TRAINING DEEP RETRIEVAL MODELS WITH NOISY DATASETS

Presented by

TOMÁS MARTÍNEZ CORTÉS

in partial fulfillment for the award of the degree of
DOCTOR IN MULTIMEDIA AND COMMUNICATIONS

UNIVERSIDAD CARLOS III DE MADRID

Tutor and advisor:
DR. IVÁN GONZÁLEZ DÍAZ

Leganés, February 2021

AGRADECIMIENTOS

Quiero agradecer especialmente a Fernando Díaz-de-María toda la confianza que ha puesto en mí, así como su ayuda, consejos y colaboración para sacar adelante esta tesis. No solo en este trabajo, sino durante toda mi trayectoria profesional en la universidad que no ha podido ser más agradable en lo personal, y satisfactoria en lo profesional.

Extiendo este reconocimiento a mi maestro y tutor, Iván González Díaz, por enseñarme durante todos estos años a buscar la excelencia, así como por su sinceridad y honestidad.

En el plano personal, agradezco a Laura su cariño y apoyo para que pudiera dedicarle el esfuerzo necesario a este trabajo, y a mis hijos por el tiempo de juego que les robé para poder terminar este documento.

Tomás Martínez

PUBLISHED AND SUBMITTED CONTENT

Parts from the following papers haven been included or extended in this thesis:

1. Martínez-Cortés, T., González Díaz, I., and Díaz-de-María, F. (2018). Automatic Learning of Image Representations Combining Content and Metadata. In International Conference on Image Processing (ICIP) (pp. 1972-1976).
   https://doi.org/10.1109/ICIP.2018.8451566

2. Martínez-Cortés, T., González Díaz, I., and Díaz-de-María, F. "Training Deep Retrieval Models with Noisy Datasets: Bag Exponential Loss". Pattern Recognition, p. 107811, 2021.
   https://doi.org/10.1016/j.patcog.2020.107811

OTHER RESEARCH MERITS

The following papers were part of my research but are not treated in this manuscript:

1. Tomás Martínez-Cortés, Miguel Ángel Fernández-Torres, Amaya Jiménez-Moreno, Iván González-Díaz, Fernando Díaz-de-María, Juan Adán Guzmán-De-Villoria, and Pilar Fernández (2014). A Bayesian model for brain tumor classification using clinical-based features. In 2014 IEEE International Conference on Image Processing (ICIP). IEEE. https://doi.org/10.1109/icip.2014.7025562

2. Iván González-Díaz, Tomás Martínez-Cortés, Ascensión Gallardo-Antolín, and Fernando Díaz-de-María (2015). Temporal segmentation and keyframe selection methods for user-generated video search-based annotation. Expert Systems with Applications, 42(1), 488502. https://doi.org/10.1016/j.eswa.2014.08.001

3. Guzmán, B. G., Cortés, T. M., López, A. R., and Armada, A. G. (2017). Design of a communication, vision and sensory system for a rescuer robot in coal mine areas. 2017 International Conference on Wireless Networks and Mobile Communications (WINCOM). https://doi.org/10.1109/wincom.2017.8238150

4. Eduardo Pla-Sacristán, Iván González-Díaz, Tomás Martínez-Cortés, and Fernando Díaz-de-María (2019). Finding landmarks within settled areas using hierarchical density-based clustering and meta-data from publicly available images. Expert Systems with Applications, 123, 315327. https://doi.org/10.1016/j.eswa.2019.01.046

RESUMEN EXTENDIDO

Este resumen contiene una descripción de los aspectos más relevantes de la presente tesis doctoral. Comienza motivando el trabajo desarrollado especificando los objetivos fundamentales e incluyendo una breve exposición de las contribuciones originales. A continuación, presenta las principales conclusiones alcanzadas durante la investigación y traza una seria de líneas de trabajo futuras relacionadas con las contribuciones generadas.

Motivación de la tesis

La visión artificial es una disciplina científica que trata de generar máquinas con capacidad de comprender el mundo visual de una forma similar a como lo hacemos los seres humanos. Millones de años de evolución han convertido la tarea de navegar y comprender nuestro entorno en sencilla pero, dado que no hemos conseguido esa habilidad mediante la implementación de complejos modelos matemáticos que la expliquen, realmente no entendemos cómo el cerebro la realiza. Esto es lo que convierte a la disciplina de la visión artificial en algo desafiante. Cuatro decadas tras los primeros intentos por atacar este problema [1] todavía somos incapaces de crear ingenios mecánicos con una capacidad visual similar a la nuestra.

En el mundo moderno, millones de imágenes son generadas y almacenadas a diario utilizando todo tipo de sensores. Existe una necesidad creciente de nuevos métodos basados en la visión artificial para indexar esos contenidos de forma que puedan ser navegados y recuperados de manera eficiente. Utilizar los nombres de los propios ficheros o descripciones textuales generadas por los usuarios, no es una alternativa satisfactoria para indexar las imágenes, debido a la cantidad de descripciones ausentes y erróneas que la anotación manual introduce. Otra aproximación a este problema consiste en utilizar herramientas automáticas basadas en visión artificial para generar descripciones de los archivos basadas en su contenido. Estas descripciones automáticas está pensadas para que una máquina pueda tomar decisiones basándose en ellas. Así, dada una imagen consulta o *query*, ésta se utiliza para localizar otros elementos relacionados en una base de datos. Este proceso es conocido como Recuperación de Imágenes Basada en Contenido o CBIR por sus siglas en inglés (*Content-Based Image Retrieval*).

Hoy en día, las herramientas software más utilizadas para generar descripciones automáticas de los archivos multimedia basándose en su contenido son las Redes Neuronales Convolucionales (CNNs) [2]. Estos modelos profundos han revolucionado el campo de CBIR; sin embargo, presentan algunas limitaciones importantes que lastran su adopción para atacar ciertos escenarios. Con las arquitecturas actuales [3], cuanto más profundo sea el modelo mejores prestaciones pueden alcanzarse. Sin embargo, a medida que aumenta el tamaño

de la red, también lo hace la cantidad de ejemplos necesarios en la base de datos para entrenarla y explotar todo su potencial. Estas grandes bases de datos, que típicamente incluyen millones de muestras, no deben contener errores en el etiquetado de las imágenes. En trabajos previos se ha demostrado cómo pequeños porcentajes de ruido en las anotaciones dañan seriamente la capacidad de las redes utilizadas para CBIR [4]. Por ello, las CNNs necesitan grandes bases de datos de entrenamiento libres de errores. Éstas son difíciles de conseguir en la práctica, porque las personas cometen errores al etiquetar debido a la fatiga y las distracciones. Además, el esfuerzo humano necesario para anotar tales bases de datos se ha vuelto prohibitivo a medida que éstas crecían de tamaño. Para empeorar la situación, algunos estudios han demostrado que el tipo de objetos contenidos en las muestras de entrenamiento deben ser los mismos que se esperan en test para conseguir un rendimiento óptimo [5]. Es por ello que se necesita una nueva base de datos de gran tamaño y libre de errores cada vez que cambia el dominio de la aplicación CBIR para la cual queremos explotar una CNNs. Para enfrentarse a esta problemática algunos autores han propuesto herramientas automáticas que permitan generar grandes bases datos de entrenamiento [6]. Desafortunadamente, los métodos automáticos son complejos y están adaptados al tipo de objetos presentes en las imágenes, lo que limita su rápida adopción en nuevos escenarios, reduciendo su aplicabilidad. Por otro lado, las aproximaciones automáticas introducen sesgos en las bases de datos en función del procedimiento empleado, lo que daña las prestaciones del sistema final.

Objectivos y contribuciones originales

Para incrementar la aplicabilidad de los modelos convoluciones profundos a nuevos problemas de CBIR, en esta tesis proponemos dos funciones de coste que permiten entrenar dichas redes utilizando bases de datos ruidosas:

1. Una función de coste que emplea etiquetas blandas y permite especializar CNNs generalistas, adaptándolas a diferentes dominios, utilizando bases de datos ruidosas. Estos conjuntos ruidosos son más fáciles de obtener sin intervención humana, lo que aumenta la aplicabilidad de estas redes para tareas de CBIR. En particular, la función propuesta emplea conjuntamente contenido y meta datos de las imágenes para inferir etiquetas blandas que se adaptan mejor al ruido, permitiendo hacer un ajuste fino de los modelos generalistas. Para ilustrar un caso de uso de nuestra función y medir sus prestaciones, se ha realizado una batería de experimentos que demuestran las posibilidades de nuestro método cuando este se aplica a la tarea de descubrimiento automático de lugares de interés.

2. Una segunda función de coste capaz de manejar la presencia de anota-

ciones ruidosas basándose únicamente en el contenido de las imágenes. En particular, proponemos una función exponencial novedosa basada en bolsas de muestras que se inspira en las técnicas de aprendizaje múltiple o MIL por sus siglas en inglés (*Multiple Instance Learning*). El fundamento del método es pesar de manera dinámica la contribución de cada instancia dentro de las bolsas. con el objetivo de que los ejemplos mal etiquetados pierdan su influencia nociva a la hora de ajustar el modelo. Los experimentos realizados muestran la superioridad de nuestra aproximación con respecto a otros métodos del estado del arte. Además, estos resultados permiten adoptar un nuevo paradigma a la hora de enfrentar problemas de CBIR: permitir que las CNNs aprendan los mejores patrones visuales para resolver un problema, al mismo tiempo que deciden de qué muestras aprenderlos.

Conclusiones

En este trabajo nos hemos centrado en el problema de la recuperación de imágenes basada en contenido. Se han presentado dos funciones de coste para entrenar redes convolucionales profundas que generan descriptores globales adecuados para este problema. La principal novedad de nuestras propuestas con respecto a las existentes en la literatura, es que nuestras aproximaciones son capaces de manejar el ruido en las bases de datos de manera explícita, en contraste con los métodos habituales que filtran el ruido antes de comentar el entrenamiento mediante complejas técnicas adaptadas a la naturaleza de las imágenes. Tener la capacidad de entrenar con bases de datos ruidosas es preferible a un preprocesado por dos razones fundamentales: 1) se evita el sesgo introducido por el método automático utilizado para eliminar el ruido del conjunto de entrenamiento; 2) permite generar soluciones para CBIR basadas en CNNs de manera mucho más ágil, ya que no es necesario diseñar una compleja etapa previa de filtrado que permita eliminar el ruido antes de poder entrenar el modelo.

En el Capítulo 3 de la tesis se presenta una función de correspondencia blanda (*Soft-Matching (SM) loss)* capaz de adaptar CNNs generalistas a nuevos dominios utilizando bases de datos de entrenamiento ruidosas. En concreto, la función de pérdida explota el propio contenido de las imágenes junto con meta datos asociados a éstas, para inferir etiquetas blandas que se adaptan mejor al ruido que las etiquetas tradicionales binarias. Esto permite llevar a cabo la especialización de los modelos. El método se presenta utilizando un tipo particular de meta datos, coordenadas GPS; sin embargo, la filosofía empleada puede ser extensible a otro tipo de aplicaciones y tipos de meta datos como se discute en la Sección 3.2.4 de este manuscrito. En el Capítulo 4 se demustra que nuestra función es útil para realizar la especialización de modelos generalistas, aunque presenta dos importantes limitaciones: 1) la función propuesta necesita

algún tipo de supervisión blanda para generar las etiquetas necesarias para el entrenamiento; 2) las etiquetas se fijan antes de comenzar a entrenar el modelo, basándose en los descriptores globales producidos por el modelo generalista, lo que condiciona las prestaciones que puede alcanzar la red especializada.

El Capítulo 4 explora los efectos de especializar CNNs utilizado la función de coste SM propuesta en el Capítulo 3. Para tal propósito, se diseña un sistema de descubrimiento automático de lugares de interés y las prestaciones del modelo generalista se comparan contras las de los modelos especializados utilizando la función de coste propuesta. Los experimentos muestran que nuestras redes consiguen mejorar hasta en un 55% las prestaciones de los modelos generalistas en la tarea. Esto implica que las redes profundas consiguen aprender las pistas visuales y peculiaridades de la región para la cual han sido entrenadas, generando descriptores de imágenes que están mejor adaptados a la localización. Además, para aquellos lugares de interés que no estaban presentes en las bases de datos, o incluso otras ciudades distintas, nuestros modelos retienen las prestaciones de las redes originales de las que derivan, lo que indica una buena resiliencia al sobre ajuste.

