



# Optimización de las técnicas de Transfer Learning para la clasificación de la calidad estética en fotografía

Fernando Rubio

SIMD Lab, I3A

UCLM

Albacete, España

Email: fernando.rubio@uclm.es

M. Julia Flores

Departamento de Sistemas Informáticos

UCLM

España

Email: julia.flores@uclm.es

Jose M. Puerta

Departamento de Sistemas Informáticos

UCLM

España

Email: jose.puerta@uclm.es

**Abstract**—La evaluación automática de la calidad estética es un problema de visión por ordenador que consiste en cuantificar el atractivo de una fotografía. Esto es especialmente útil en las redes sociales, donde la cantidad de imágenes que se generan cada día requieren de la automatización para su procesamiento.

Aunque ha habido progresos notables en la investigación de este campo, aún es difícil encontrar soluciones aplicables. Con este trabajo buscamos la optimización de las soluciones más prometedoras basadas en Transfer Learning. Para ello, hemos reducido la complejidad de las redes neuronales propuestas manteniendo los resultados obtenidos, mediante técnicas de “finetuning” sobre redes pre-entrenadas.

**Index Terms**—Deep Learning, Transfer Learning, Finetuning, Classification, Computer Vision

## I. INTRODUCCIÓN

El campo de visión por ordenador es uno de los más activos en la comunidad científica debido a la gran cantidad de aplicaciones que tiene como la robótica y la seguridad. En los últimos años, las redes neuronales profundas han permitido resolver problemas, que hasta hace poco parecían inabordables. Esto ocurre también con el problema de la calidad estética.

El concepto de calidad estética en la fotografía hace referencia a las propiedades de las imágenes que las hacen atractivas o “bonitas” para la mayoría de la gente, como pueden ser los filtros aplicados, armonía de los colores, etc. No hay que confundir con la calidad de una imagen en términos de resolución. Se trata de uno de los problemas más complejos dentro del campo de visión debido a la subjetividad de la tarea, ya que la opinión de diferentes personas sobre la calidad de una única imagen puede ser muy distinta. Incluso entre expertos de fotografía puede haber opiniones diferentes.

A pesar de su dificultad, se trata de un problema que ha visto incrementado su interés enormemente debido a la gran cantidad de imágenes que se generan continuamente con las redes sociales. La automatización de esta tarea tiene aplicaciones muy interesantes como la ordenación de álbumes de imágenes en base a su calidad, especialmente útil para sitios como Flickr o Instagram. Pero también se puede utilizar para recomendaciones de filtros o incluso para evaluaciones online

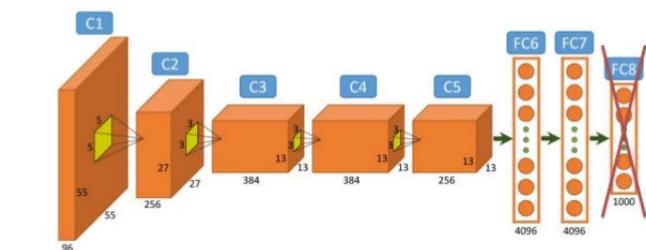


Fig. 1. Extracción de características de una red neuronal.

en una cámara, para mejorar la calidad de las fotografías que tomamos. Otro de los campos de aplicación es la publicidad, con la creación de imágenes más atractivas.

Tradicionalmente, la automatización de la evaluación de la calidad estética se centra en resolver un problema de clasificación binaria, donde las imágenes son clasificadas como “snapshots” (mala calidad) o “professional shots” (buena calidad). El uso del Deep Learning mediante las redes convolucionales ha mejorado los resultados en los últimos años e incluso ha permitido utilizar las probabilidades generadas en la última capa como indicadores más precisos de la calidad de la imagen.

En este artículo nos centramos en una de las estrategias más utilizadas recientemente en Deep Learning, Transfer Learning. Este concepto se basa en utilizar redes neuronales pre-entrenadas con otros datasets y aplicar dicho conocimiento a nuestro problema.

