Búsquedas por Similitud de Logos: Extracción de Características usando IA en Escenarios de Datos Escasos

Andrés J. Pascal
FRCU - Universidad Tecnológica Nacional
pascala@frcu.utn.edu.ar

Agustina Bonti
FRCU - Universidad Tecnológica Nacional
agustinabontiutn@gmail.com

Zoe Florencia Vidal
FRCU - Universidad Tecnológica Nacional
zoevidal523@gmail.com

Iván Federico Bonti
FRCU - Universidad Tecnológica Nacional
ivanbonti300@gmail.com

Lucas Francisco Tonelotto
FRCU - Universidad Tecnológica Nacional
tonelottolucas@gmail.com

Resumen

En el panorama actual, las Búsquedas por Similitud emergen como un ámbito de profundo interés. La evaluación de la similitud entre objetos generalmente involucra el empleo de funciones métricas de distancia aplicadas a vectores que representan características extraídas a partir de los mismos. Este artículo se enfoca en la extracción de características aplicada a imágenes de logos de clubes, utilizando técnicas modernas de aprendizaje automático; en particular, Redes Neuronales Profundas Convolucionales (CNN), Redes Siamesas y Transfer Learning/Fine Tuning. Si bien estas técnicas son muy potentes, su aplicación conlleva en algunos casos el desafío del entrenamiento ante datos escasos (One Shot Learning, en este caso). En este estudio comparamos dos enfoques de extracción de características en el contexto de escasez de datos, proponemos un método eficaz de preprocesamiento, y evaluamos experimentalmente el rendimiento de ambos métodos aplicados a la búsqueda por similitud de logos.

Palabras Clave: Búsquedas por Similitud, Logos, Extracción de Características, CNNs, Redes Siamesas, One-Shot Learning, Aumentación.

1. Introducción

Con la incorporación de datos no estructurados como imágenes, audio, video y texto, los modelos de búsqueda tradicionales se vuelven insuficientes debido a la imposibilidad de compararlos por exactitud. La búsqueda por similitud emerge como una solución para encontrar objetos similares a un elemento de consulta específico en las bases de datos no tradicionales. Los Espacios Métricos [1] son un modelo que formaliza las búsquedas por similitud y permite el uso de métodos de acceso más eficientes.

Este artículo presenta un estudio sobre la extracción de características en imágenes a color con el propósito de llevar a cabo búsquedas de similitud de logos de clubes. A diferencia de trabajos previos que se centraron en imágenes en blanco y negro o escala de grises, este estudio se adentra en el terreno de las imágenes a color, ofreciendo resultados prometedores.

En la última década, las Redes Neuronales Convolucionales (CNN) y sus variantes han emergido como métodos de procesamiento de imágenes preeminentes, eclipsando notablemente el rendimiento de técnicas previas que eran consideradas como el estado del arte. Si bien las CNN son ampliamente reconocidas por su habilidad en la extracción de características de imágenes, el contexto de las consultas por similitud presenta dos desafíos específicos: a) se dispone usualmente de una sola muestra de cada imagen (One-Shot Learning) y b) el tamaño de la base de datos se encuentra en un continuo crecimiento, lo que torna costoso y poco práctico modificar y reentrenar el modelo cada vez que se incorpora un nuevo elemento.

En este artículo, se experimentan modelos de CNN y técnicas de preprocesamiento y aumentación para la extracción de características de imágenes de logos de clubes, que luego se utilizan para medir la similitud (disimilitud, en realidad) utilizando la distancia euclidiana.

El resto de este documento está organizado de la siguiente manera: la Sección 2 presenta trabajos previos

relacionados, incluida una breve explicación de la búsqueda de imágenes por contenido, el modelo de espacios métricos, CNNs, Transfer learning y Redes Siamesas y búsqueda/reconocimiento de logos. En la Sección 3 se explican los procesos de extracción de características y modelos utilizados. En la Sección 4 describe el esquema de un sistema de consultas por similitud. La Sección 5 muestra los experimentos realizados y la 6 los resultados obtenidos. Finalmente, en la Sección 7 se exponen las conclusiones del estudio.

2. Trabajos Previos

Esta sección presenta el contexto del presente estudio. Se abordan las técnicas claves para utilizar CNNs y Redes Siamesas para la obtención de vectores de características para las comparación por similitud de logos. También se describen trabajos anteriores que abordan el mismo problema, aunque de diferente manera.

2.1 CBIR

La Recuperación de Imágenes Basada en Contenido (CBIR, por su sigla en inglés) [2] es un proceso que permite recuperar imágenes de una base de datos teniendo en cuenta diversas características visuales. Los principales tipos de características utilizados en CBIR incluyen el color, la textura y la forma [3, 4]. Este enfoque ha revolucionado la forma en que organizamos y exploramos vastas colecciones de imágenes digitales, acelerando la búsqueda de imágenes relevantes.