La segunda contribución de la tesis es la función de pérdida exponencial en bolsa (*Bag Exponential (BE) Loss*) que se presenta en el Capítulo 5. Esta función está inspirada en el aprendizaje multi-instancia (MIL) y trabaja con bolsas de pares de imágenes en lugar de pares aislados. Esto permite pesar la relevancia de cada muestra de entrenamiento dinámicamente, a medida que progresa el modelo. La función BE aumenta la aplicabilidad de las CNNs a problemas de CBIR, dado que el proceso de limpieza de las bases de datos de entrenamiento es uno de los más intensivos en horas de trabajo, y nuestro sistema lo convierte en innecesario. Además, al permitir que sean los propios modelos profundos los que manejen el ruido, eliminamos los sesgos introducidos por los algoritmos de filtrado. Del mismo modo que las CNNs aprenden los mejores patrones visuales para resolver una tarea específica, proponemos este mecanismo que las habilita también para decidir de qué muestras aprenderlos para optimizar los resultados.

El Capítulo 6 explora los efectos de entrenar CNNs para CBIR utilizando la función BE presentada en el Capítulo 5. Los resultados experimentales nos permiten extraer las siguientes conclusiones: 1) la función BE es más robusta al ruido en las bases de datos de entrenamiento que las alternativas presentes en la literatura; 2) la formulación que define nuestra función de coste es lo suficientemente general para ser aplicada con otros propósitos diferentes a la lucha contra el ruido. Por ejemplo, puede ser utilizada para incrementar la influencia de las muestras más difíciles; 3) la manera más efectiva y rápida de desplegar una CNN para CBIR en un nuevo dominio es emplear la función de coste BE sobre una base de datos ruidosa; 4) la función propuesta mejora las prestaciones del estado del arte al permitir al modelo elegir al mismo tiempo

no sólo los mejores patrones visuales para resolver la tarea, sino también las muestras de las cuales aprenderlos.

Líneas futuras de trabajo

Tras analizar las principales contribuciones y conclusiones del trabajo presentado, a continuación se exponen dos líneas futuras de investigación de especial relevancia.

En primer lugar, la función de coste SM propuesta en la primera parte de la tesis ha sido utilizada para especializar CNNs a ciudades o regiones particulares. Los modelos adquieren esta adaptación a la región, encontrando patrones arquitectónicos específicos que diferencian a cada objeto de interés (un edificio, estatua, fuente, etc,) del resto de objetos presentes en la zona. En otras palabras, los modelos están incentivados para ignorar aquellas características visuales que son comunes a todos los objectos de una región. Por tanto, una extensión natural de este trabajo consiste en modificar la función de coste propuesta para que utilice otras fuentes de meta datos distintas, y pueda así adquirir esta habilidad en nuevos escenarios.

Una de las principales limitaciones de la función SM es la necesidad de contar con un modelo inicial general que permita fijar las etiquetas blandas antes de comenzar el entrenamiento. El rendimiento final de este método está de algún modo limitado por las capacidades del método general inicial. El desarrollo de nuevas técnicas que permitan actualizar las etiquetas a medida que avanza el entrenamiento constituye otro importante campo de trabajo futuro.

En segundo lugar, la función BE propuesta en la segunda parte de la tesis ha sido explotada para entrenar modelos profundos para CBIR bajo bases de datos ruidosas. La hipótesis bajo la que trabaja nuestro método es la siguiente: asumiendo un conjunto de entrenamiento con un cierto porcentaje de ruido en cada categoría, muestrear un único par de imágenes de una categoría concreta puede dar lugar a una correspondencia falsa, causando inestabilidad durante el entrenamiento. Sin embargo, si muestreamos una bolsa de pares suficientemente grande, aunque muchas muestras pueden ser ruidosas, debe haber al menos algunas imágenes relevantes y bien etiquetadas de las que aprender. La manera mediante la cual la función propuesta pesa la relevancia de cada par de imágenes es mediante una función de similitud basada en distancias euclídeas.

Una línea futura de trabajo interesante consiste en diseñar esquemas que generen estos pesos basados en otros criterios. Por ejemplo: se puede considerar la conectividad entre muestras, limites que saturen bajo similitudes muy grandes o pequeñas, o la inclusión de los falsos negativos en escenarios donde puedan ser un problema.

Otra línea interesante consiste en transformar la función BE de un método basado en pares a uno basado en listas. Las funciones de coste basadas en listas

están adquiriendo mucho protagonismo para la comunidad de la visión artificial, debido a que optimizan las redes utilizando una métrica más ajustada a la que se emplea finalmente para evaluar, lo que puede incrementar el rendimiento del sistema.

ABSTRACT

In this thesis we study loss functions that allow to train Convolutional Neural Networks (CNNs) under noisy datasets for the particular task of Content-Based Image Retrieval (CBIR). In particular, we propose two novel losses to fit models that generate global image representations. First, a Soft-Matching (SM) loss, exploiting both image content and meta data, is used to specialized general CNNs to particular cities or regions using weakly annotated datasets. Second, a Bag Exponential (BE) loss inspired by the Multiple Instance Learning (MIL) framework is employed to train CNNs for CBIR under noisy datasets.

The first part of the thesis introduces a novel training framework that, relying on image content and meta data, learns location-adapted deep models that provide fine-tuned image descriptors for specific visual contents. Our networks, which start from a baseline model originally learned for a different task, are specialized using a custom pairwise loss function, our proposed SM loss, that uses weak labels based on image content and meta data.

The experimental results show that the proposed location-adapted CNNs achieve an improvement of up to a 55% over the baseline networks on a landmark discovery task. This implies that the models successfully learn the visual clues and peculiarities of the region for which they are trained, and generate image descriptors that are better location-adapted. In addition, for those landmarks that are not present on the training set or even other cities, our proposed models perform at least as well as the baseline network, which indicates a good resilience against overfitting.

The second part of the thesis introduces the BE Loss function to train CNNs for image retrieval borrowing inspiration from the MIL framework. The loss combines the use of an exponential function acting as a soft margin, and a MIL-based mechanism working with bags of positive and negative pairs of images. The method allows to train deep retrieval networks under noisy datasets, by weighing the influence of the different samples at loss level, which increases the performance of the generated global descriptors. The rationale behind the improvement is that we are handling noise in an end-to-end manner and, therefore, avoiding its negative influence as well as the unintentional biases due to fixed pre-processing cleaning procedures. In addition, our method is general enough to suit other scenarios requiring different weights for the training instances (e.g. boosting the influence of hard positives during training). The proposed bag exponential function can bee seen as a back door to guide the learning process according to a certain objective in a end-to-end manner, allowing the model to approach such an objective smoothly and progressively.

Our results show that our loss allows CNN-based retrieval systems to be trained with noisy training sets and achieve state-of-the-art performance. Furthermore, we have found that it is better to use training sets that are highly correlated with the final task, even if they are noisy, than training with a clean

set that is only weakly related with the topic at hand. From our point of view, this result represents a big leap in the applicability of retrieval systems and help to reduce the effort needed to set-up new CBIR applications: e.g. by allowing a fast automatic generation of noisy training datasets and then using our bag exponential loss to deal with noise. Moreover, we also consider that this result opens a new line of research for CNN-based image retrieval: let the models decide not only on the best features to solve the task but also on the most relevant samples to do it.

Contents

List of Figures

List of Tables

Acronysm

AP	Average Precision. 23
BE	Bag Exponential. 8
BoW	Bag of Visual Words. 15
CBIR	Content-Based Image Retrieval. 1
CNNs	Convolutional Neural Networks. 4
CT	Contrastive. 36
DML	Distance Metric Learning. 28
FGIR	Fine-Grained Image Retrieval. 22
FV	Fisher Vectors. 17
GL	Google landmarks. 59
GMM	Gaussian Mixture Model. 17
GPS	Global Positioning System. 23
IR	Image Retrieval. 2
L	Landmarks. 59
mAPq	Quantized mAP. 61
MIL	Multiple Instance Learning. 23
MS	Multi-Similarity. 61
SfM	Retrieval SfM-120k. 59
SIFT	Scale-Invariant Feature Transform. 13
SM	Soft-Matching. 8
T1KP	Top 1000 Paintings. 59
TL	Triplet. 36
VLAD	Vectors of Locally Aggregated Descriptors. 17

Chapter 1

Introduction and Thesis Objectives

This introductory chapter is intended to give the reader a quick overview of the work presented on this dissertation. Section 1.1 presents the main motivations that guide our contributions in three steps. First, Subsection 1.1.1 introduces the main field of this thesis: Content-Based Image Retrieval (CBIR), making special emphasis on the nature of the task, its principal applications, strengths, and weaknesses. Second, Subsection 1.1.2 discusses the main factors that influence the performance of CBIR systems, as well as the main tools that are applied today to tackle the problem. Third, Subsection 1.1.3 focuses on the current limitations of CBIR systems, how they are tackled in the literature, and to what extent current solutions address them which, ultimately, motivates the work presented on this thesis. Section 1.2 is devoted to main objectives of this thesis. It introduces our approach to improve current CBIR limitations and points out the main differences with respect to current solutions in the literature. Then, the section gives a brief description of the two main objectives for our work. Finally, Section 1.3 contains a detailed description of the structure of this document.

1.1 Motivation

1.1.1 From Image Retrieval (IR) to Content-Based Image Retrieval (CBIR)

Millions of new images are created and stored everyday around the world. There is a pressing need for new methods to index them so they can be efficiently searched and retrieved. The most immediate source of information available to tackle this problem is the textual context surrounding the images. For instance, the original file name given by the authors can offer clues about the content of the picture. Image meta data is another piece of information that can be exploited as well. Some online platforms allow users to add textual descriptions

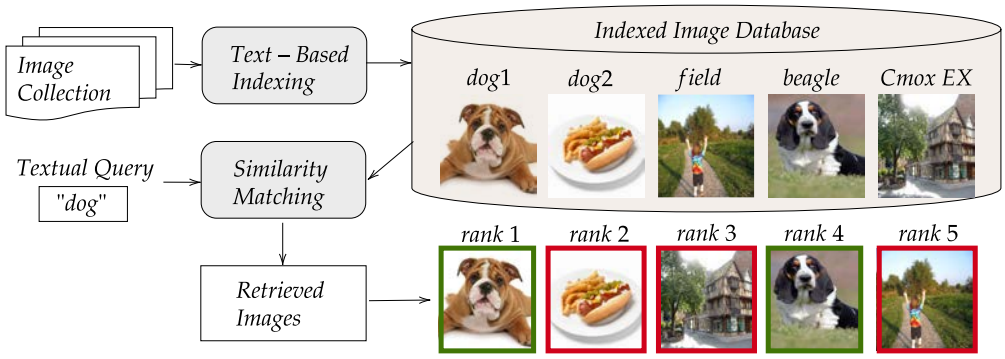


Figure 1.1: A text-based Image Retrieval (IR) system. Green and red rectangles indicate relevant and irrelevant images with respect to the query respectively.

to images that can be relevant. In the case of images inserted on web pages, an analysis of the surrounding text can give insightful clues to index the pictures. The approach of exploiting text for indexing and retrieving visual content was the first taken in the 80s, back then, this practice was referred to as Image Retrieval (IR). Figure 1.1 contains an schematic illustration of an early image retrieval system based on text. First, a group of images along their associated textual descriptions are collected. Then, those descriptions are used to generate an indexed image database. Later, a user inputs a textual query which is compared against the indices on the database using some text-based measure of similarity. Finally, a list of ranked images ordered by relevance with respect to the query is returned.