Actualmente, dos son las técnicas principales de Transfer Learning. La primera de ellas, puede verse en la Fig. 1 y que consiste en la extracción de características de una red neuronal, lo que se conoce como **ConvNet features** o **DeCaf** [1]. En este caso, obviamos la salida de la red y obtenemos las activaciones que se producen en las capas anteriores, para utilizarlas como *inputs* en otros modelos.

La segunda es el concepto de **finetuning**, que puede verse en la Fig. 2. Esta técnica de Transfer Learning consiste en la modificación de la (o las) última(s) capa(s) para ajustar

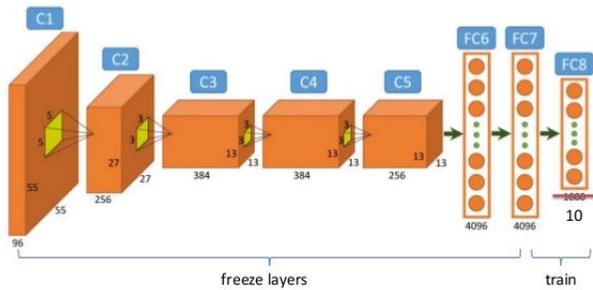


Fig. 2. Proceso de *finetuning*, donde se ha modificado la última capa de una red pre-entrenada, cuyos pesos van a ser aprendidos. El resto de capas no sufre modificaciones.

la salida de una red a nuestro problema. El siguiente paso consiste en el entrenamiento de los pesos sólo de las capas que han sido modificadas. Esto reduce el tiempo y la cantidad de datos necesarios para entrenar la red.

Cuando hablamos de Transfer Learning en imágenes, existen diferentes modelos pre-entrenados que podemos utilizar como son AlexNet [2], VGG [3], Inception [4], ResNet [5] o MobileNet [6]. Todos estos comparten una serie de propiedades y la base de datos con la que fueron entrenadas, ImageNet [7]. Este dataset consiste en un conjunto de más de 2 millones de imágenes de objetos (valla, barco, avión), animales (perro, gato, tortuga) o conceptos (atardecer, paisaje) que han sido etiquetadas con 1 de 1000 posibles “tags”. Por lo tanto, ImageNet es un problema de clasificación donde la clase puede tomar 1000 posibles valores. El objetivo de utilizar Transfer Learning sobre las redes aprendidas para este problema, es aprovechar todo el conocimiento generado por una red capaz de identificar 1000 conceptos diferentes en imágenes.

Los mejores resultados obtenidos para la clasificación de la calidad estética vienen de propuestas de Transfer Learning sobre dichos modelos, especialmente con *finetuning* [8]. Sin embargo, en el estudio de estas soluciones, por lo general, sólo la última capa es modificada para el reentrenamiento. En este trabajo proponemos realizar un proceso de *finetuning* de más capas, con el objetivo de reducir el tamaño de la red, pero sin afectar a los resultados obtenidos. Esto permitirá que las redes neuronales puedan utilizarse en dispositivos más limitados en recursos computacionales como son los dispositivos móviles o las cámaras fotográficas, permitiendo la creación de aplicaciones reales.

## II. ESTADO DEL ARTE

Las primeras aproximaciones para la evaluación de la calidad estética consisten en una clasificación binaria de las imágenes en “snapshots” o “professional shots”. Sin embargo, al tratarse de un problema con tanta incertidumbre y donde la clase no está bien definida, las principales bases de datos están compuestas por imágenes donde un grupo de individuos han asignado unos ratings o votos [9]. En la Fig. 3, podemos observar dos imágenes con su distribución de votos.