El color se destaca como la característica visual más ampliamente utilizada en CBIR, en gran parte debido a la facilidad de extraer datos cromáticos de las imágenes [5]. En contraste, obtener información sobre la forma y la textura [6] es un proceso mucho más complejo y costoso. Estos dos últimos componentes son fundamentales en la descripción del contenido de las imágenes, pero presentan desafíos únicos en su análisis y representación.

Los histogramas [7] son una solución popular para modelar características de imágenes en CBIR. Cada histograma representa la distribución de niveles de gris o colores presentes en una imagen dada. Si bien los histogramas son computacionalmente eficientes, presentan limitaciones importantes. Por ejemplo, carecen de información espacial y son sensibles a cambios en el brillo general de la imagen [8]. Esto los hace menos adecuados para describir formas y objetos en imágenes de manera precisa.

La forma es una característica visual fundamental utilizada para describir el contenido de las imágenes, pero su representación y descripción plantean desafíos significativos. La forma de un objeto puede verse afectada por diversos factores, como defectos, ruido, oclusión y distorsiones arbitrarias, lo que dificulta su análisis. Una forma puede ser descrita desde múltiples perspectivas [9], incluyendo el centro de gravedad (centroide) [10], masa, media, dispersión, varianza, eje de menor inercia, rectangularidad y convexidad. Para una mejor representación de formas, se han adoptado enfoques más avanzados, como el uso de descriptores invariantes, como Momentos de Hu, Legendre o Zernike [11, 12, 13], que ofrecen resultados más confiables y precisos en el análisis de formas complejas.

A medida en que se avanza en esta área de investigación, es fundamental seguir explorando nuevas técnicas y enfoques para mejorar la precisión y eficiencia en la descripción y recuperación de formas en imágenes.

2.2 Búsquedas en Espacios Métricos

Los Sistemas de Recuperación de Imágenes Basados en Contenido (CBIR) pueden ser generalizados y modelados mediante Espacios Métricos con el fin de lograr búsquedas eficientes. En el artículo [1], se muestra que el problema de búsqueda de similitud puede ser formulado de la siguiente manera: dado un conjunto U de objetos y una función de distancia d definida entre ellos para cuantificar su similitud, el objetivo es recuperar todos los elementos similares a un objeto dado utilizando d como criterio.

Esta función d satisface las propiedades requeridas para ser una métrica:

(a) $\forall x \in U,\ d(x, x)=0$ (reflexividad)

(b) $\forall x, y \in U,\ d(x, y) \geq 0$ (positividad)

(c) $\forall x, y \in U,\ d(x, y)=d(y, x)$ (simetría)

(d) $\forall x, y, z \in U,\ d(x, z) \leq d(x, y)+d(y, z)$ (desigualdad triangular)

En un espacio métrico (U, d), donde U es el conjunto de objetos y d es la función de distancia, la similitud entre dos objetos aumenta a medida que su distancia disminuye. Un subconjunto finito X de U, denominado base de datos, se utiliza para realizar la búsqueda. En espacios métricos, existen dos importantes tipos de consultas por similitud:

(a) Consulta por Rango o $(q, r)_d$: devuelve todos los elementos que se encuentra como máximo a una distancia r de q.

$$(q, r)_d = \{x \in X / d(q, x) \leq r\}$$

(b) Consulta de los k Vecinos Más Cercanos o $NN_k(q)_d$: recupera los k elementos de X, más cercanos a q.

$$NN_k(q)_d = A,$$

$$|A|=k,$$

$$A=\{x \in X / \forall y \in (X-A),\ d(q, x) \leq d(q, y)\}$$

Dada una base de datos con n objetos, responder a estas consultas de forma trivial tiene orden $O(n)$, lo cual puede resultar altamente costoso en aplicaciones prácticas. La relevancia de modelar estas consultas mediante espacios métricos se basa en la posibilidad de

utilizar índices que aprovechan la propiedad de la desigualdad triangular para descartar elementos de la base de datos sin necesidad de compararlos directamente con la consulta. Esto optimiza significativamente el proceso de búsqueda [14-18], convirtiéndolo en una solución altamente eficiente y beneficiosa para diversas aplicaciones.

2.3 Redes Neuronales Convolucionales, Redes Siamesas y Transfer Learning

Si bien las Redes Neuronales Convolucionales (CNNs) fueron propuestas a fines de los ‘80 y durante los ‘90 [19, 20], recién durante esta última década experimentaron enormes avances que las convirtieron en algunas de las técnicas principales en el procesamiento de imágenes.