There are several problems that limit the success of image retrieval systems based on text. The first is that the user needs to formulate a textual query, which implies that he must be able to verbalize the kind of content that he expects to retrieve. This can be difficult for several reasons. For instance, a user might want to retrieve examples from a specific breed of dog he can visually recognize (beagle in Figure 1.1), but ignore the specific breed name. Thus, a generalization is used as query (”dog”), and the relevant samples for the user are not included in the top ranking positions. Another source of difficulties is the heterogeneity and polysemy of the language itself. For instance, dogs in the database can be indexed by: dog, doggy, pooch, hound, canine, etc. This forces the similarity matching mechanism to include a language model which is a challenging task in itself. Beyond the complex nature of language, human generated textual descriptions can be wrong or missing. Wrong labels make irrelevant instances to be returned and vice versa, while missing information is often superseded by some form of automatic generated meta data, like camera model, that rarely relates to the visual content of the pictures (*”Cmox EX”* in Figure 1.1).

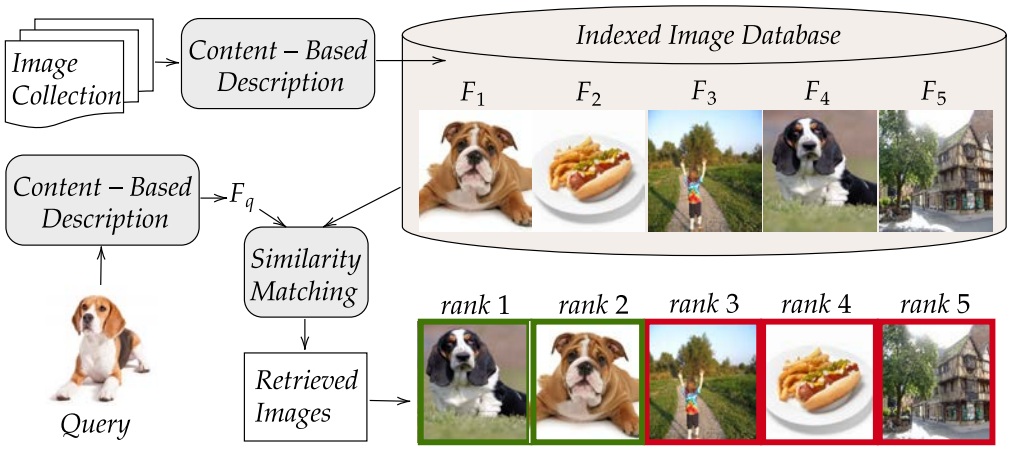


Figure 1.2: A Content-Based Image Retrieval (CBIR) system. F_i is the automatic content-based representation of image I_i in the database. F_q is the representation for the query. Green and red rectangles indicate relevant and irrelevant images with respect to the query respectively.

To address the problems of the first image retrieval systems, the community took a different but challenging approach: to employ computer vision algorithms to describe the images based on their visual content. Figure 1.2 illustrates this approach. A machine *looks at* a collection of images and generates some form of convenient textual or numerical description (F_i), that is later employed as a key to index the pictures on database. The same process is done for a query image (F_q), and the resulting content-based representations are used by a similarity function to rank the database instances based on their relevance with respect to the query. This whole process is commonly referred to as Content Based Image Retrieval (CBIR). Computing descriptions for the images based on their content mitigates the problems of classical text-based IR systems. In CBIR, it is possible to do *queries-by-example*, thus, the user no longer needs to be able to verbally articulate with precision the kind of objects that wants to retrieve. The difficulties related to the language are also avoided, since the descriptions can be numerical vectors instead of words. Finally, there are no missing descriptions since they are automatically generated by an algorithm.

Nowadays, CBIR systems are used in many different applications. For instance, in automatic face recognition, a collection of pictures for which the names of the subjects appearing on them are known is compared against a query image containing the face of an unknown individual. The labeled photographs are ranked based on their similarity with respect to the query, and the identity of the person is inferred considering the most similar images from the database. A similar process can be employed to recognize car make and models, artworks, landmarks, and in general, any visual concept that benefits from transferring

meta data between related images [13]. Beyond using images as queries, it is also possible to automatically generate textual content-based descriptions, which allow a later search and retrieval process using textual queries. For instance, a system to automatically generate image captions describing the images in a human-like way can be employed to index them on a database [14]. Later, a textual query can be compared against the generated captions to find relevant images.

1.1.2 Convolutional Neural Networks (CNNs) as the Dominant Paradigm in CBIR

The key for the success of CBIR applications lies in their ability to represent the content of the images in a way that a mathematical function can easily measure the relevance of each dataset picture with respect to the query. Nowadays, the most common form of representation for images is a numerical vector, usually referred to as *feature vector* or *visual descriptor*, and the most common function to assess the relevance is some sort of distance between feature representations.

In the early days of CBIR, image representations were computed based on global color, shape or texture features, i.e., taking into account all image pixels. That strategy worked for isolated objects in controlled scenarios but suffered when facing more open-world applications with high levels of cluttering and occlusions. This limitation gave rise to the development of local descriptors, where images are represented by a set of feature vectors, each of them computed over a local image neighborhood. These methods are hand-crafted algorithms where the designer is in charge of deciding what kind of information is relevant and will be contained in the feature.

The hand-crafted features paradigm shifted in 2012 when Convolutional Neural Networks (CNNs) were brought back to stage in the work presented in [2]. There, a deep network called AlexNet surpassed the state-of-the-art performance in an image classification task by a large margin. In essence, CNNs are hierarchical layered structures where simple abstract visual patterns from previous layers are combined to form increasingly complex and specific ones. As an example, consider the visual patterns contained in the filters of the CNN specialized on human faces depicted on Figure 1.3. The first layer gets as input the original images and searches for elementary patterns on them: blobs and edges. The second layer is applied over the outputs of the first one, thus, it looks for combinations of elementary elements that give rise to more specific visual patterns related to human faces: eyes, mouths, eyebrows, etc. Finally, the last layer combines the outputs from the second to detect complete human faces. The main difference between this approach taken by CNNs and traditional methods for face detection [15] is the lack of designers input to decide what visual patterns were relevant to detect the faces. Instead, the model inferred them from

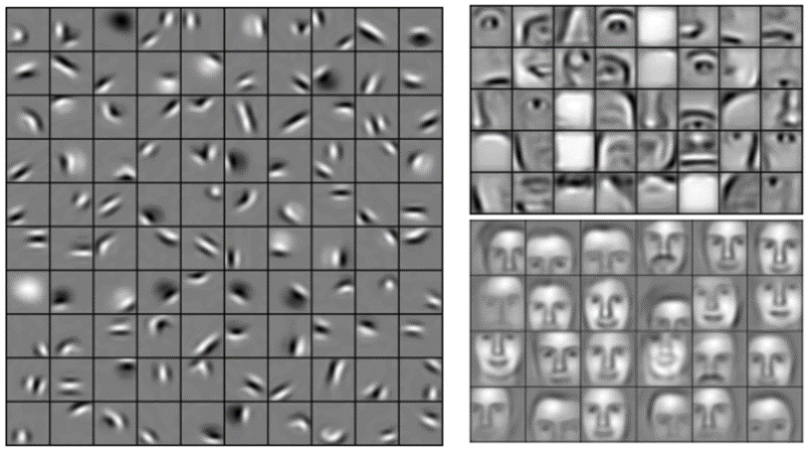


Figure 1.3: Visual patterns contained in the kernels of the first layer (left), second layer (top-right) and third layer (down-right), of a CNN specialized in human faces. Taken from [7].

data.

In this thesis, we employ CNNs to generate the image descriptors for CBIR. CNNs learn how to extract useful features for a particular application by means of an optimization target called the loss. The loss is a differentiable mathematical function that receives as input the image descriptors computed by the CNN and measures how much they deviate from achieving the goal of the task. The internal parameters of a CNN model are then adjusted such that the loss is minimized for a set of examples contained on a corpus known as the training set. Thus, CNNs are used to tackle CBIR problems by means of specific training sets and loss functions.

1.1.3 The Need of Large Correlated Training Datasets

Today, classical methods to learn image representations have been replaced by CNNs because of their superior performance. However, these deep networks also come with some limitations. With current model architectures, bigger networks usually yield better results [3]. However, using very deep models with thousands of filters and millions of parameters to adjust, requires the creation of huge labeled training sets to fully exploit their potential. In addition to being large, training datasets need to be specific for the task at hand, which prevent us from exploiting the massive corpuses available for related fields like image classification [16]. But even if a great deal of human effort is directed towards the creation of a very large retrieval dataset to fit the models, the expected performance for the network is heavily dependent on the topic correlation between the training and test sets. To understand this phenomenon, let us consider the

network from Figure 1.3 and suppose that it has been build for a face recognition tasks. The model will excel when dealing with human faces, but the visual patterns learned for faces are not as useful to detect cars or trees. One can consider including other topics than faces on the training set, but then, for the same database size, there will be less human faces and the model will not be so proficient in that particular area of expertise. Building a dataset containing all possible objects is not feasible, and generating specific ones every time the domain changes is too labor intensive for being practical.

In the literature, this dilemma is tackled using different alternatives although none of them is ideal for several reasons. The most straightforward method to build a large correlated retrieval dataset consists in defining the topic of interest and use textual tags (or other metadata) in search engines to retrieve a collection of potentially suitable images. This is perfectly feasible and can generate a large database with limited human effort but, unfortunately, the resulting collection of pictures will very likely contain a considerable amount of noise (mostly due to labeling errors or imprecisions on the metadata). Indeed, the presence of noise in the training datasets is known to hinder the learning process of deep retrieval models [6]. The second, and more common approach, extends the previous one by introducing (semi)automatic algorithms to post-process the initial set of retrieved images, filtering out non-relevant samples and reducing the noise level. For instance, the authors in [17] employ invariant keypoint matching combined with spatial verification, to remove all instances of the database with no related images on their respective categories. In [4], an Structure From Motion (SfM) pipeline is employed to generate 3D building models using all available images. Those instances that do not fit well the produced models are discarded as noise. In [18], the associated geo-location of images is exploited to cluster all instances into training categories, that are later refined using the same keypoint matching with spatial verification strategy from [17].

The design of automatic methods to generate the necessary large and topic dependent datasets is not straightforward and requires extensive engineering work and innovation. Furthermore, these techniques need to manage a hard tradeoff: on the one hand, if the filtering is too restrictive, both the size of the database and its diversity will be dramatically reduced. Furthermore, the process will be particularly aggressive with those relevant samples that are less representative (e.g. hard positives). On the other hand, if the filtering is not restrictive enough, the resulting training set will still contain some degree of noise, which, even in small proportions, will significantly degrade the performance of current approaches [4]. A final concern regarding automatic filtering methods is that post-processing algorithms inevitably introduce a bias into the training dataset, which conditions the subsequent learning task. Such bias can limit the effectiveness of CNNs for retrieval to that of the method that was used to filter the noise. In the next section, we discuss how we avoid these limita-

tions imposed by the post-processing approaches, by designing loss functions capable of training under noisy datasets.

1.2 Thesis Objectives

In this thesis, we propose a paradigm shift from the use of complex pipelines to reduce the presence of noise in the retrieval datasets to the design of novel retrieval loss functions that can handle noise in a end-to-end manner. The benefit of our approach is two fold. On the one hand, it facilitates the generation of new topic-adapted datasets with minimal human intervention, because retrieving a large noisy collection of images is fast using modern search engines. Since the loss handles the noise internally, there is no longer a need for ad-hoc cleaning stages, and in consequence, our method enables a quick and broad adoption of CNNs for new retrieval scenarios. On the other hand, putting aside the noise filtering process avoids the biases that come with it. Hence, our deep networks are no longer conditioned by an external process and can decide the best visual patterns to solve the task, as well as the best samples to do it.

We have established the following objectives for this thesis:

1. The development of mechanisms to generate retrieval training sets without the need of human annotations or complex filtering pipelines. These methods should be based on image content and, optionally, image meta data.

2. Prove that it is possible to exploit noisy training sets to improve the performance of well established CNNs models, on some end-user applications such as landmark discovery.