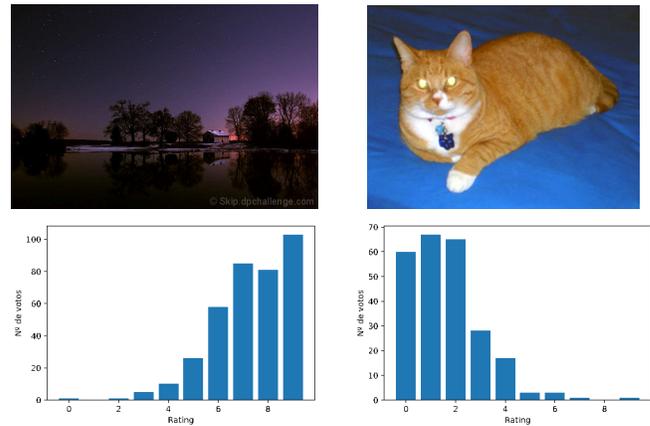


Fig. 3. Ejemplo de una imagen profesional (arriba, izquierda) y de una de mala calidad (arriba, derecha). En la parte inferior se muestra la distribución de los votos para cada imagen.

Generalmente, a partir de los ratings asignados a una imagen, obtenemos estadísticos que nos permiten convertir el problema en uno de clasificación binaria. Por ejemplo, podemos utilizar métricas como la media o la mediana para separar las imágenes en “snapshots” o “professional shots”. Normalmente, se fija un umbral en el punto medio del rango de los ratings, es decir, si los votos van en una escala del 1 al 10, el corte se establece en 5. Finalmente, se obtiene la media para cada imagen  $\mu_i$  a partir de sus votos y se comparan con el umbral. En la Fig 3, en el caso de la izquierda, tenemos una  $\mu = 8.31$ , y en la derecha  $\mu = 2.62$ . Al tener un rango de 10 posibles valores, el umbral se sitúa en el 5, por lo que la imagen de la izquierda sería clasificada como “professional shot” y la de la derecha como “snapshot”.

[10] y [11] proponen resolver el problema con características hechas a mano de bajo nivel para tratar de identificar propiedades más complejas de las fotografías y que sean capaces de separar ambas categorías. Sin embargo, estas propuestas pronto fueron superadas por técnicas generales de extracción de características, como GIST o SIFT [12].

Con la aparición del dataset AVA [13], el problema de la calidad estética ya contaba con una considerable base de datos de imágenes que permitía el aprendizaje de redes de Deep Learning como en [14]. Sin embargo, los resultados de entrenar una red neuronal desde cero, para este problema concreto, no han sido tan relevantes como en otros campos de visión.

[8] demuestra que realizando el proceso de *finetuning* en la última capa de los modelos AlexNet y VGG es posible obtener unos resultados más fiables que los presentados hasta ese momento. [15] también hace uso de esta técnica de Transfer Learning para predecir directamente la distribución de los votos, donde la capa de *output* tiene un tamaño de 10, correspondiente al rango de ratings de AVA, en vez de 2 de la clasificación binaria. En este último caso, en vez de *softmax* como función de activación de la última capa, utilizan Earth Mover’s Distance (EMD), que obtiene la distribución de probabilidad acumulada para la imagen.



Hay que destacar que la mayoría de los trabajos sufrían de un problema con la evaluación, como se indica en [16], ya que la única métrica utilizada en muchas propuestas para validar los modelos era la tasa de aciertos o *accuracy*. En AVA, esta métrica es poco informativa, ya que si binarizamos la clase a partir de los votos, cogiendo como umbral el 5 de media (ya que el rango va de 1 a 10), observamos un desbalanceo de la clase, donde el 70% de los casos son “professional shots” y el 30% “snapshots”. En este caso, reportar una tasa de aciertos del 70% no tiene valor, ya que se pueden estar clasificando todas las imágenes como buenas.

Para resolver esta situación, [16], [8] y [15] hacen uso de diferentes métricas como son el *balanced accuracy* que tiene en cuenta la tasa de aciertos por cada una de las clases o el valor AUC (Area Under the Curve), que relaciona la tasa de Verdaderos Positivos con la tasa de Falsos Positivos. En este trabajo utilizaremos esas métricas cuando trabajemos con problemas desbalanceados.

