These patterns can be detected by supervised learning models from the machine learning field (ML) of the Artificial Intelligence. There are several types of models that, after searching for patterns on the input images set and the input labels set during the training process, can classify a set of images among the different established categories. Certainly, in most cases, these detected patterns and features are invisible to those who use the models to make predictions of a set of images over several classes, as almost all the most used models only provide, given an image as an input, the attributed class, without showing the important zones of the image by which the model has decided to relate it with one a group of images or another.

Goals

Along this thesis it will be carried out a study of the different existing classification models applied to images, with the final objective of determining which model and parameters are the best ones to solve a problem of in-face emotions classification. Without leaving behind this objective, it will also be carried out a study of the different existing explainability algorithms applied to images, with the final objective of being able to show in the original image the areas selected by the best black box model obtained, to verify the correct detection of the different emotions present on the images and therefore, the correct classification of the images in the existing categories.

Work plan

Along this thesis it will be shown the development of the work. The stages of the work performed are detailed bellow:

Investigation

During this phase it has been carried out a research over a wide range of topics related with the development of the work, topics related to machine learning of Artificial Intelligence, specifically, supervised learning and convolutional neural networks. Not only has been made a study about the Explainable Artificial Intelligence applied to images, but also about the practical applications of the use of in-face emotion detection models.

Development and experimentation

In this subsection it will be talked about the datasets which have been used and the models and features that have been experimented with. First, to make an initial approach to the work, it has been used a small dataset of images which contain faces that represent three types of emotions. After this, to start with the final work it is used a dataset called FER2013, which contains 48x48 grayscale images of seven types of emotions. This dataset is bigger than the initial one. The first tests have been made over a subset of the final dataset, more specifically, over three types of emotions. Finally, there were held some tests over the entirely final dataset. After testing and setting the best model, some explainability algorithms from ALIBI library are used to show the important regions of the input image. All the work done is available on the next link: <https://github.com/javieg25/Explicabilidad-aplicada-a-modelos-de-reconocimiento-de-emociones-en-imagenes.git>

Documentation

As long as the knowledge was expanded during the research stage, and the development and experimentation with the models and explainability algorithms was carried out, this document has been filled in, including the data used and the conclusions reached after studying the results obtained.

Diagrama de Gantt

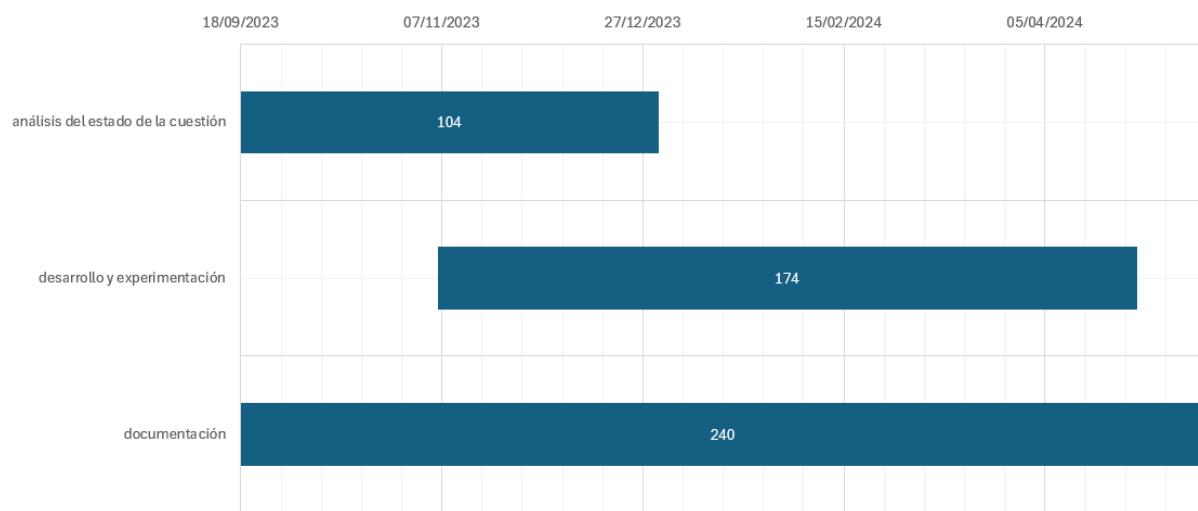


Ilustración 25.- Gantt diagram of the work plan

Conclusions and future work

Conclusions

This section will talk about the conclusions of the different approaches that have been carried out.

Conclusions over the first approach to the classification of emotions in images:

The high resolution of the images is a problem when training the models, because they spend a lot of time on training without obtaining good accuracies. We could say that the models get lost when training, as they are not able to extract features of most of the images, because there are very few images available, and because the images are very different: there are both individuals and groups of people, as well as bodies and faces.

In relation to the MLP models used (96 in total), 48% of them obtained an accuracy between [0.0-0.09], 16% of them between [0.1-0.19], 17% of them between [0.2-0.29], 17% of them between [0.3-0.39] and 2% of them between [0.4-0.49]. Most of them obtained an accuracy lower than 0.33 (which corresponds to a random classification).