3. Generate a novel CNN-based solution to the problem of CBIR that is robust to noise on the training sets.

4. Propose a new way to efficiently deploy retrieval CNNs for new domains with minimal human effort.

To fulfill the previously stated objectives, two novel noise-robust retrieval loss functions are presented on this dissertation:

1. A modified contrastive loss [19] that, making use of soft labels, is better suited to handle noise than its original counterpart that employs a binary categorization. In particular, image content and metadata are jointly exploited to infer the soft labels that allow to fine tune the models under noisy datasets. To illustrate a case of use, the designed loss is used to specialize general CNNs to better represent images from particular cities

or regions, boosting the system performance in a subsequent landmark discovery task.

2. A novel Bag Exponential loss capable of handling noise considering only image content. In particular, we propose a novel retrieval loss that, inspired by the Multiple Instance Learning framework, works with bags of matching images, and allows to dynamically weight the relevance of each sample as the training progresses. Experimental evidence show the superior performance of our loss with respect to the state-of-the-art, propose a novel way of facing new CBIR domains using noisy data, and paves the road towards a new interesting goal: let the models decide not only on the best features to solve the task, but also on the most relevant samples to do it.

1.3 Structure of the Document

This section presents a description of the contents present on the different chapters of this thesis.

Chapter 2 revisits the previous work related to the main contributions on this thesis. First, a historic revision of the CBIR field is presented. It starts by discussing the earlier systems and follows the subsequent trends that have given rise to the present era dominated by Convolutional Neural Networks. Then, it categorizes and discusses loss functions used to train deep retrieval models in modern systems, and traces similarities and differences with the ones proposed on this work. Finally, the chapter revisits the different methods employed in the literature to handle noise on the training datasets, and situates our own contributions with respect to them.

Chapter 3 describes the theoretical framework behind our first contribution, the Soft-Matching (SM) loss. First, an introduction presents the main problematic that leads to the design of the proposed loss function, and then, the intuitions and mathematics behind the method itself are explained. This chapter also includes a discussion about possible scenarios where our proposal can be exploited.

Chapter 4 contains experiments where the SM loss introduced on the previous chapter is used to adapt CNNs models to new topics, boosting their performance on a landmark discovery task. First, the chapter discusses and justifies the use of this particular domain of application for the experiments. Then, a description of the landmark discovery system itself is included. The final part of the chapter is devoted to describing the experimental setup and main results that lead to the final conclusions.

Chapter 5 describes the theory behind our second contribution, the Bag Exponential (BE) loss. The chapter begins with an introduction describing the

main motivations that leads to our proposal. Then, the mathematical framework behind our contribution is presented, as well as a discussion of some efficiency aspects of the method in comparison with others from the literature.

Chapter 6 presents a battery of experiments where the BE loss from the previous chapter is subjected to different tests and cases of use. It begins with an introduction that includes a description of our claims as well as the parts of the chapter where they are fulfilled. It continues with a description of the experimental setup. Then, a set of experiments are presented that measure the noise resilience of different losses from the literature with respect to our own. Also, an state-of-the art comparative as well as ablation studies are included. This chapter also includes an experiment proposing a novel approach to CBIR problems based on our method. The chapter ends outlining the main conclusions achieved through the experiments.

Chapter 7 closes this thesis by presenting the main conclusions achieved through this work and outlining future lines of research related to our contributions.

Chapter 2

Related Work

In this chapter we revisit literature related to Content-Based Image Retrieval, the field of computer vision that represents the focus of this thesis. Section 2.1 contains a historical review of the field, describing the thinking process and the most important contributions that have led to the current era dominated by Convolutional Neural Networks. Section 2.2 revisits modern CNN methods to generate content-based image descriptors for CBIR, with particular emphasis on loss functions. Finally, Section 2.3 explores previous deep learning literature that deals with the presence of noise in the training sets.

2.1 A Historical Overview of CBIR Systems

There is a vast amount of literature related to CBIR. It is so large and so varied with respect to each particular module of the complete retrieval pipeline that addressing all of it falls out of the scope of this thesis. Instead, we are going to focus on the generation of content-based image descriptors and the metrics used to measure their similarity. Thus, we will skip topics such as query definition and interaction with the user, efficient and compact encoding and storing of descriptors, taxonomies for semantic categorizations, different system architectures, and performance metrics, among others. The reader interested in a more exhaustive overview, including those and other topics, is referred to the works presented in [20] and [21].

2.1.1 The early CBIR systems (1990 - 1999)

Descriptors

Color [22] is a descriptor that has been commonly used for CBIR since the very early systems [23]. The reasoning behind using color to describe image content is that certain objects and concepts are related to predominant colors.

For instance, fire extinguishers are mainly red, trees usually green and so on. Thus, characterizing images based on color is a sensible choice. However, color often varies among object instances of the same class due to factors as position and spectrum of the illumination source, camera viewpoint, object material and surface orientation. For this reason, color descriptors are particularly useful for narrow scenarios where the image capturing conditions are under control. In broader scenarios, like open world imagery, using color shows more limitations. As an illustrative example, images a) and b) from Figure 2.1 share the exact same histogram even though they depict completely different objects, at the same time, images a) and d) have different histograms although both depict a house in a forest.

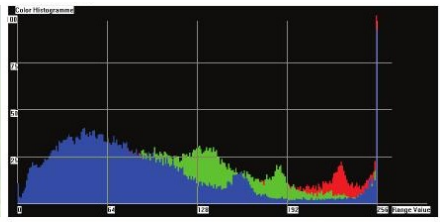
Shape [24] was another popular descriptor in the early days, probably because it was thought that humans frequently used it for solving the task of visual retrieval. However, the lack of precise and robust methods to exploit shape at the time, made that the problem was considered a hard challenge [25] and limited the usability of shape. Broadly speaking, two main variants existed for computing shape features: the global and the local. Both exploited some form of segmentation to generate binary masks containing objects shapes [26]. Global shape features considered the whole shape of the object and summarize its content by measuring its area, perimeter, eccentricity, etc [23], which gave place to a compact feature presentation. Local shape features focused on describing each point of the shapes separately [27], yielding 2D signatures for the images. This approach was able to better accommodate partial occlusions for the objects.

Texture [28] is another visual element that has thoroughly exploited since the beginning of CBIR. The first methods used for retrieval usually focused on measuring variants of contrast, directionality and coarseness [23]. Contrast can give a sense of how prominent a texture pattern is; directionality aims to find principal directions or how isotropic are the surfaces; and coarseness gives a sense of scale, i.e., fine grained versus coarse elements.

During the 90s, a great effort was made to improve the robustness of those initial features against common natural variation that occur on real world scenarios: changes in illumination, backgrounds, scale, cluttering, perspective and occlusions. By the end of the decade, there were available color [29], shape [30] and texture [31] features invariant to some of those transformations. However, cluttering and occlusions were still a major problem because most existing methods extracted *global descriptors*, i.e., features based on all image pixels. If an object of interest is small with respect to the background, or is occluded, only a small fraction of image pixels conveys useful information. A global descriptor dilutes the contribution from the relevant pixels limiting the discriminative power of the resulting feature. In this regard, the next decade introduced *local descriptors*, a major advance that boosted the performance and applicability of new CBIR systems.

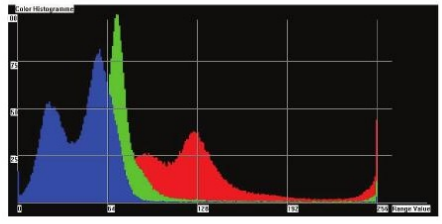
(a) House, forest, cabin.

(b) Tiger, head, grass.



(c) Color Histogram for (a) and (b)

(d) House, forest, cabin.



(e) Color Histogram for (d)

Figure 2.1: Images a) and b) have exactly the same histogram but are not related, while a) and d) have different histograms but are related. Taken from [8].

Similarity metrics

The most common approach to measure similarity between two images consisted in assuming that the descriptors were vectors in a feature space. Then, similarity was simply the inverse of the euclidean distance [23]. For histogram descriptors, mostly used for color, common choices were the histogram intersection distance [32] or the cumulative histogram distance [33]. Other methods tried to use the feature vectors to model similarities as a probabilistic concept using Bayesian analysis of images differences [34]. To address similarity between shapes, the first option was to extract some information from the segmentation masks, generate vectors, and employ any of the previous methods. However, there were also alternatives to compute similarity based on the full silhouettes [35].

2.1.2 From global to local descriptors (1999-2012)

Descriptors

Previous methods computed global descriptors that take into account all image pixels at the same time. Although this had proved to work for images containing isolated objects, it did not apply well to scenarios with clutter and partial occlusions. This was a known problem for the community and researchers tried to employ sliding windows to compute features for all image locations [36]. Nevertheless, this strategy presented additional issues: on the one hand, if features from all positions were finally aggregated, the dilution problem from the global

Figure 2.2: Keypoints detected for the same object under an image transformation using the Harris corner detector [9].

descriptors still appeared; on the other hand, if all individual descriptors were stored and compared, the computational cost dramatically increased and, besides, some method was needed to establish correspondences between patches of different images. To avoid the brute force approach of the sliding window, the focus was shifted toward strategies for selecting a small subset of interesting points, called *keypoints* [37], that defined where to compute *local descriptors*. Keypoints are image locations that are distinct and repeatable. Distinct means that they are easy to identify and distinguish from any other local regions, and repeatable means that they are invariant to image transformations, and can be detected as long as they appear in a scene. Figure 2.2 shows the keypoints detected using a popular method, the Harris corner detector [9], for the same object and different image transformations: illumination, rotation, view point, and a non-rigid transformation (neck and head).

Keypoints have been around the computer vision community since the 80s [39], although they were used in other tasks like stereo matching. The first work to exploit them in CBIR was [40], where the Harris corner detector [9] was employed to select a set of keypoints, and a multiscale and rotation-invariant descriptor was assigned to each local region. Since the amount of available keypoints was not fixed, each image was described by a different number of feature vectors. This first approach was soon followed by a very influential work known as Scale-Invariant Feature Transform (SIFT) [41] illustrated in Figure 2.3. SIFT introduced a new keypoint detector based on blobs, that is scale-invariant in contrast to the original Harris detector, and uses the scale information to choose the sizes of the local regions from where extracting the

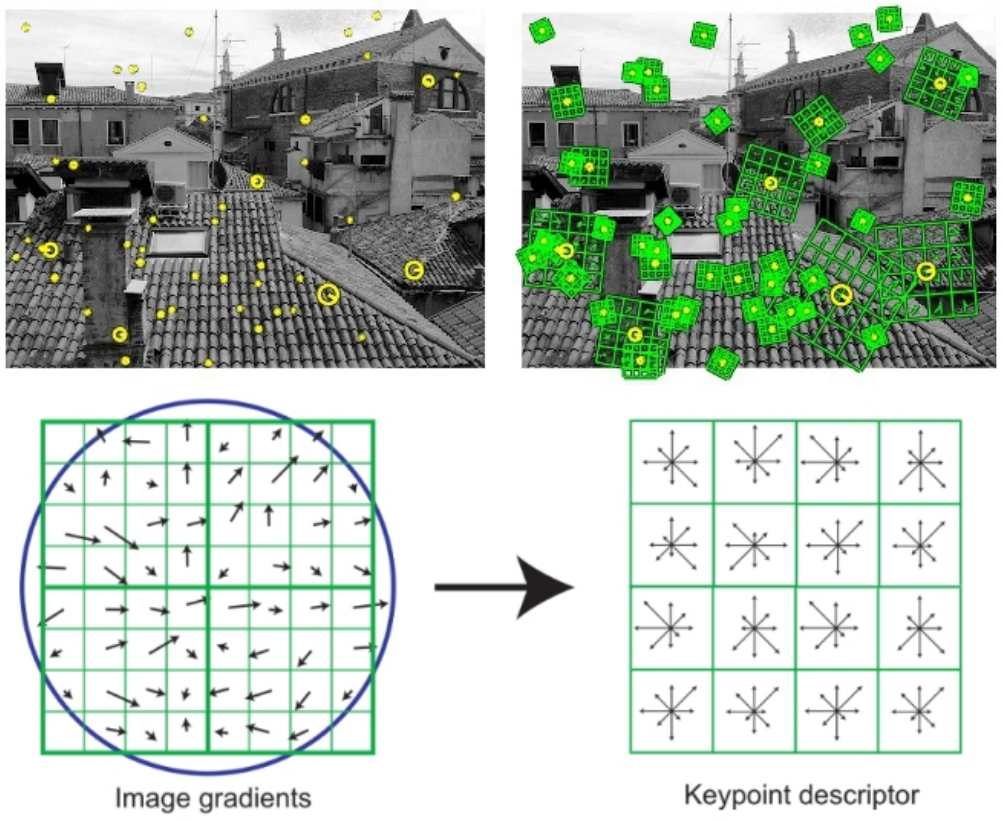


Figure 2.3: The process of computing local features using the SIFT method. First, a blobs detector is employed to find interest points (top left). Second, the scale of the blob is used to establish the size of a local neighborhood, from where dominant gradient orientations are computed on a grid (top right). Finally, gradient orientations are aggregated into histograms that describe the local region (down). Taken from [38].

descriptors. Besides, the method defines a canonical direction for the descriptors so that there is a common reference frame across images. The descriptor itself its a scale and rotation-invariant 128-dimensional vector, containing histograms of oriented gradients. One aspect that was later improved with respect to the original method, was its robustness to affine transformations. In [42], the authors proposed a new multi-scale Harris detector that brings affine invariance to the interest points. The affine information was exploited to compute SIFT descriptors over affine covariant local regions. Another common topic of the decade was reducing the computational complexity of SIFT. Faster keypoint detectors such as FAST [43], AGAST [44] and MSER [45], as well as binary descriptors such as ORB [46], BRIEF [47], and FREAK [48], made possible applications with real-time constrains [49].