Es frecuente encontrar en la literatura propuestas que tienden a reducir las bases de datos originales en subconjuntos. Principalmente se eliminan imágenes cuya media de votos se encuentra cerca del umbral de corte. [14] y [12] utilizan una  $\delta$ , de forma que si consideramos 5 el punto de corte, las imágenes  $< 5 - \delta$  son consideradas “snapshots” y las  $> 5 + \delta$  son etiquetadas como “professional shots”, descartando el resto. Otros, como [17] o [18] seleccionan el 10% mejor y peor valoradas y el resto no se tienen en cuenta.

Es comprensible tratar de dar más peso a aquellas imágenes más informativas y tratar de reducir el ruido que pueden generar los casos cercanos a la frontera de decisión. Sin embargo, en algunas de estas propuestas, los resultados presentados han eliminado del conjunto de evaluación las imágenes, con el fin de simplificar el problema. Consideramos que estos resultados no son válidos para un escenario real, donde la mayoría de las imágenes se encuentran cerca del umbral de corte. La eliminación de imágenes sólo debe realizarse en los conjuntos de entrenamiento.

Por último, en las Fig.4 y 5 se muestran resultados preliminares de la evaluación de la calidad estética en AVA utilizando características generales de la imagen. En ambas se refleja la evolución de los resultados en base al valor  $\delta$  donde sólo se descartan las imágenes del conjunto de entrenamiento. Se observa que eliminar las imágenes cercanas al punto de corte no afecta de forma significativa a los resultados, he incluso en el caso del AUC vemos un peor comportamiento. Esto corrobora lo resultados de [14], donde la reducción de las imágenes también perjudica a los modelos de Deep Learning. Por estos motivos, en este trabajo utilizaremos las bases de datos completas.

### III. TRANSFER LEARNING

#### A. Dataset

Actualmente, la base de datos referente para el tratamiento de la calidad estética es AVA. Este dataset esta compuesto por cerca de 250.000 imágenes pertenecientes a una página de retos fotográficos, DPChallenge. Cada foto ha sido valorada

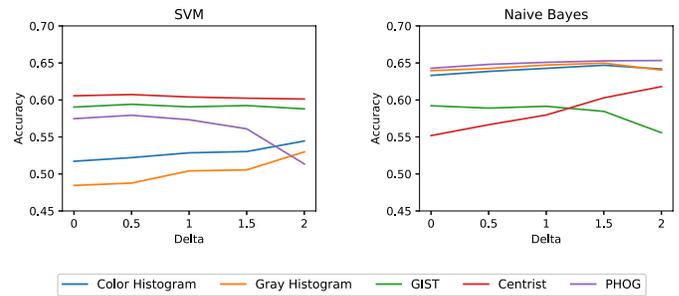


Fig. 4. Evolución del *accuracy* (tasa de aciertos) en base a la  $\delta$  (Delta) para la reducción de imágenes de la base de datos AVA.

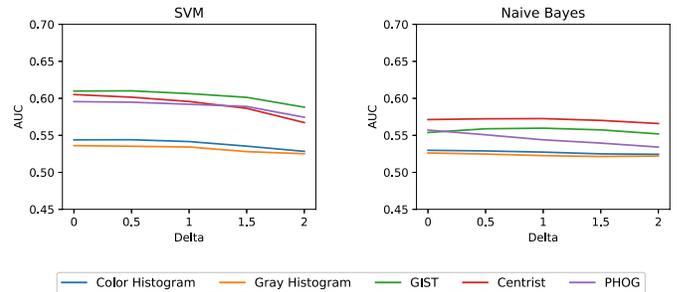


Fig. 5. Evolución del AUC en base a la  $\delta$  (Delta) para la reducción de imágenes de la base de datos AVA.

del 1 al 10 por diferentes usuarios, siendo el 1 que la foto es muy mala y 10 que es muy buena. Además, cada imagen viene etiquetada con el tipo de reto en la que se subió, el estilo fotográfico (si tiene alguno) y ciertas etiquetas sobre objetos que aparecen en la imagen. En los trabajos previos, esta base de datos es particionada en train y test, siendo este último de unas 20k imágenes y las 230k restantes para entrenamiento. En este trabajo utilizaremos la misma partición.