La arquitectura general de una CNN consta de dos componentes principales: la extracción de características mediante capas de convolución y submuestreo, y el clasificador, que generalmente utiliza capas densas para obtener resultados óptimos en tareas de clasificación [21, 22]. Estas características hacen de las CNNs una herramienta poderosa y versátil para el análisis y reconocimiento de patrones en imágenes, lo que ha impulsado su amplia adopción y éxito en una variedad de aplicaciones prácticas y académicas.

Las arquitecturas actuales de CNNs consisten típicamente en la combinación de varias capas convolucionales y de pooling, en su mayoría con activación ReLU, seguidas por capas densas más SoftMax hacia el final. Algunos ejemplos importantes de tales modelos son AlexNet [23], VGG Net [24] DenseNet [25], GoogLeNet (Inception) [26, 27, 28], y Residual Networks (ResNet) [29]. Los componentes básicos son casi los mismos para todas las arquitecturas, sin embargo, las diferencias topológicas producen distintos resultados tanto en la eficiencia en el entrenamiento como en la precisión en la clasificación.

El algoritmo de aprendizaje de las Redes Neuronales Convolucionales (CNNs) no incluye de forma inherente el concepto de similitud entre imágenes. Sin embargo, en investigaciones recientes, se han introducido arquitecturas como las Redes Neuronales Siamesas [30, 31] y funciones de pérdida especiales como Triplet Loss [32, 33]. Estas arquitecturas han demostrado ser eficaces para capturar la noción de similitud entre imágenes de entrada, especialmente en aplicaciones como el reconocimiento facial. Las Redes Siamesas se componen de dos CNNs idénticas que operan en paralelo. Estas redes se utilizan para extraer vectores de características de las imágenes de entrada, y luego estos vectores se comparan a través de una función de distancia. Durante el proceso de entrenamiento, la red ajusta sus parámetros para minimizar la distancia entre los vectores correspondientes a imágenes similares y, al mismo tiempo, maximizar la distancia entre los vectores correspondientes a imágenes diferentes. Este enfoque de aprendizaje ha arrojado resultados prometedores en tareas que dependen de la similitud visual entre imágenes.

El uso de Redes Siamesas/CNNs para la extracción de características orientadas a las Búsquedas por Similitud posee dos importantes problemas:

1. El primero es la escasez de instancias disponibles para el entrenamiento, problema conocido como One-Shot Learning o Few-Shots Learning [34, 35]. Esta situación imposibilita el entrenamiento directo de los modelos. La mejor estrategia hasta el momento para abordar este problema es el Transfer Learning/Fine Tuning [36-39], donde se utilizan modelos previamente entrenados con bases de datos que contienen objetos de características similares. Sin embargo, esta técnica se ve limitada por la disponibilidad de dichas bases de datos. Recientes investigaciones, como en [40], proponen algoritmos que podrían superar estas limitaciones y mejorar la capacidad de reconocimiento en escenarios de pocos ejemplos de entrenamiento.
2. El segundo problema consiste en que cuando se usan las CNNs para extraer vectores de características, dichos vectores están altamente ligados a las clases con las cuales se entrenan los modelos, lo que limita su capacidad de generalización. Estos vectores no logran adaptarse de manera efectiva a nuevas imágenes incorporadas a la base de datos, correspondientes a las nuevas "clases".

2.4 Reconocimiento/Consultas por Similitud de Logos

En [41] se propone un novedoso enfoque para la detección y reconocimiento de logotipos en imágenes mediante el uso de características locales y la consideración de la geometría y el contexto espacial de estas características. El método se basa en la minimización de una función de energía que combina la calidad de la coincidencia de características, la co-ocurrencia y geometría de características, y un término de regularización para controlar la suavidad de la solución. Se demuestra su eficacia a través de experimentos en el conjunto de datos MICC-Logos.

[42] presenta un método para la identificación rápida de similitudes en imágenes de logotipos utilizando el algoritmo SIFT para extraer características invariantes a la escala y la rotación, y la medida de similitud se basa en el número de puntos clave coincidentes. El método se valida utilizando una base de datos de imágenes preparada por los autores. El enfoque se centra en identificar elementos comunes (patrones) en lugar de reconocer o identificar logotipos en sí.

En estudios más recientes [43] se propone la recuperación de logos mediante la fusión de información de DarkNet-19 y DarkNet-53, utilizando el conjunto de datos de Flickr Logos-47 para experimentar y evaluar el rendimiento del sistema; y [44] en el cual se buscó medir el nivel de similitud entre logos mediante

el uso de ResNet-18, alcanzando un 93,65% de precisión luego de 84 iteraciones.