The success and influence of the SIFT local descriptor for image retrieval have been staggering. Twenty years after it was proposed, is still found amongst the top performing methods (including CNNs) in several recent benchmarks comparing local descriptors [50, 51].

Similarity metrics

By using SIFT-like methods, an image is described by a set of local feature vectors, where the cardinality of the set is defined by the amount of keypoints found. The most simple way of measuring similarity between images, is to count the number of matches between their local features. A common strategy is to consider that two local features form a match if they are the closest neighbor to each other. However, when comparing two sets of vectors, there is always a closest neighbor, so filtering the candidates has been a prolific field of study. Particularly, methods based on global image transformations (affine homography with RANSAC) or local coherence (ordering, neighbors) have been proposed in the literature [41, 52, 42, 53, 54, 55, 56, 57, 58, 59].

The main drawback of measuring similarity using a nearest neighbor process followed by a filtering stage is its computational complexity. In 2003, the next milestone for image retrieval systems, the Bag of Visual Words (BoW) model, was introduced in [60]. The goal of the model was to perform image retrieval analogously to how text retrieval was done. In text retrieval, a document is represented by a vector containing the number of times each word from the dictionary occurs. Thus, to make image retrieval alike, an image is considered a document and a local feature plays the role of a text word. The main difficulty is that in text, the dictionary of possible words is well established by common language, but local features from images are real valued vectors with infinite possibilities. To circumvent this problem, the authors generated a dictionary of *visual words* by keeping the centroids resulting from clustering a large amount of local features. Then, each local feature from an image can be vector-quantized by association to its closest dictionary visual word. At the end, an image is represented by a BoW signature containing the amount of times each visual word from the dictionary appears on it. The full process is illustrated in Figure 2.4. In essence, this strategy is pre-computing the matches during the quantization process. For a new query, once its BoW signature is computed, the retrieval process is identical to the highly efficient text retrieval analogous. One important concern is that, since the matches between local patches are done using approximate quantized versions of the feature vectors, some performance might be lost. The authors in [60] proved that the possible lost in performance was compensated by the applicability of all the text retrieval existing tools that resulted to transfer well for images.

The BoW model was established as the most common tool for measuring

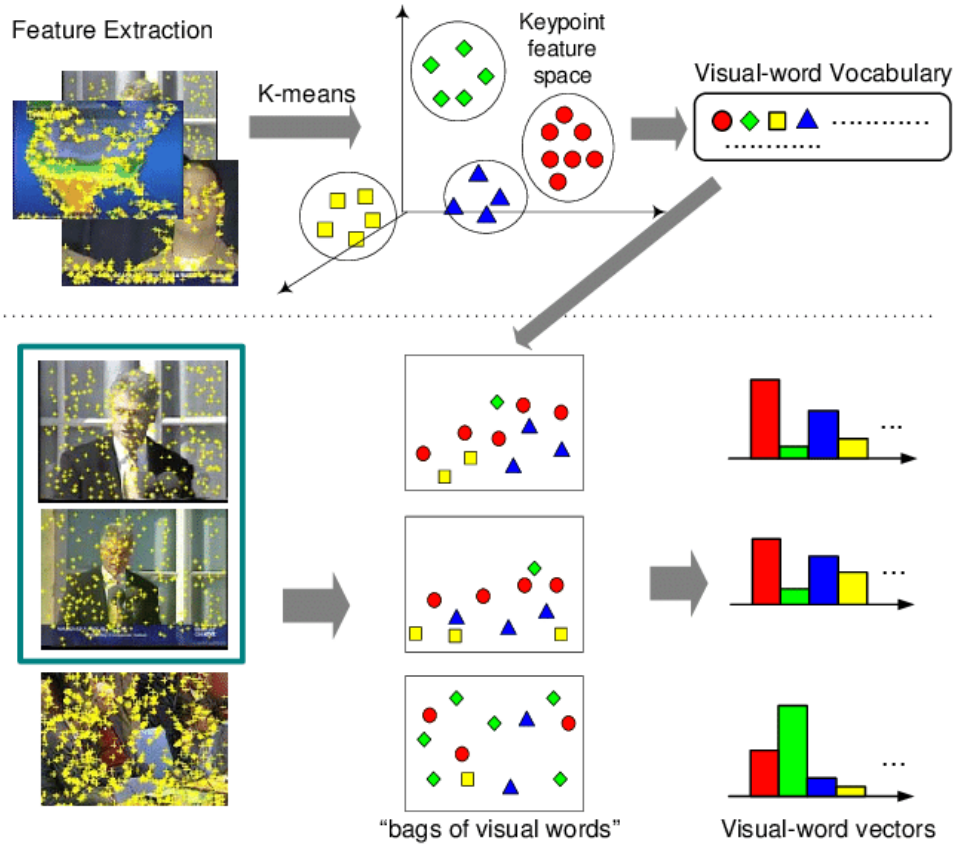


Figure 2.4: The Bags of Words (BoW) model for image matching. First, a collection of feature vectors are clusterized to establish the visual vocabulary. Then, feature from images can be assigned to the centroids and described based with a histogram of visual word occurrences. Taken from [10].

similarity between images for over a decade. Still, the negative effects of quantizing the features to assign them to visual words of the dictionary continued to be studied. There is a clear trade-off between using large visual dictionaries and the time it takes to associated each visual word to them. If the vocabulary is small, the association is fast but the quantization is poor and two different visual descriptors might be wrongly assigned to the same word. If the vocabulary is large, association is slow, but the quantized version of the feature vectors is more similar to the original. In two contemporary works [61, 62], the authors introduced methods for fast assignation of local features to dictionary words exploiting approximate and hierarchical clustering algorithms. This allowed the use of much larger visual dictionaries efficiently (up to several millions of words), which improved the performance and applicability of CBIR systems.

However, BoW signatures showed two main problems. First, with increasing vocabulary sizes, the signatures were growing very fast, harming the com-

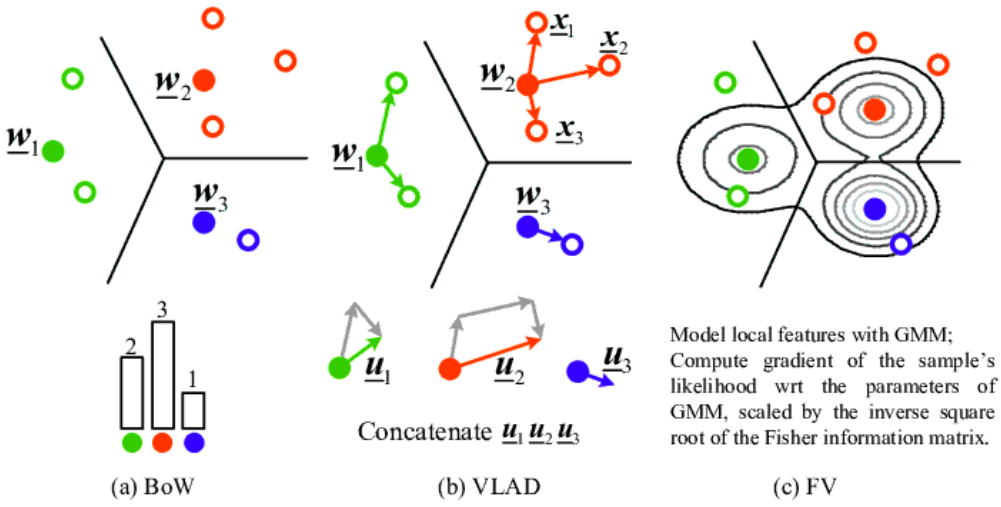


Figure 2.5: Comparison between the generation of image signatures using: a) Bag of Words (BoW); b) Vectors of Locally Aggregated Descriptors (VLAD); and c) Fisher Vectors (FV). Taken from [11].

putational efficiency. Second, the descriptors were hardly assigned to dictionary centroids without considering how far they were from them. In consequence, some important information was discarded that limited the performance of the method.

Two contemporary works addressed these issues: Fisher Vectors (FV) [63] and Vectors of Locally Aggregated Descriptors (VLAD) [64]. A comparison between BoW, FV and VLAD is illustrated in Figure 2.5. The idea behind FV was to replace the k-means algorithm used to generate the visual dictionary in Bow by a Gaussian Mixture Model (GMM) fitting the feature space. Hence, each new local feature was assigned to a particular Gaussian from the mixture in a similar way as in the k-means algorithm. But, importantly, instead of keeping the index of the centroid, a set of derivatives from the underlaying GMM distribution were used to encode the feature. With this approach, less information is lost during the quantization, which improved the results and allowed the use of smaller vocabularies. The underlying idea behind VLAD is similar, but, instead of modeling the feature space with a GMM and keeping derivatives from it, it encodes the accumulated differences between the image local descriptors and their associated centroid (computed with k-means). Both FV and VLAD are attacking the same problem using a very similar idea: do not simply use the index of the centroid but some additional information that encodes the position of the feature with respect to the centroid in the feature space. Finally, in order to measure similarity between images, FV or VLAD signatures can be used in a similar manner to standard BoW.

2.1.3 From local descriptors to deep retrieval (2012-2020)

In 2012, Convolutional Neural Networks (CNNs) were brought back to stage in the work presented in [2], where a deep network called AlexNet surpassed the state-of-the-art performance in an image classification task by a large margin over existing methods. Even though the paper was focused on classification, in a qualitative experiment, the authors noted the potential of the image global features extracted using their network for other visually related tasks like image retrieval. Soon, the retrieval community started to propose methods to: 1) exploit CNNs to generate global descriptors; 2) substitute the SIFT descriptor by a CNN to generate the local feature vectors; 3) employ hybrid approaches that exploit both local and global CNN-based descriptors. The remainder of this section is devoted to the analysis of CBIR methods that use CNNs as deep retrieval approaches.

CNN global descriptors

An important work for deep retrieval using global descriptors was introduced in 2014 [6]. The authors analyzed the retrieval performance of a classification model [2] and found that the hand-crafted features based on Fisher Vectors [63] were still superior. However, if they fine tunned the classification model using a retrieval dataset, then they obtained a large boost in performance. Finally, they were able to produce CNN-based highly compressed global features that surpassed the state-of-the-art based on FV for smaller feature dimensions.

Up to this point, the architectures of deep retrieval models and the loss functions used for training were the same ones used for classification. The first CNN that was specifically tailored for image retrieval was proposed in [65]. The model combined a *triplet network* architecture [66] with a rank-based loss, known as *triplet loss* [67]. This work presented the first method that surpassed the state-of-the-art hand-crafted features by a significant margin. This approach still employed the fully connected features as in classification, but some authors discovered that convolutional features from the previous layers were more effective for retrieval [68, 69, 70]. Thus, a lot of work have been directed towards developing pooling layers for the aggregation of the convolutional features into compact and global representations that are useful for retrieval: max-pooling [71], average pooling [68], weighted average pooling [70], sum-pooling [69], hybrid pooling [72], regional pooling (R-MAC) [73] and generalized mean pooling (GeM) [74].