#### B. Extracción de ConvNet features

Durante la evaluación de una imagen en una red neuronal (proceso *forward*), no sólo se obtienen los valores de salida, en cada capa de la red se generan una serie de activaciones que pueden ser extraídas. Aunque es difícil interpretar esta información, puede utilizarse como vectores de características de la imagen, ya que tienen una gran capacidad descriptiva. Esto es lo que se conoce como *ConvNet features* y se utilizan generalmente en problemas donde no tenemos suficientes datos como para entrenar una red neuronal. Estas características son extraídas de redes pre-entrenadas y se utilizan para aprender otros modelos de clasificación.

En [16] se realiza un estudio del rendimiento de las *ConvNet features* extraídas de dos redes neuronales (AlexNet y ResNet) en la evaluación de la calidad estética. En las Fig. 6 y 7 se observan los resultados, donde los modelos entrenados con las *ConvNet features* superaban a los clasificadores que utilizaban descriptores generales de la imagen. Sin embargo, debido a la gran cantidad de características que se obtienen de las redes neuronales, no todos los modelos se muestran en este estudio.

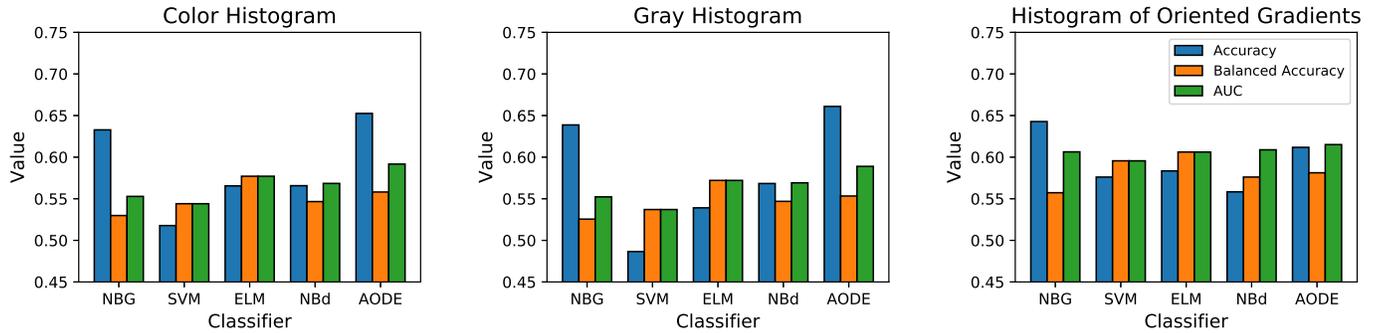


Fig. 6. Resultados de 5 clasificadores entrenados con descriptores generales de la imagen en términos de *accuracy*, *balanced accuracy* y *AUC*.