3. Estructura del Sistema de Consultas por Similitud

En un sistema de búsqueda por similitud de imágenes, estos elementos se almacenan en la base de datos junto a sus vectores de características. El cálculo de estos vectores implica inicialmente un proceso de preprocesamiento de cada imagen para su limpieza y normalización, seguido de la ejecución de un algoritmo encargado de extraer sus características distintivas. Luego, se establece un índice métrico que utiliza como entrada los vectores de características, para mejorar la eficiencia del proceso de búsqueda.

A continuación, se resumen los pasos relacionados a la preparación de la base de datos y los pasos correspondientes a las consultas.

a) Alta de Objetos: cada vez que se agrega un nuevo elemento se ejecutan los siguientes pasos:
   1. Preprocesamiento de la Imagen: este paso incluye procesos tales como obtención del rectángulo delimitador mínimo (MBR), redimensionamiento de la imagen a una medida estándar, reducción de ruido y normalización.
   2. Extracción de Características: una vez que la imagen está normalizada, se ejecuta un proceso que devuelve el vector de características correspondiente.
   3. Almacenamiento: la imagen y su vector característicos se registran en la base de datos.
   4. Actualización del índice métrico: mediante la inserción del vector correspondiente al nuevo elemento.

b) Consulta por Similitud: dada una imagen de consulta, se realiza una Búsqueda por Rango o de los *k* Vecinos más Cercanos (NN_k) ejecutando los pasos 1 y 2 del punto anterior sobre la consulta y luego utilizando el índice para descartar elementos y obtener las imágenes similares con mayor eficiencia.

Es de destacar, que el modelo de extracción de características debe servir tanto para la imágenes con las cuales fue entrenado como para imágenes nuevas que se agreguen a la BD.

4. Extracción de Características

En esta sección se presenta el mecanismo completo utilizado para realizar la extracción de características, incluyendo preprocesamiento, aumentación y entrenamiento de la Red Siamesa, y la transferencia de aprendizaje a partir de arquitecturas conocidas preentrenadas. Además se describe la base de datos utilizada.

4.1 Base de Datos de Logos

En este estudio se utilizó como base de datos un conjunto compuesto por 100 imágenes de logos de clubes extraídos de internet. La imágenes están en formato PNG de 32 bits de profundidad y resolución de 100x100 pixels.

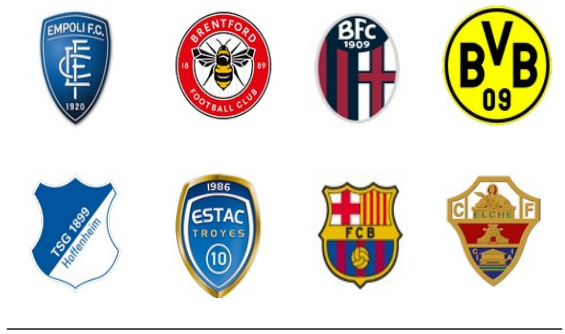

Figura 1. Elementos de la Base de Datos de Logos

En la Figura 1 se muestran algunos ejemplos de dichos logos. En su mayoría corresponden a clubes de fútbol.

4.2 Preprocesamiento

Si bien el método de preprocesamiento es simple, resulta extremadamente importante tanto para el entrenamiento de la Red Siamesa como para la normalización de la consulta. El primer paso es la obtención del Minimum Bounding Box, con el objetivo de descartar la parte no utilizada del "lienzo". Una vez calculado la imagen se recorta obteniendo así el área donde está ubicado el logo. Este proceso es fundamental para que el sistema sea robusto ante el escalado de las imágenes.

Posteriormente se reescala a 100x100 con deformación, es decir, sin mantener las proporciones. Esto hace que la imagen en sí ocupe toda el área del lienzo. Esta transformación produce robustez ante la traslación y ante pequeñas rotaciones.

Dicho proceso se realiza tanto para las imágenes de la base de datos como para las de entrenamiento y para las consultas.

4.3 Aumentación

En los problemas reales de Búsquedas por Similitud se suele contar con una sola muestra de cada objeto de la base de datos utilizada. Esto es un gran problema cuando se decide utilizar CNNs como método de extracción de características.

En este caso de estudio, a partir de 100 imágenes originales en la base de datos, se generaron 320 pares

por cada una de ellas mediante las siguientes técnicas de aumento estándares de datos:

- *Rotación*: en sentido horario y antihorario, hasta un ángulo de 20 grados.
- *Brillo*: se disminuyó hasta un 50% de la imagen original y se aumentó hasta un 150%.
- *Contraste*: desde un 25% hasta 175% de la imagen original.

Debido al método utilizado de preprocesamiento, no fue necesario generar imágenes escaladas o trasladadas.