Other approaches have tried to pool the convolutional activations by imitating the aggregation mechanisms designed for previous hand-crafted features. In [75], the authors developed a trainable VLAD layer and created NetVLAD. Their model is capable of generating VLAD feature vectors in a single forward pass. Fisher Vectors approaches have also been explored in a similar way [76],

by adding layers that encode the FV parameters and losses that jointly learn global image representations and the FV parameters. However, those lines of work have been discontinued as simpler pooling mechanisms over the convolutional features (see GeM pooling [74]), have showed to outperform them.

Beyond architectures and pooling mechanisms to generate the global image representations, the other line of research that has received a great deal of attention is the design of retrieval loss functions. Since the contributions presented on this dissertation belong to this particular field, loss functions have their own related work (see Section 2.2).

CNN local descriptors

Given the ability of CNNs to generate successful feature vectors in many computer vision applications, its naturally appealing to explore the possibility of using them to describe image local patches. The first attemps to employ deep models to substitute SIFT-like methods were focused on individual parts of the SIFT pipeline. In [77], the authors proposed a learnable keypoint detector that outperformed the classical hand-crafted ones. In [78] a deep learning method is used to predict stable feature orientations.

Describing local patches found with classical detectors was another popular field. In [79], the authors present MatchNet, a model trained to learn local feature representation and the best metric to perform the later matching. Other similar approaches to learn how to describe local patches for different tasks, such as stereo matching, include [80, 81, 82, 83].

A milestone for the field came with the first deep local detector and descriptor that outperformed SIFT, it was called LIFT: Learned Invariant Feature Transform [84]. It borrowed and improved the ideas of previous approaches that only focused on particular steps of the complete pipeline, and integrated them into a end-to-end differentiable architecture. The next big step came with the introduction of DELF (DEep Local Features) [85]. The method included an attentive mechanism to perform a task-dependent keypoint selection, thus, avoiding to include patches with no relevant information for retrieval. Besides new architectures, advances were also made with respect to the loss functions required for learning the local representations. In [86], a novel loss function including an intra-batch mining strategy was used to learn local descriptors that outperformed models trained with the most common contrastive and triplet losses.

An important limitation of previous approaches is that they used a classical feature detector to generate an initial training set from which to learn. This in fact limits the performance of deep feature detectors to that of the reference method used to create the training corpus. To avoid this phenomenon, the LF-Net architecture was introduced [87]. The idea behind this approach is to exploit stereo pairs to train the networks without the need of explicitly giving the model

a set of pre-computed keypoints. Stereo pairs have the advantage that the geometric transformation between images is known a priori. Thus, LF-Net can learn from raw image data the best strategy to select keypoints with no bias. A similar approach is taken in AffNet [88], with particular emphasis on generating affine invariance descriptors.

Modern CNN-based local descriptors have shown a great improvement with respect to classical approaches using hand-crafted features. However, the CNN-based detectors have shown only limited success when compared to descriptors. The authors in [89] claim that this low repeatability for the detectors is due to their wrong estimation of the region affine parameters and their lack of robustness against scale variations. Thus, they propose Key.Net, a new architecture that makes use of both handcrafted and automatically learned feature detectors at the same time, as well as a multi-scale representation of images. In a more recent work [90], the cause for the lack of repeatability in CNN-based detectors is attributed to the low level image representations from which they are computed. Thus, instead of a "first detecting then describing" strategy, the authors propose to densely describe the whole image, and make the keypoints selection at the end of the process where high-level information is available. Other relevant work [91] claims that the models should detect not only salient points, but the subset of those that is expected to be reliably matched. Thus, they propose a new architecture that includes a regression mechanism to estimate the matching reliability of each keypoint, which allows a filtering process of irrelevant regions.

Lastly, there are some works trying to improve the descriptors too. In [92] the authors propose to augment off-the-self local descriptors and aggregate information related with the 2D keypoints distribution as well as visual context from high-level image representations.

CNN hybrid methods

Jointly exploiting global and local image representations is common in classical approaches. An initial raking is retrieved based on some form of BoW embedding, followed by a re-ranking strategy based on geometrical constrains applied over the matched local features. This strategy works because global descriptors have an edge for recall, given that they have the highest level information across the image, while matching local descriptors with geometric constrains is usually more reliable, and boosts precision [12]. In [93], a system to perform indoor localization based on global and densely-local CNN features is proposed. The authors employ the NetVLAD [75] global representation to retrieve a first set of candidates which are later re-ranked exploiting correspondences between low level convolutional activations from the same model. Other work that exploits global and local CNN descriptors is [94]. The method uses

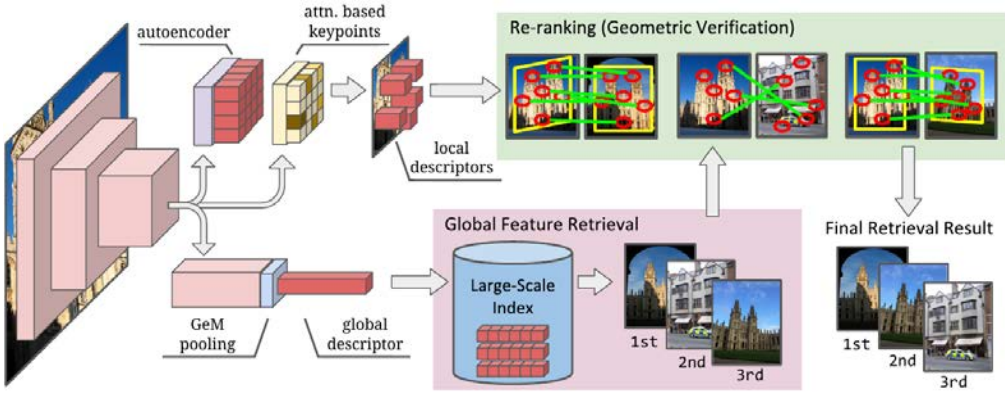


Figure 2.6: The hybrid CNN proposed in [12] to simultaneously compute image global and local descriptors.

the aggregated global descriptors to generate an initial ranking and, then, the convolutional activations before the pooling layer are interpreted as local features that refine the results via geometric verifications. The main drawback of these approaches is that they interpret intermediate activations of models trained to produce global features as local descriptors. However, since those intermediate activations were not optimized to be independently matched against other local regions, they yield suboptimal performance for such task.

Recently, some works propose to train CNN models to produce global and local features explicitly, using several output branches and multi-task losses as illustrated in Figure 2.6. In [95], the authors use the outputs from global and local independent CNN teachers, to train a single unified CNN student that generates both global and local outputs using several branches on a hierarchical structure. In [12], a method is proposed to train a single CNN to generate both local and global features in an end-to-end manner, avoiding the need of mimicking individual disjoint networks. The resulting model contains two output branches: one produces global descriptors and tries to minimize a global loss function, while the other incorporates and attention model to produced sparse local features trying to minimize a local loss objective.

In this thesis, we propose two loss functions to produce image global descriptors using deep models trained under noisy datasets. Our contributions can be employed with both global and hybrid CNN architectures to generate suitable feature vectors for CBIR. In the remainder of this chapter, we review the literature related to loss functions over global image representations in CBIR in Section 2.2, and methods to deal with noise on the training datasets in Section 2.3.

2.2 CNN Loss Functions to Learn Image Global Representations for CBIR

In this section, we categorize and review the main loss functions used for generating global CNN features for Content-Based Image Retrieval (CBIR), and discuss their connections with our proposals. In particular, our classification is based on a granularity analysis of the feature space and leads to the following three categories:

1. **Pair-based**: the methods of this family compute the loss by directly measuring distances between image representations. They work with small tuples of samples such as pairs [19], triplets [67, 96] or n-tuples [97, 98, 99]. Their goal is to enforce the proper distribution of features in a *local* neighborhood of the feature space. Given that the loss deals with small tuples, one at a time, uniformly sampling them is not a good strategy: most of the tuples might be either too easy to learn anything meaningful, which slows down the training, or redundant with others, which causes overfitting. Several authors have made comparative studies about the importance of mining on image retrieval systems [100, 101, 102], which in turn has stimulated the development of several complex mining techniques [103, 104, 105].

2. **Center-based**: this family of losses exploits the hidden structure of the feature space by setting-up local neighborhoods around identified centroids and following an approach similar to the one used in classification. During training, each category is represented by a centroid [106, 107, 108] (or a set of them [109, 110]) and the losses aim to concurrently learn the feature representation and the set of centroids that minimize intraclass distances while maximizing interclass ones. Hence, the difference of this approach and traditional classification losses is that the functions are modified to rank distances instead of providing per-category probabilities. Then, during test, new samples (representing new unseen objects/categories) are projected onto the set of centroids and image similarities are computed using these projections. These methods are typically used for Fine-Grained Image Retrieval (FGIR) [111] tasks, where all train and test instances share a common topic. Thus, the visual patterns encoded by the learned centers are likely to fit well unseen test images of the same topic.

3. **List-based**: these loss functions also make use of distances between pairs of images but, instead of computing the loss directly over them (as the pair-based do), distances are used to build intermediate list-like structures, such as soft-binning histograms, from which a final loss is derived

[112, 113, 114, 115]. Since these losses take into account (potentially) long lists containing all images in the training set, they circumvent the problem of locality, and provide a global analysis of the feature space. Besides, these rank-list structures can be used to approximate performance measures such as the Average Precision (AP), which is more closely related to goal of the task than those performance functions implicitly considered by the tuple-based methods [116].

In this thesis, we contribute to the field with two novel losses. The first one is a classical pair-based loss but modified to internally estimate soft labels based on image content and metadata. This loss is paired with a classical hard sample mining mechanism. The second one, although belongs to the pair-based family, shows a fundamental difference regarding the granularity of the analysis: inspired by the Multiple Instance Learning (MIL) framework, we use large tuples of matching images, called bags, that therefore provide a global view of the feature space, and a loss that automatically identifies the relevance of each training image pair. To be more precise, the proposed bag loss involves a sample weighting mechanism which allows the models to perform a soft-mining of samples which adapts dynamically as learning progresses.

2.3 Dealing with Noise in Training Datasets for CBIR with CNNs

Dealing with the noise present in a training dataset has also received a great deal of attention in the field of CNN-based image retrieval. The most common approach consists in reducing noise before starting the training process. The simplest method is to do it manually [6], but it is time-consuming and prone to errors, which prevents its broad adoption. Conversely, automatic systems based on meta-data or image content are popular in the literature. Some authors filter noise by making use of contextual information such as Global Positioning System (GPS) coordinates [117]; others employ local feature matching with spatial verification [118]; and others exploit 3D models generated with a structure-from-motion algorithm to label outlier images as noise and remove them from the training set [4].

In contrast, training deep models for image retrieval with noisy datasets is uncommon due to the resulting performance degradation, as it has been pointed out by some authors [4]. Nevertheless, approaches for reducing the influence of noise in performance and the biases introduced by automatic cleaning tools or human annotators have been explored in other tasks [119, 120]. In classification problems, a transition matrix has been proposed to model the probability of a sample switching its label to another class. Some authors estimated this matrix

by means of an affine layer [119], others proposed a loss correction mechanism to derive it [121], and others generated a subset of noise-free data from which to carry out the estimation [122]. Other approaches for training classification models with noisy datasets include the creation of noise resilient losses [123]. However, as noted in [124], though those noise-robust losses tend to work well on synthetic data, due to certain unrealistic constrains or assumptions, they underperform on real world datasets. Finally, another family of methods suggests the use of secondary networks to correct the training labels [125], perform meta-learning by comparing the gradient directions from different samples [126], or simulate gradient-update steps with synthetic noisy labels to avoid overfitting [127].