In relation to the CNN models used (32 in total), 59% of them obtain an accuracy between [0.28-0.39], 31% of them between [0.40-0.49], and 10% of them between [0.50-0.54]. These models are much better than the MLP used before, but they do not obtain good results.

Although the models obtain low accuracies from the test data, the class that classifies better is: "happy", and the worst: "sad". The classes that confuse the least are "sad" and "happy", so we can say that the model has extracted features that allow differentiating between classes, although they are not enough.

Conclusion: The dataset is not suitable for the problem that it is trying to be solved.

Conclusions over the final approach to the classification of emotions in images:

Classification and explainability of three types of emotions:

Because MLP models are not the best ones to process images, many of the models obtain extremely low accuracies (less than 0.33), and others overfit from the training data. The best model is not the one that takes more time to train, nor the one that works over big sizes for images, so we can say that sometimes it is necessary to disregard some percentage of accuracy to make the model faster. The class that the model classifies better is: "happy" and the worst: "angry". It also extracts some important features, such as the presence of teeth, that allows to differentiate the classes "happy" and "angry".

The convolutional models achieve very good results in low times, but most of them do not obtain an accuracy higher than 0.6, which may be due to the combination of features used to create the model not being ideal, or because the data is not balanced. The best model behaves like the best MLP but obtains higher accuracies – specifically: 0.73 vs 0.51 of the MLP.

In general, as it is a problem of classification of very few classes, the explainability shows that the features obtained are very representative of each class, with enough variations among them to be able of discriminating quite well the classes.

Classification and explainability of seven types of emotions:

The convolutional models achieve very good results in relatively low times for the big number of images used. Despite the fact there is a class with very few number of instances, most of the models achieve a good accuracy (more than 0.5) This may be because the features detected do really enhance pattern detection. In comparison to the previous best CNN model, the accuracy has not decreased a lot, since four more classes have been added. Although the hit rate is higher than the failure rate, the class that classifies better is: "happy" and the worst is: "disgust". On the other hand, the class that confuses more is: "sad" with "fear", and the least is: "disgust" with "happy", "neutral" and "surprise". As there are few instances of "disgust", the model cannot extract features to discriminate this class from the others. The accuracy of the best CNN model is 0.56.

While there are very recognizable classes such as: "happy" and "neutral", there are others very similar to each other: "angry", "fear" and "surprise", and "disgust" and "sad".

The explainability shows that, although they have features in common, there are small variations that make the model confuse "disgust" with "sad" very few times. Nevertheless, as the model has not found a unique characteristic for "angry", this class is often confused because it bases on characteristics that are also common to "fear" and "sad". It is the same with "fear", which is often confused with "angry" and "surprise".

In relation to the XAI methods used, they are very useful to show the important regions detected by the model to which they are applied to. The explanations are easy to understand, making it easy to obtain important conclusions for the study.

Conclusion: The data set used is suitable for the problem to be solved, because there are models that manage to extract many features of very good quality. In general, explainability shows regions that make sense with what is supposed the model to find, such as mouths, eyes, eyebrows, or wrinkles. This is the main reason why the best models obtained are very good models.

Future Work

This section details some proposals for future work that would help to complete the work done:

On the one hand, an important point to develop is to expand the final data set used until the data set is completely balanced. Once tests with the completed dataset are done, the initial dataset can also be filled until balanced to test it. The number of emotions can be expanded too, choosing much more complex emotions.

On the other hand, due to the fact that it has not been possible to execute the MLP models over big resolution images, such as the ones with sizes: (1152, 809) y (768, 539), it is proposed to use more powerful devices in the cloud such as Google Cloud to execute the training process of these models and see if the current trend of obtaining better accuracies as the image size decreases, is maintained. It can also be used other Artificial Intelligence frameworks to build the models, such as PyTorch.

Another interesting issue that has not been considered, mainly to avoid extremely high training times, is to make the study over images in RGB and then make a comparison

between the new features obtained and the old ones to verify if they match and therefore, determine that the information of the colour is an important issue to consider.

Bibliografía

- [1] IBM, «¿Qué es el aprendizaje supervisado?», 2024. [En línea]. Available: <https://www.ibm.com/es-es/topics/supervised-learning>.
- [2] Y. LeCun, L. Bottou, Y. Bengio y P. Haffner, «Gradient-Based Learning Applied to Document Recognition», 1998. [En línea]. Available: <http://yann.lecun.com/exdb/publis/pdf/lecun-01a.pdf>.
- [3] A. Krizhevsky, A. Krizhevsky y G. E. Hinton, «ImageNet Classification with Deep Convolutional», 2012. [En línea]. Available: https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.
- [4] K. He, X. Zhang, S. Ren y J. Sun, «Deep Residual Learning for Image Recognition», 2015. [En línea]. Available: <https://arxiv.org/pdf/1512.03385>.
- [5] K. Simonyan y A. Zisserman, «Very Deep Convolutional Networks for Large-Scale Image Recognition», 2014. [En línea]. Available: <https://arxiv.org/abs/1409.1556v6>.
- [6] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke y A. Rabinovich, «Going Deeper with Convolutions», 2014. [En línea]. Available: <https://arxiv.org/abs/1409.4842>.
- [7] M. D. Zeiler y R. Fergus, «Visualizing and Understanding Convolutional Networks», 2013. [En línea]. Available: <https://arxiv.org/abs/1311.2901v3>.
- [8] M. T. C. Ribeiro, S. Singh y C. Guestrin, «"Why Should {} Trust You?": Explaining the Predictions of Any Classifier», 2016. [En línea]. Available: <https://github.com/marcotcr/lime>.
- [9] Lundberg, S. M y S.-I. Lee, «A Unified Approach to Interpreting Model Predictions», 2017. [En línea]. Available: <https://shap.readthedocs.io/en/latest/index.html>.