Posteriormente, tomando como base las imágenes aumentadas, se obtuvieron en forma aleatoria 32.000 pares, balanceando las cantidades de pares similares y no similares y asegurando que cada logo participe en al menos 320 pares. De la generación de los pares similares, la mitad se realizó utilizando la imagen original vs una aumentada y la otra mitad entre dos imágenes aumentadas. Para los pares distintos se siguió un criterio similar. Un 30% de los mismos se utilizó para la validación.

4.4 Red Siamesa

El modelo interno de la Red Siamesa, es decir, el extractor de características utilizado, está formado por 5 capas convolucionales y una de pooling máximo, intercaladas con capas de normalización. Para disminuir la posibilidad de overfitting se utilizaron tres capas dropout intercaladas. Además se utilizó una capa de pooling promedio global y por último la capa flatten. En la Tabla 1 se muestra su arquitectura.

Tabla 1. Modelo Interno de la Red Siamesa

Capa	Kernels	Activación
Conv2D	32, 3x3	ReLU
BatchNormalization		
Conv2D	64, 3x3	ReLU
BatchNormalization		
Dropout		
Conv2D	32, 3x3	ReLU
BatchNormalization		
MaxPooling2D	2x2	
Conv2D	64, 3x3	ReLU
BatchNormalization		
Dropout		
Conv2D	256, 3x3	ReLU
BatchNormalization		
GlobalAveragePooling2D		
Dropout		
Flatten		
Dense (FC, 128)		

Es de destacar que todas las capas convolucionales utilizaron “padding=same”, para mantener las dimensiones de las matrices resultantes.

Por último el modelo contiene una capa densa de 128 neuronas y sin función de activación, que produce el vector de características. El modelo posee 237,856 parámetros entrenables.

La Red Siamesa en sí contiene dos capas de entrada, una para cada imagen, que están enlazadas al modelo interno. Los vectores resultantes obtenidos a partir de las dos imágenes de entrada se restan y luego se aplica el valor absoluto. El vector resultante se une a una última capa densa de una sola neurona con función de activación Sigmoide, que debe devolver un valor cercano a 0 si las imágenes son similares, y cercano a 1 si no lo son.

Como función de pérdida, en lugar de utilizar Binary Cross Entropy (que es la más común en este tipo de redes), se utilizó Mean Squared Error, con la cual se obtuvo mucho mejores resultados en los experimentos. El optimizador fue Adam.

4.5 Transfer Learning/Fine Tuning

En cuanto a la extracción de características a partir de redes preentrenadas, se seleccionaron las arquitecturas DenseNet121, InceptionV3 y ResNet50, todas preentrenadas con ImageNet. Se utilizó el preprocesamiento descripto previamente tanto para normalizar las imágenes como para adaptar su resolución a las esperadas por cada una de estas redes.

5. Descripción de los Experimentos

Para verificar la eficacia de la extracción de características mediante la Red Siamesa y los mecanismos de transferencia del aprendizaje, se realizaron distintos experimentos. Para ello se seleccionaron 60 logos descargados de internet como lote de consulta, asegurándose en todos los casos que las imágenes no eran exactamente iguales a las incluidas en la base de datos (ya que en este caso la respuesta a la consulta es trivial). Luego se realizaron las búsquedas por similitud de los k vecinos más cercanos (NN_k) para k =1, 3 y 5. La función de distancia empleada fue la Euclidiana y se utilizó como métrica de rendimiento el porcentaje de aciertos, ya que es el indicador que mejor representa la eficacia de las consultas por similitud en las aplicaciones reales.

Se realizaron seis experimentos, uno para cada uno de los métodos de extracción de características:

- Siamesa20: Red Siamesa entrenada durante 20 épocas con solo una muestra por cada logo, utilizando la aumentación estándar descripta.
- DenseNetU: extracción de características utilizando DenseNet121 entrenada sobre ImageNet, tomando una de las últimas capas como vector de características.
- DenseNetUFT: extracción de características utilizando DenseNet121 entrenada sobre

ImageNet, fijando un 90% de las capas y dejando el 10% restante entrenable y posteriormente realizando ajuste fino sobre dichas capas.

- DenseNetM: extracción de características utilizando DenseNet121 entrenada sobre ImageNet, tomando una capa del centro de la red como vector de características.
- InceptionM: extracción de características utilizando InceptionV3 entrenada sobre ImageNet, tomando una capa del centro de la red como vector de características.
- ResnetM: extracción de características utilizando Resnet50 entrenada sobre ImageNet, tomando una capa del centro de la red como vector de características.

Previo a las pruebas, se asoció a cada consulta la etiqueta del elemento de la base de datos que debía devolverse como similar, con el fin de calcular el porcentaje de aciertos en forma automática.