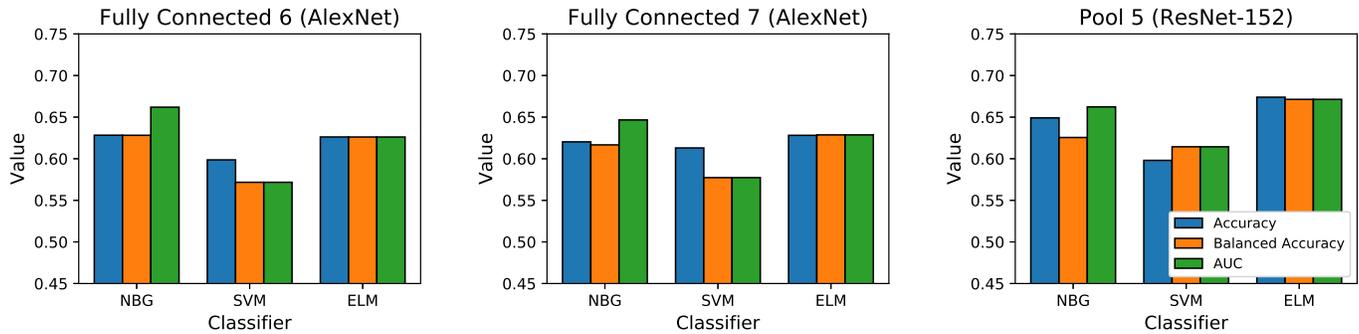


Fig. 7. Resultados de 3 clasificadores entrenados con *ConvNet features* de AlexNet y ResNet en términos de *accuracy*, *balanced accuracy* y *AUC*.

La reducción que vamos a ver a continuación, también tiene el objetivo de disminuir el tamaño de los vectores de características que se extraen de las redes, con el fin de poder utilizar modelos más complejos con esta técnica de Transfer Learning.

### C. Reducción de las capas densas mediante finetuning

Al igual que en la extracción de *ConvNet features*, para el proceso de *finetuning* es necesario disponer de modelos pre-entrenados en problemas similares para utilizar la información en nuestro beneficio.

AlexNet y VGG16 son redes convolucionales que siguen estructuras muy parecidas y que cuyas tres capas finales son del tipo *fully connected* o densas, y la última corresponde con la salida de las 1000 etiquetas del problema de ImageNet. Podemos ver la estructura de ambas redes en las Fig. 8 y 9, respectivamente. Las dos capas densas previas a la salida de la red, son llamadas “fc6” y “fc7”, y tienen dimensiones de 4096 nodos en cada una.

Como se ha comentado antes, la técnica de *finetuning* consiste en realizar modificaciones a una red pre-entrenada, para adaptar la salida del modelo a nuestro problema (Fig. 2). En el caso actual, donde la evaluación de la calidad estética se realiza mediante una clasificación binaria y las redes pre-entrenadas son del problema de ImageNet, sustituimos la capa densa de salida con 1000 nodos por una de 2 nodos.

En los trabajos propuestos que hacen uso de *finetuning*, todas las capas del modelo, exceptuando la de *output*, permanecen inalterables. Sin embargo, consideramos que esta

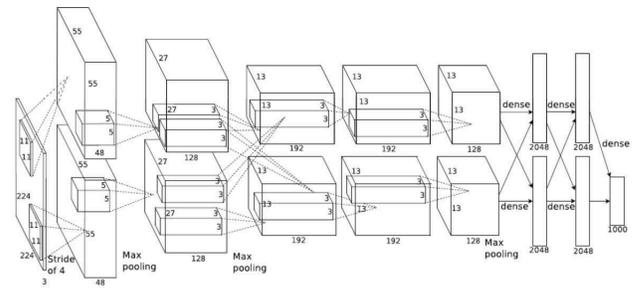


Fig. 8. Estructura original de AlexNet en dos columnas.

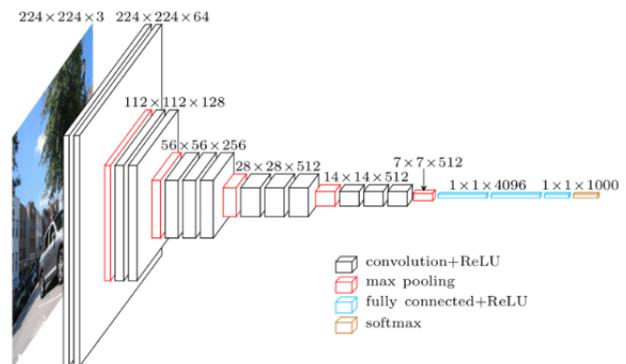


Fig. 9. Estructura de la red VGG16.