Other computer vision tasks, such as object detection, have also benefited from using robust-to-noise loss functions. In [120] the authors deal with noise in the training labels by generating bag of samples and using a constrained weak loss with inequalities applied over the accumulated probabilities within the bag. In [128], the authors relied on Multiple Instance Learning (MIL) to ensure that data augmentation techniques that are not label-preserving could be used safely. MIL is a form of weakly supervised learning where training instances are arranged in sets, called bags, and a label is provided for each bag, instead of for each instance. It has been traditionally used in tasks where the granularity of the expected outcomes and the available labels is not the same, and aggregation methods are required to bridge the gap between data and labels. The application of MIL principles in the context of CNNs involves the use of aggregation operators, such as average [129], maximum [130, 131], Log-Sum-Exp (LSE) [129], global weighted rank-pooling [132], negative evidence models [133], or weighted average of regions with maximum and minimum scores [134].

In this thesis, we first explore the concurrent use of visual and contextual information, such as GPS coordinates, to generate weak labels that prevent losses from overreacting to noisy data. Later, we exploit only visual content within a MIL framework. From our point of view, MIL arises as a natural formulation to deal with noise in training databases for image retrieval. Therefore, inspired by this concept, we propose a novel loss function that builds bags of matching images to naturally handle noise.

Chapter 3

A Novel Soft-Matching Loss to Learn Image Global Representations Based on Content and Meta data

In this chapter, we introduce the theory behind the Soft-Matching (SM) loss, the first contribution of this thesis. Section 3.1 contains an introduction describing the problem and main motivations that guide our work, that is later described in Section 3.2. In particular, Subsection 3.2.1 sets forth a series of intuitions that help to understand the fundamentals of our method, while Subsections 3.2.2 and 3.2.3 presents the mathematics. Finally, Subsection 3.2.4 exposes some possible applications for our approach as well as its most notorious current limitations.

3.1 Introduction

In 2017, Facebook's users generated a total of 300 million photos per day*. The amount of new multimedia content has grown exponentially for the past decade, and it is now so staggering that storing, managing, indexing and organizing user-generated files efficiently is one of the main technological challenges for the industry. During the last few years, the scientific community has tackled this problem by means of novel computer vision techniques aiming to automatically obtain content-based image descriptors which are distinctive, compact, and allow an efficient search [135][136][137].

The best tool we have today to compute the necessary content-based descriptors to index images are Convolutional Neural Networks (CNNs) [2]. These deep architectures have shown its superior performance in a wide variety of

*https://zephoria.com/top-15-valuable-facebook-statistics/

tasks. Modern deep classification models, such as residual networks [138], have achieved human-like performance on the ImageNet challenge [16], where a thousand object categories are recognized in a set of a few million images. Segmentation models [139], that cluster together pixels that belong to the same object class, have also surpassed traditional approaches in tasks such as city scene understanding [140] or general objects segmentation [141]. Object detection [142], which aims to locate all instances of a predefine set of objects on images, has also benefited from the use of CNNs, autonomous driving is probably the most known case of use of such networks [143]. Image retrieval, where pictures are recovered based on the presence of specific object instances, is another field that has experience a notable improvement with the advent of CNNs [144].

Although all the previous applications share a common underlying CNN architecture, the nature of the computed image descriptors is task-dependent. For classification, the model should learn the common visual patterns of the objects belonging to the same category, i.e., objects of the same category must be described by similar feature vectors regardless of the intra-class variability (e.g. all cars). For retrieval, only instances of the same specific object should lie nearby in the feature space (e.g. a specific car make and model), while other objects should stay far away, even if they belong to the same semantic category (e.g. other cars). The models learn to adjust to the given nature of the problem by means of specific task-oriented training sets and loss functions. This is one of the weaknesses of CNNs with respect to classic local image descriptors which were applicable over a wider range of application with little to no change.

Another known issue with the use of CNNs is that their performance strongly depends on the correlation between the scene or objects present in the train and test sets [6]. It is not only that a task-related training set is needed, but also that the topic correlation between the train and test sets needs to be high to get an optimal performance. For instance, a retrieval model trained with cars is not expected to perform well when facing animals, since the visual patterns learned from the cars will unlikely fit well the animals. Under these circumstances, to deploy a retrieval system on a new domain, we envisage two options: 1) use a highly general model trained under thousands of varied topics, which is expected to fit reasonably well the new visual domain or, 2) create a new specific topic-related dataset for the task. None of the above options is ideal for different reasons: the general model approach is quick but only moderately effective, whereas the creation of a new dataset is slow and labor intensive, although it ensures optimal performance.

In the following sections, we introduce a pair-based Soft-Matching (SM) loss that is capable to adapt general CNNs to new domains with noisy training sets, which are easier to generate without human intervention. In particular, image content and meta data are jointly exploited to infer soft labels that, en-

Figure 3.1: (left) Geo-location of six photos shown on a map of Jerez (Spain). (right) Corresponding photos.

coding the probability of a pair of images of being a true match, allow to fine tune general models to better represent certain topics. To illustrate one case of use for the proposed SM loss, in Chapter 4 we will adapt general classification CNNs to better represent images from particular cities or regions, boosting the performance in a subsequent landmark discovery task.

3.2 The Soft-Matching (SM) Loss

3.2.1 Intuitions

To introduce the proposed loss function we will make use of geotagged landmarks datasets, where images are enriched with their corresponding GPS coordinates. We have selected the landmarks domain because of the easiness to obtain large collections of images with their corresponding metatada from public repositories. In Section 3.2.4, other potential scenarios of application for our loss will be discussed.

Geo-location systems can incur in measuring errors when determining positions. Even though we are aware of this fact, in our work we assume accurate GPS coordinates. Hence, the main source of noise in our scenario is not due to meta data inaccuracies, but to the fact that two pictures taken on spatially close locations are not necessarily visually related. To illustrate this issue, let us introduce Figure 3.1, where six pictures with their geo-location are shown. The pair of images (D,E), were taken barely ten meters away from each other, similar to what happens with the pair (A,D). However, D and E show different objects while A and D show the same. Thus, geo-location, even when accurate, yields noisy visual relations between samples. In consequence, in our approach we consider image meta data as a form of weak supervision, which is complemented with the image content itself. By jointly considering both sources of

information, it is possible to draw more reliable relations between the dataset instances, which can be further exploited with the final objective of CNN specialization using Distance Metric Learning (DML) [145].

DML and its application on pair-based loss functions has been extensively studied in the context of pure matching tasks [19][146]. The goal is, given a pair of either matching ($y = 1$) or non-matching ($y = 0$) images, to learn how to embed their representations in the feature space, leading to matching pairs that locate closer than non-matching ones. In general, this objective is analytically stated by penalizing the square euclidean distance for matching pairs, and the negative counterpart for non-matching ones. In noise-free scenarios, this is a sensible approach that achieve very good results. Unfortunately, these loss functions are very sensitive to noise. For instance, false positives are likely to present larger euclidean distances than true positives. Furthermore, since the loss penalty is square w.r.t. the distances, false positives quickly become dominant in the training process.

3.2.2 From image and meta data to soft labels

The common binary labels used on pair-based loss functions for DML are not adequate for training sets where the visual relations are noisy. Instead, we propose to estimate the true match probability of any given pair of images, and exploit those estimations by encoding them into weak labels that can be swapped in place of their existing binary counterparts in pair-based losses from the literature. In this manner, we can turn noise-sensitive losses into noise-robust ones.

In order to estimate the true match probability of a pair of images is not enough to use their GPS coordinates. As illustrated in Figure 3.1, two images taken nearby do not necessarily share a visual relation. Thus, the geo-location is only a clue that needs to be complemented with visual content. GPS coordinates are available on image meta data, but visual information needs to be extracted by other means. In our work, to extract the complementary visual clues we propose to employ a *general CNN*. By "general" we understand a model that has been trained for a related task on a large variety of topics. Thus, it is expected to fit reasonably well any kind of object on our datasets. Once this general CNN is used to compute the visual descriptors for every image in our training sets, it will be used as initialization for our adapted models. Hence, for the remainder of this chapter, we refer to general model as the *baseline* CNN and to our fine-tuned version as the *adapted* CNN.

The process for estimating the true match probability for the i-*th* pair of images in the training set is based on their content and geo-location, as depicted in Figure 3.2. First, the pictures (I_1, I_2) of the i-*th* pair are forwarded through the baseline-CNN to obtain their feature representations encoding visual information (f_{B_1}, f_{B_2}). Then, the visual similarity between the images is computed as

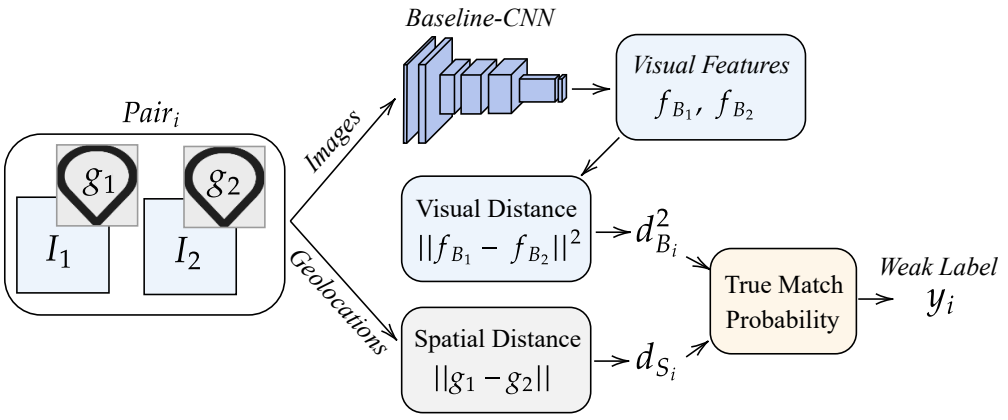


Figure 3.2: The process for estimating the true match probability of the i-*th* pair of images on the training set. d_{B_i} is the visual euclidean distance between the feature representations computed with the baseline-CNN. d_{S_i} is the spatial euclidean distance between their GPS coordinates. y_i is the true match probability used as a weak label in the proposed loss.

the square euclidean distance between their feature representations ($d^2_{B_i}$). Similarly, the geo-location of the images is used to compute their spatial distance, i.e., how close in meters the pictures were taken ($d^2_{S_i}$). Finally, the true-match probability (y_i) of the i-*th* pair, is estimated by means of a function working with their visual distance (computed with the baseline model) and their spatial distance (as extracted from the GPS meta data).

Let us now discuss the nature of the function computing the true-match probability that will be used as a weak label in the proposed loss. We would like to assign a true-match probability $y_i > 0.5$ to those pairs of images that jointly exhibit a low visual and spatial distances. In other words, pictures that were taken nearby and look alike (e.g. pair (A,B) in Figure 3.1). In contrast, we expect to assign a value $y_i \leq 0.5$ to pairs of images that are visually very different (e.g. pair (E,D) in Figure 3.1) or, that are visually related but were taken too far away to physically belong to the same landmark (e.g. pair (D,F) in Figure 3.1). For this purpose, we set up two thresholds: 1) T_B related to the distance of visual features computed with the baseline network; and 2) T_S associated with the spatial distance of the landmarks in the physical world. In order to obtain a robust solution that is aware of the actual data statistics, we derive the threshold T_B from the distribution of square visual distances $d^2_{B_i}$ computed using the baseline model. Specifically, assuming a Gaussian distribution:

$$T_B = \mu_B - k\sigma_B \tag{3.1}$$

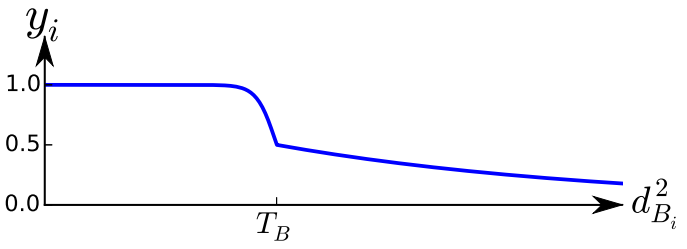


Figure 3.3: Piece-wise function defined by equation 3.2 for $d_{S_i} \leq T_S$ as a function of $d^2_{B_i}$. The left hand side of T_B is almost flat giving positive pairs ($y_i > 0.5$) large weights. The right hand side decreases slowly to avoid penalizing too much false negatives lying close to the T_B threshold .

where μ_B is the average and σ_B the standard deviation of the $d^2_{B_i}$ distribution. We have found experimentally that a suitable range for k is between 2.0 to 2.5. Concerning the spatial distance, a simple threshold of $T_S = 300$ meters performs well in our training sets and generalizes properly on unseen locations.