En el caso de la Red Siamesa, el entrenamiento se realizó en 20 épocas en dos horas una PC con procesador i5, 16 GB RAM y GPU GeForce GTX 960 con 1024 núcleos CUDA.

6. Resultados Obtenidos

La Fig. 2 muestra tres consultas por similitud (primera columna) correspondientes a los logos del A.C. Milan, Aston Villa y Deportivo Alavés. Las columnas restantes representan los tres vecinos más

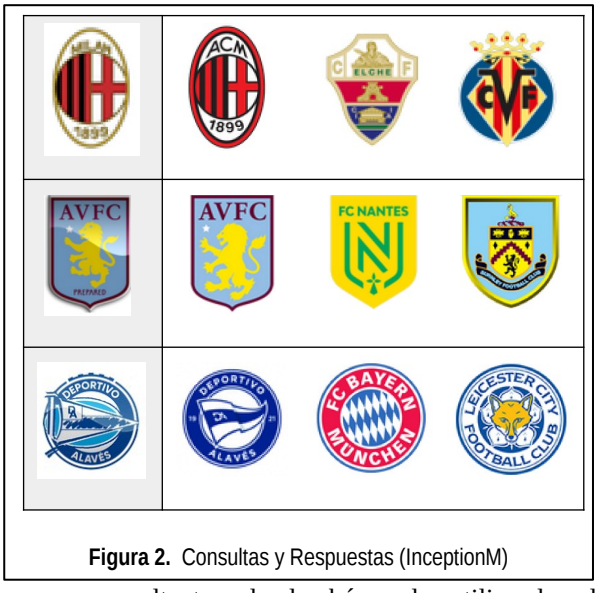

Figura 2. Consultas y Respuestas (InceptionM)

cercanos resultantes de la búsqueda utilizando el modelo InceptionM . Como se puede ver, las imágenes de consulta tienen diferencias relativamente importantes en comparación con las contenidas en la BD. En el caso del Milan, la imagen de consulta tiene la leyenda “MILAN” mientras que la de la respuesta dice ACM. Por otro lado, los colores del contorno son distintos y además poseen diferentes resoluciones: la consulta es de 100x56 y la respuesta en 100x100.

En el caso de Aston Villa, la consulta tiene superpuesta una “sombra” que produce partes más oscuras y otras más claras y la figura del león es más pequeña que la de la respuesta. Por último, el logo de consulta del Deportivo Alavés difiere tanto en colores como en la forma del banderín interno y tiene otros detalles que no posee la imagen correspondiente en la base de datos.

Por otro lado, es de notar que estos modelos interpretan muy bien las formas. Aunque en el caso del Milan los escudos en la segunda y tercera posición no tienen exactamente la misma forma que la consulta, en los otros dos casos ésto es evidente.

En cuanto a los resultados, en la Tabla 2 se muestran los porcentajes de acierto en las búsquedas por similitud para los *k* vecinos mas cercanos (NN_k), para k=1, 3 y 5. Como podemos observar, la Red Siamesa entrenada sólo con aumentación estándar luego del preprocesamiento de las imágenes obtiene buenos resultados (89,65% para los 5 vecinos más cercanos) e incluso supera a DenseNet entrenada sobre ImageNet, con y sin Fine Tuning, para el caso en que se toma una de las últimas capas previas a las FC para obtener los vectores característicos (DenseNetU y DenseNetUFT). Como se ve en la tabla, el ajuste fino produjo mejoras pero no fueron significativas.

Tabla 2. Porcentajes de Acierto

Modelo	NN_1	NN_3	NN_5
Siamesa20	70,68 %	81,03 %	89,65 %
DenseNetU	67,24 %	77,58 %	81,03 %
DenseNetUFT	70,68 %	75,86 %	82,75 %
DenseNetM	89,65 %	94,82 %	96,55 %
InceptionM	96,55 %	100,00 %	100,00 %
ResNetM	96,55 %	100,00 %	100,00 %

Lo notable es el rendimiento del Transfer Learning tanto de DenseNet, Inception y ResNet (89,65%, 96,55% y 96,55% para el vecino más cercano) cuando se utilizan las capas centrales para la extracción de los vectores característicos. En el caso de las búsquedas por similitud, parece no ser necesario y probablemente tampoco conveniente realizar el ajuste fino ya que no es deseable que la red se adapte específicamente a las “clases” que contiene la base de datos, sino que tiene que generalizar suficientemente bien como para permitir el agregado de nuevas imágenes sin realizar ninguna modificación del modelo y sin reentrenamiento, independientemente de los objetos que se agreguen a la base de datos.

Estos resultados son excelentes para una aplicación real de consultas por similitud y pueden ser utilizados con otros tipos de imágenes, con la limitante de que en los casos de Transfer Learning tienen que existir modelos preentrenados sobre bases de datos de características parecidas. Por otro lado, la Red Siamesa

no tiene esa limitación, aunque su rendimiento en este estudio sea un poco menor.