Modelo	Nodos en las capas densas	Nº de params de la red	Tamaño del modelo
AlexNet	4096	60M	223MB
	1000	14M	54MB
	500	8,6M	33MB
	250	6,1M	24MB
VGG16	4096	134M	513MB
	1000	41M	156MB
	500	28M	106MB
	250	21M	81M

TABLA I

NÚMERO DE PARÁMETROS (EN MILLONES) DE DIFERENTES REDES Y DEL ESPACIO NECESARIO EN MEMORIA (MEGABYTES) PARA ALMACENARLA, EN FUNCIÓN DEL TAMAÑO DE LAS CAPAS DENSAS.

técnica de Transfer Learning puede utilizarse de forma más eficaz aplicando modificaciones a las capas previas “fc6” y “fc7”, tanto en AlexNet como en VGG16.

En este trabajo, se propone reducir el número de nodos de las tres últimas capas de los modelos AlexNet y VGG16 para resolver la clasificación binaria de la calidad estética. Las capas densas son las que concentran el mayor número de parámetros de una red, por lo que mediante este proceso, disminuiríamos considerablemente el tamaño de la red.

#### D. Implementación

Para este trabajo se ha utilizado la librería TensorFlow [19] en Python para realizar el proceso de *finetuning* a las capas “fc6”, “fc7” y el *output* de las redes neuronales AlexNet y VGG16. En ambos casos se ha utilizado el optimizador SGD (Stochastic Gradient Descent) con un learning rate de 0.001 con un tamaño de batch de 128 y se han realizado 10 epochs. Para las capas de Dropout se ha utilizado un factor de 0,5. Todo esto se ha llevado a cabo sobre una GPU Tesla K40c, donde los tiempos de entrenamiento son de 6-7 horas para AlexNet y de 23 horas para VGG16.

#### IV. RESULTADOS

Se han realizado experimentos reduciendo los nodos de las capas densas “fc6” y “fc7” a 1000, 500 y 250. En la tabla I se muestra la diferencia de tamaño de las redes y la memoria necesaria. Como se puede observar, la mayoría de los parámetros de nuestra red se encuentran en estas capas densas, por lo que al reducir su tamaño afecta significativamente al modelo.

Una vez reentrenadas las redes, vamos a comparar sus resultados utilizando tres métricas de evaluación sobre el conjunto de test. Estas son el *accuracy* o tasa de acierto, el *balanced accuracy* y el valor AUC (Area Under the Curve).

En la tabla II se puede observar que la modificación de las capas densas en AlexNet afecta al *balanced accuracy* y al AUC. Existe un empeoramiento de un 3% del modelo con las capas densas de tamaño 250, frente a las de 4096, pero la red reducida ocupa un 10% de memoria con respecto a la original. Con VGG16 los resultados son casi idénticos en todas las redes, independientemente del tamaño de las capas densas. Cabe destacar que sólo el *accuracy* se ha visto afectado y que

Modelo	Nodos en las capas densas	Accuracy	Balanced Accuracy	AUC
AlexNet	4096	0.70	0.69	0.76
	1000	0.70	0.67	0.74
	500	0.69	0.67	0.73
	250	0.70	0.66	0.73
VGG16	4096	0.70	0.71	0.79
	1000	0.72	0.71	0.79
	500	0.74	0.71	0.79
	250	0.72	0.71	0.79

TABLA II

RESULTADOS EN TÉRMINOS DE ACCURACY, BALANCED ACCURACY Y AUC DE LAS REDES EN BASE AL TAMAÑO DE LAS CAPAS DENSAS.

la red con las capas densas de tamaño 500 funcionan mejor que las originales de 4096.

#### V. CONCLUSIÓN

En este trabajo se ha presentado una estrategia basada en *finetuning* capaz de reducir el tamaño de redes pre-entrenadas, en este caso AlexNet y VGG16, sin perder eficacia en la resolución del problema de la calidad estética.

Hemos visto como podemos obtener los mismos resultados con redes del 10% del tamaño de las propuestas hechas hasta ahora. Esto permitirá que los requisitos para la evaluación de nuevos casos sea mucho menor, permitiendo utilizar dichos modelos en dispositivos como móviles o, por ejemplo, una Raspberry. Estamos seguros de que es un gran paso para el desarrollo de aplicaciones reales que se beneficien del Deep Learning para la evaluación de la calidad estética.