- [10] M. T. C. Ribeiro, S. Singh y C. Guestrin, «Anchors: High-Precision Model-Agnostic Explanations,» 2018. [En línea]. Available: <https://github.com/marcotcr/anchor>.
- [11] A. Kapishnikov, T. Bolukbasi, F. Viégas y M. Terry, «XRAI: Better Attributions Through Regions,» 2019. [En línea]. Available: <https://arxiv.org/abs/1906.02825>.
- [12] M. Sundararajan, A. Taly y Q. Yan, «Axiomatic Attribution for Deep Networks,» 2017. [En línea]. Available: <https://arxiv.org/abs/1703.01365>.
- [13] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh y D. Batra, «Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization,» 2019. [En línea]. Available: <https://arxiv.org/abs/1610.02391>.
- [14] V. Petsiuk, A. Das y K. Saenko, «RISE: Randomized Input Sampling for Explanation of Black-box Models,» 2018. [En línea]. Available: <http://bmvc2018.org/contents/papers/1064.pdf>.
- [15] S. Muzellec, T. Fel, V. Boutin, L. andéol, R. VanRullen y T. Serre, «Saliency strikes back: How filtering out high frequencies improves white-box explanations,» 2024. [En línea]. Available: <https://arxiv.org/abs/2307.09591>.
- [16] J. Klaise, A. V. Looveren, G. Vacanti y A. Coca, «Alibi Explain: Algorithms for Explaining Machine Learning Models,» 2021. [En línea]. Available: <https://github.com/SeldonIO/alibi>.
- [17] M. Chan, «This AI reads children's emotions as they learn,» 2021. [En línea]. Available: <https://edition.cnn.com/2021/02/16/tech/emotion-recognition-ai-education-spc-intl-hnk/index.html>.
- [18] P. Farley, N. Mehrotra y E. Urban, «What is the Azure AI Face service?,» 2024. [En línea]. Available: <https://learn.microsoft.com/en-us/azure/ai-services/computer-vision/overview-identity#face-recognition>.
- [19] A. W. Services, «Amazon Rekognition,» 2023. [En línea]. Available: <https://aws.amazon.com/es/rekognition/>.

- [20] G. Cloud, «detect-faces,» 2024. [En línea]. Available: <https://cloud.google.com/vision/docs/detecting-faces?hl=es-419>.
- [21] P. Gundecha, «IBM Watson just got more accurate at detecting emotions,» 2016. [En línea]. Available: <https://www.ibm.com/blog/announcement/watson-has-more-accurate-emotion-detection/>.
- [22] EnableX, «EnableX Face AI,» 2024. [En línea]. Available: <https://www.enablex.io/cpaas/faceai/>.
- [23] M. Gorini, «El Magic Mirror asombra en el Mobile World Congress,» 2017. [En línea]. Available: <https://blog.bismart.com/magic-mirror-asombra-mwc>.
- [24] L. Bruno y A. I. Cuéllar, «“EMOTONGUE”, UNA APP PARA MANEJAR TUS PROPIAS EMOCIONES,» 2022. [En línea]. Available: https://www.uah.es/export/sites/uah/es/investigacion/.galleries/Oferta-Cientifico-Tecnologica/SOC_03ES.pdf.

APÉNDICES

Abreviaturas

AA. Aprendizaje Automático

API. Application Programming Interface (INterfaz de programación de aplicaciones)

CNN. Convolutional Neural Networks (Redes Neuronales Convolucionales)

FORGrad. FOurier Reparation of the Gradients

GPU. Graphics Processing Unit (Unidad de Procesamiento Gráfico)

Grad-CAM. Gradient-weighted Class Activation Mapping

KNN. K-Nearest Neighbors (K vecinos más cercanos)

LIME. Local Interpretable Model-agnostic Explanations

LSVRC. Large Scale Visual Recognition Challenge

ML. Machine Learning

MLP. Multi-layer Perceptron (Perceptrón multicapa)

MNIST. Modified National Institute of Standards and Technology

ReLU. Rectified Lineal Unit (Unidad Lineal Rectificada)

RGB. Red,Green,Blue

RISE. Randomized Input Sampling for Explanation

SHAP. SHapley Additive exPlanations

SVM. Support Vector Machine (Máquina de Vector Soporte)

UAH. Universidad de Alcalá de Henares

XAI. eXplainable Artificial Intelligence