7. Conclusiones

En este estudio se exploró la extracción de características aplicada a imágenes de logos de clubes con el propósito de permitir búsquedas por similitud en escenarios de datos escasos. Se investigaron y compararon diversos métodos, incluyendo el uso de Redes Neuronales Siamesas y Transfer Learning con arquitecturas preentrenadas como DenseNet, Inception y ResNet. Los resultados revelaron que la Red Siamesa, entrenada con técnicas de preprocesamiento previo y aumentación de datos, alcanza un rendimiento razonable en casos de One-Shot Learning y no depende de la existencia de modelos preentrenados sobre un conjunto de datos con similares características. Sin embargo, el Transfer Learning superó notablemente su rendimiento cuando se extrajeron características de las capas centrales de las redes preentrenadas. Estos hallazgos ofrecen valiosas perspectivas para la implementación práctica de sistemas de búsqueda por similitud de imágenes en aplicaciones que involucran logos de clubes y objetos similares.

Las tareas previstas para el futuro próximo son las siguientes:

- Realizar nuevamente los experimentos pero con un conjunto de datos mayor. Actualmente contamos con más de 10.000 logos con los cuales realizaremos nuevos experimentos.
- Extender estos estudios a otros tipos de imágenes a color.
- Modificar la Red Siamesa para que utilice Triplet Loss como función de pérdida.
- Analizar estrategias para que la Red generalice mejor, de tal manera de que sea robusta ante la incorporación de nuevas imágenes.

Referencias

[1] Chávez, Edgar, et al. Searching in metric spaces. *ACM computing surveys (CSUR)* 33.3: 273-321, (2001).

[2] Wang, J. Z., Li, J., Wiederhold, G., & Firschein, O. Content-Based Image Retrieval at the End of the Early Years. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(12), 1349-1380, (2001).

[3] Aslandogan, Y. Alp, and Clement T. Yu. Techniques and systems for image and video retrieval. *IEEE transactions on Knowledge and Data Engineering* 11.1 (1999): 56-63.

[4] Smeulders, Arnold WM, et al. Content-based image retrieval at the end of the early years. *IEEE Transactions on pattern analysis and machine intelligence* 22.12 (2000): 1349-1380.

[5] Valova, Irena, Boris Rachev, and Michael Vassilakopoulos. Optimization of the algorithm for image retrieval by color features. *International Conference on Computer Systems and Technologies-CompSysTech*. (2006).

[6] Sarfraz, Muhammad, and Ahmad Ridha. Content-based image retrieval using multiple shape descriptors. *2007 IEEE/ACS International Conference on Computer Systems and Applications*. IEEE, (2007).

[7] Pass, Greg, and Ramin Zabih. Histogram refinement for content-based image retrieval. *Proceedings Third IEEE Workshop on Applications of Computer Vision. WACV'96*. IEEE, (1996).

[8] Zhang, HongJiang, et al. Image retrieval based on color features: An evaluation study. *Digital Image Storage and Archiving Systems*. Vol. 2606. SPIE, (1995).

[9] Zhang, Dengsheng, and Guojun Lu. Review of shape representation and description techniques. *Pattern recognition* 37.1 (2004): 1-19.

[10] Traina, Agma JM, et al. Content-based image retrieval using approximate shape of objects. *Proceedings. 17th IEEE Symposium on Computer-Based Medical Systems*. IEEE, (2004).

[11] Celebi, M. Emre, and Y. Alp Aslandogan. A comparative study of three moment-based shape descriptors. *International Conference on Information Technology: Coding and Computing (ITCC'05)-Volume II*. Vol. 1. IEEE, (2005).

[12] Zhang, Dengsheng, and Guojun Lu. Content-based shape retrieval using different shape descriptors: A comparative study. *IEEE International Conference on Multimedia and Expo, 2001. ICME 2001*. IEEE Computer Society, (2001).

[13] Li, Shan, Moon-Chuen Lee, and Chi-Man Pun. Complex Zernike moments features for shape-based image retrieval. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 39.1 (2008): 227-237.

[14] Brisaboa, Nieves R., et al. Similarity search using sparse pivots for efficient multimedia information retrieval. *Eighth IEEE International Symposium on Multimedia (ISM'06)*. IEEE, (2006).

[15] Aronovich, Lior, and Israel Spiegler. CM-tree: A dynamic clustered index for similarity search in metric databases. *Data & Knowledge Engineering* 63.3 (2007): 919-946.