Como trabajo futuro, planeamos seguir optimizando las redes pre-entrenadas en el problema de la calidad estética, especialmente los diseños presentados en [15], donde la salida es la distribución de los votos, en vez de la clasificación binaria en “snapshots” y “professional shots”.

La reducción del número de nodos de las últimas capas de las redes neuronales, además del ahorro de memoria que supone, permite extraer *ConvNet features* con una menor dimensionalidad, pero con la misma capacidad descriptiva. Esto es de especial utilidad para aprender modelos más complejos que antes no podíamos, como por ejemplo, algunos clasificadores probabilísticos, ya que estos modelos manejan de forma natural la incertidumbre y son especialmente útiles en problemas donde la clase no está definida.

#### AGRADECIMIENTOS

Este trabajo ha sido parcialmente financiado con los fondos FEDER y el Gobierno Español (MICINN) a través del proyecto TIN2016-77902-C3-1-P.

#### REFERENCIAS

- [1] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, “Decaf: A deep convolutional activation feature for generic visual recognition,” in *International conference on machine learning*, 2014, pp. 647–655.
- [2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in neural information processing systems*, 2012, pp. 1097–1105.
- [3] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

- [4] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich *et al.*, “Going deeper with convolutions.” *Cvpr*, 2015.
- [5] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [6] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” *arXiv preprint arXiv:1704.04861*, 2017.
- [7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 248–255.
- [8] Y. Deng, C. C. Loy, and X. Tang, “Image aesthetic assessment: An experimental survey,” *IEEE Signal Processing Magazine*, vol. 34, no. 4, pp. 80–106, 2017.
- [9] R. Datta, J. Li, and J. Z. Wang, “Algorithmic infereencing of aesthetics and emotion in natural images: An exposition,” in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*. IEEE, 2008, pp. 105–108.
- [10] R. Datta, D. Joshi, J. Li, and J. Z. Wang, “Studying aesthetics in photographic images using a computational approach,” in *European Conference on Computer Vision*. Springer, 2006, pp. 288–301.
- [11] Y. Ke, X. Tang, and F. Jing, “The design of high-level features for photo quality assessment,” in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 1. IEEE, 2006, pp. 419–426.
- [12] L. Marchesotti, F. Perronnin, D. Larlus, and G. Csurka, “Assessing the aesthetic quality of photographs using generic image descriptors,” in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1784–1791.
- [13] N. Murray, L. Marchesotti, and F. Perronnin, “Ava: A large-scale database for aesthetic visual analysis,” in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2408–2415.
- [14] X. Lu, Z. Lin, H. Jin, J. Yang, and J. Z. Wang, “Rapid: Rating pictorial aesthetics using deep learning,” in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 457–466.
- [15] H. Talebi and P. Milanfar, “Nima: Neural image assessment,” *IEEE Transactions on Image Processing*, vol. 27, no. 8, pp. 3998–4011, 2018.
- [16] F. Rubio, M. J. Flores, and J. M. Puerta, “Drawing a baseline in aesthetic quality assessment,” in *Tenth International Conference on Machine Vision (ICMV 2017)*, vol. 10696. International Society for Optics and Photonics, 2018, p. 106961M.
- [17] X. Tian, Z. Dong, K. Yang, and T. Mei, “Query-dependent aesthetic model with deep learning for photo quality assessment,” *IEEE Transactions on Multimedia*, vol. 17, no. 11, pp. 2035–2048, 2015.
- [18] Z. Dong, X. Shen, H. Li, and X. Tian, “Photo quality assessment with dcnn that understands image well,” in *International Conference on Multimedia Modeling*. Springer, 2015, pp. 524–535.
- [19] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, “TensorFlow: Large-scale machine learning on heterogeneous systems,” 2015, software available from tensorflow.org. [Online]. Available: <https://www.tensorflow.org/>