[16] Almeida, Jurandy, Ricardo da S. Torres, and Neucimar J. Leite. BP-tree: An efficient index for similarity search in high-dimensional metric spaces. *Proceedings of the 19th ACM international conference on Information and knowledge management*. (2010).

[17] Novak, David, Michal Batko, and Pavel Zezula. Metric index: An efficient and scalable solution for precise and approximate similarity search. *Information Systems* 36.4 (2011): 721-733.

[18] Britos, Luis, A. Marcela Printista, and Nora Reyes. DSACL+-tree: A dynamic data structure for similarity search in secondary memory. *International Conference on Similarity Search and Applications*. Springer, Berlin, Heidelberg, (2012).

[19] Fukushima, Kunihiko. Neocognitron: A hierarchical neural network capable of visual pattern recognition. *Neural networks* 1.2 (1988): 119-130.

[20] LeCun, Yann, et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* 86.11 (1998): 2278-2324.

[21] Hinton, Geoffrey E., Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation* 18.7 (2006): 1527-1554.

[22] Nair, Vinod, and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. *Icml*. (2010).

[23] Krizhevsky, Alex, Sutskever, Ilya and Geoffrey E. Hinton. ImageNet classification with deep convolutional

neural networks. In Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1 (NIPS'12). Curran Associates Inc., Red Hook, NY, USA, (2012): 1097–1105.

[24] Simonyan, Karen, and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[25] Huang, Gao, et al. Densely connected convolutional networks. *Proceedings of the IEEE conference on computer vision and pattern recognition*. (2017).

[26] Szegedy, Christian, et al. Going Deeper with Convolutions. *Proceedings of the IEEE conference on computer vision and pattern recognition*. (2015).

[27] Szegedy, Christian, et al. Inception-v4, Inception-Resnet and the Impact of Residual Connections on Learning. *Thirty-first AAAI conference on artificial intelligence*. (2017).

[28] Szegedy, Christian, et al. Rethinking the Inception Architecture for Computer Vision. *Proceedings of the IEEE conference on computer vision and pattern recognition*. (2016).

[29] He, Kaiming, et al. Deep Residual Learning for Image Recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*. (2016).

[30] Fierro, Atoany N., et al. Redes Convolucionales Siamesas y Tripletas para la Recuperación de Imágenes Similares en Contenido. *Información tecnológica* 30.6 (2019): 243-254.

[31] Melekhov, Iaroslav, Juho Kannala, and Esa Rahtu. Siamese network features for image matching. *2016 23rd international conference on pattern recognition (ICPR)*. IEEE, (2016).

[32] Dong, Xingping, and Jianbing Shen. Triplet loss in siamese network for object tracking. *Proceedings of the European conference on computer vision (ECCV)*. (2018).

[33] Hermans, Alexander, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737* (2017).

[34] Wang, Yaqing, et al. Generalizing from a few examples: A survey on few-shot learning. *ACM computing surveys (csur)* 53.3 (2020): 1-34.

[35] Lake, Brenden, et al. One shot learning of simple visual concepts. *Proceedings of the annual meeting of the cognitive science society*. Vol. 33. No. 33. (2011).

[36] Pan, Sinno Jialin, and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering* 22.10 (2009): 1345-1359.

[37] Storkey, Amos. When training and test sets are different: characterizing learning transfer. *Dataset shift in machine learning* 30 (2009): 3-28.

[38] Pan, Sinno Jialin, and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering* 22.10 (2009): 1345-1359.

[39] Kolesnikov, Alexander, et al. Big transfer (bit): General visual representation learning. *European conference on computer vision*. Springer, Cham, (2020).

[40] Yu, Haizi, et al. Learning from One and Only One Shot. *arXiv preprint arXiv:2201.08815* (2022).

[41] Swathi, P., Ravi Kumar, S., & Ajay Kumar, Y. L. Implementation of Logo Matching and Recognition System using Context Dependent Similarity Based on Interest Points. International Journal of Scientific Engineering and Technology Research, 3(32), 6441-6447, (2014).

[42] Bejinariu, S. I., Costin, M., Ciobanu, A., & Cojocaru, S. Similarities Identification in Logo Images. Proceedings of the International Conference on Intelligent Information Systems (IIS'2013), Chisinau, Republic of Moldova, (2013).

[43] Pinjarkar, L., Agrawal, P., & Kaur, G. Content-based Image Retrieval for Color Logo Images using Deep Learning Model. European Chemical Bulletin, 12(10), 263-274, (2023).

[44] L. N. Rani and Y. Yuhandri, Similarity Measurement on Logo Image Using CBIR (Content Base Image Retrieval) and CNN ResNet-18 Architecture, International Conference on Computer Science, Information Technology and Engineering (ICCoSITE), Jakarta, Indonesia, 2023, pp. 228-233, (2023).