Once we have settled both thresholds, we use the following piece-wise function to compute the true match probability y_i based on visual $d^2_{B_i}$ and spatial d_{S_i} distances:

$$y_i = \begin{cases} 0 & \text{if } d_{S_i} > T_S \\ \frac{1}{1+\exp\left(d^2_{B_i} - T_B\right)} & \text{if } d_{S_i} \leq T_S, d^2_{B_i} \leq T_B \\ \exp\left(\frac{\ln(1/2) d^2_{B_i}}{T_B}\right) & \text{if } d_{S_i} \leq T_S, d^2_{B_i} > T_B, \end{cases} \tag{3.2}$$

where it should be noted that, given $d_{S_i} \leq T_S$ (i.e., the images of the i-*th* pair are close enough according to their geolocation), the threshold T_B on the visual distance decides between $y_i \geq 0.5$ and $y_i < 0.5$ as illustrated in Figure 3.3.

Let us gain some insight into eq. (3.2) by discussing its main advantages: (a) the true-matching probability y_i can be used as a soft label in a loss function to put more or less emphasis on certain pairs of images, thus, gaining the ability to produce stronger gradients for high-confidence pairs and weaker gradients for more doubtful cases; (b) the piece-wise function allows to establish asymmetric behaviors at both sides of the threshold T_B. In particular, we have observed that, by setting an appropriate conservative threshold (i.e., a low T_B value that requires highly visual similar images to produce a $y_i > 0.5$ label), most image pairs with distances below T_B show the same visual scene, and the variations in the distance are usually due to factors like varying viewpoints or

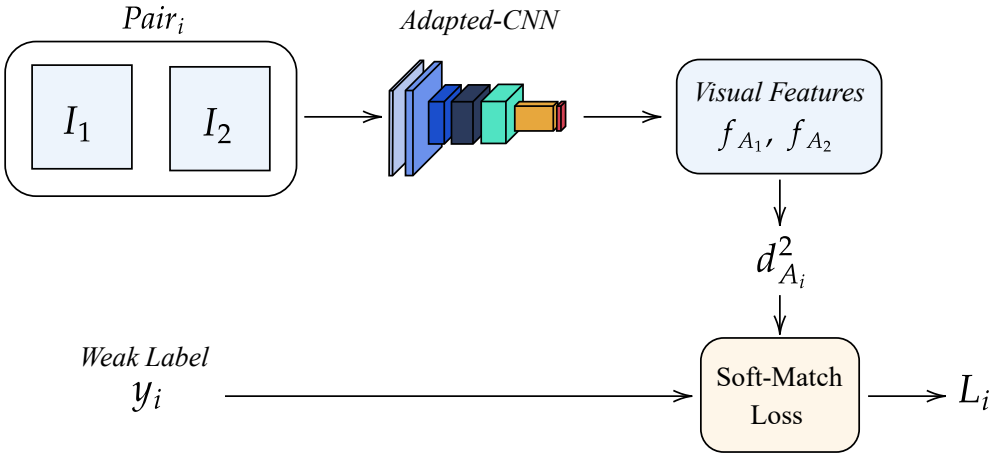


Figure 3.4: The process for computing the Soft-Matching (SM) loss for the i-*th* pair of images on the training set. d_{A_i} is the visual euclidean distance between the feature representations computed with the adapted-CNN. y_i is the fixed true match probability computed using the baseline-CNN for the i-*th* pair. L_i is the final computed SM loss

illuminations. Hence, we have designed a flat curve on this piece of the equation to ensure a similar contribution for all of them. However, if the threshold is conservative, we may yet find related image pairs with distances above T_B. In order to avoid the assignment of too low true-match probabilities to those cases, a slowly decreasing slope seems to be more appropriate (see right part of the curve in Figure 3.3).

3.2.3 Analytical shape

The proposed Soft-Matching loss addresses the noise sensitivity problem on other pair-based approaches, by substituting the classic hard binary matching labels by soft probabilistic versions that encode the true-matching probability of each pair. The goal is to mitigate the influence on the training process of those pairs of images for which the true label is more uncertain, and give more weight to the others. The process for computing the loss for the i-*th* pair on the training set is illustrated in Figure 3.4. First, the true-match probabilities for all possible pairs in the training database are estimated and fixed, using the procedure explained in Subsection 3.2.2 that employs the baseline-CNN and the image geo-location. Second, a new adapted-CNN model is initialized using the weights of the baseline CNN. Then, the pictures (I_1, I_2) inside the i-*th* pair are forwarded through the adapted-CNN to obtain their feature representations (f_{A_1}, f_{A_2}), which are then used to compute the visual similarity as the square euclidean distance between them $(d^2_{A_i})$. Finally, the Soft-Matching loss receives

as input the true-match probability for the pair (y_i) and uses it as a soft label in combination with the visual distance ($d^2_{A_i}$) to calculate the loss for the i-*th* pair (L_i).

Our function borrows the general analytical form of the contrastive function [19] and computes the loss for the i-*th* pair as follows:

$$L_i = \frac{1}{2} y_i d^2_{A_i} + \frac{1}{2} (1 - y_i) \, max \left(0, m - d^2_{A_i}\right) \tag{3.3}$$

where $d^2_{A_i}$ is the square Euclidean distance between the visual features of the i-*th* pair computed using the adapted-CNN; $m \geq 0$ is a margin that avoids that very dissimilar pairs keep contributing to the loss; and $y_i \in [0, 1]$ is the true-match probability computed in Subsection 3.2.2, that signals whether the generated visual descriptors for a pair of images need to be pushed closer ($y_i \geq 0.5$) or further away ($y_i < 0.5$) in the feature space.

Let us now discuss the effects of different weak label (y_i) values in the SM loss (equation 3.3). First, for an i-*th* pair of visually similar images that were taken nearby, we expect to have $y_i \to 1$. This, will increase the weight of the first term of the loss that penalizes large square euclidean distances. Hence, the function will generate a strong gradient response aiming at pulling together the images in the feature space. Second, if images are visually different or were taken too far away from each other, the weak label will take values $y_i \to 0$. This causes the contribution from the second term of the loss to decrease, and a strong gradient response towards pushing the features away will dominate. Last, for those pairs of images where there is more uncertainty about their matching relation, we expect our method to produce $y_i \approx 0.5$, leading to a balanced contribution from the *pull-push* terms in the loss, and avoiding large gradients on either direction.

To ensure convexity, both the soft labels (y_i) and the margin (m) must be fixed during learning. To that end, we compute the margin and the soft labels for every pair of images in the dataset only once, using the baseline model before training begins.

3.2.4 Applications for the SM loss and current limitations

Other SM loss applications

We have introduced the SM loss for a particular source of metadata, the geo-location. However, there are other scenarios and sources of weakly supervision that will benefit from this kind of approach. For instance, suppose a cars dataset where only the manufacturer is known, but the information about the specific

model is missing or contain errors. In this scenario, the car make will take the role of the geo-location in our loss, and again, a general model will be used to extract image features, which, combined with the car make information, might generate the soft labels necessary to specialize a CNN for retrieval of specific car make and models. A similar strategy can be used to generate a specialized network for paintings retrieval. By knowing with certainty the name of the author, and combining that metadata with some content-based features from a general model, a set of soft labels might be generated to adapt a model to carry out retrieval of particular paintings. Other similar visual domains, where only partial information about the labeling is known, might benefit from a similar approach to the one proposed in this chapter, based on jointly exploiting content and meta data.

Current limitations

Our method presents two important limitations. First, it needs some form of weakly supervision to limit the presence of false positives. For instance, consider the pair of images (D,F) from Figure 3.1: they are known to be a non-matching pair because of the geo-location. However, they are visually quite similar. Thus, the baseline model will fail to locate them far apart in the feature space before any adaptation is carried out.

Second, generating fixed soft labels with the baseline model prior to the training process is a limiting factor for the degree of proficiency that the specialized model can attain. For instance, suppose that a soft label $y < 0.5$ is assigned to the pair of images (C,D) from Figure 3.1. At the same time, pairs (A,B), (A,C), (A,D) could get $y > 0.5$. Since all those images are actually depicting the same facade of the cathedral, the network is asked to perform a contradictory visual task. Unfortunately, and due to the high expressibility of modern CNNs, they might be able to do so via over-fitting. A possible solution to prevent this effect consists in recomputing the soft labels at the beginning of each epoch, using the updated specialized model from the previous one. If the model has learned from the true positives how to describe that particular facade, maybe the pair (C,D) can change its label and become positive ($y > 0.5$). We have experimentally found that this is rarely the case. Once the model has been instructed to push away a pair of images, the label of this pair rarely changes from negative to positive. Besides, changing the labels continuously, makes the optimization process non-convex, and leads to worse results than keeping the labels fixed from the beginning. We hypothesize that this phenomenon is due to the small size of our training datasets and could be alleviated using a larger corpus.

Chapter 4

Assessment of the Soft-Matching Loss in a Landmark Discovery Task

In this chapter, we present a set of experiments testing the capabilities of the Soft-Matching loss presented in Chapter 3. Section 4.1 contains the main motivations justifying the choice of a landmark discovery task to assess out method. Section 4.2 describes the general system for landmark discovery. Sections 4.3 to 4.6 introduce the datasets, compared losses, evaluation metrics and experimental setup necessary to replicate our results. Section 4.7 contains the main results achieved, and Section 4.8 performs two ablation studies showing how these results are affected when different parts of our method are suppressed. The chapter ends outlining some conclusions in Section 4.9

4.1 Introduction

The main goal of our approach is to provide location-adapted CNN visual features enabling subsequent end-user tasks. To prove the effectiveness of our method, in this chapter we have chosen the specific task of automatic landmark discovery to assess our model. There are several important reasons for choosing this application domain to prove our claims. First, there exist public repositories containing thousands of landmark images with their associated meta data that are readily available to train the models. Second, the landmark discovery task has been a common topic in the computer vision literature; thus, the task in itself is important for the community. Finally, and most importantly, the landmark discovery problem allow us to show how jointly exploiting image content and meta data can be used to learn location-adapted deep models that provide tuned image descriptors for specific visual contents.

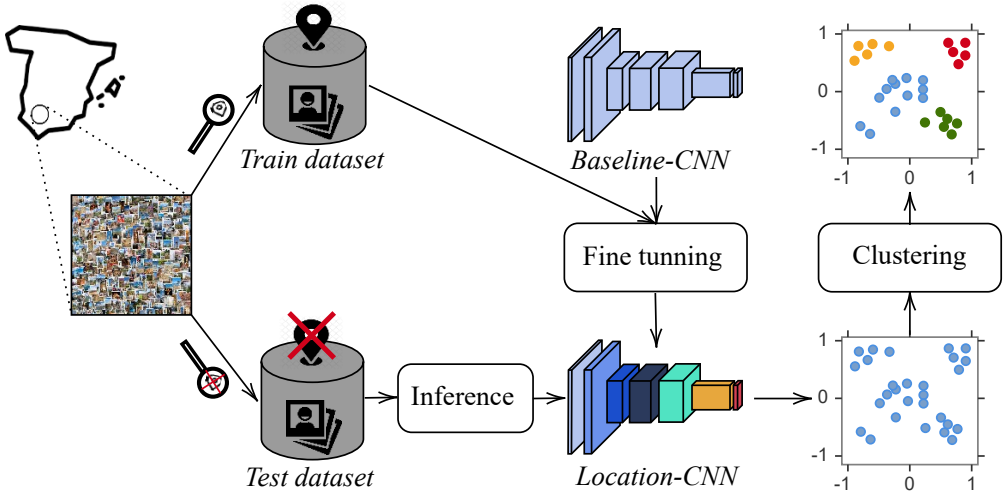


Figure 4.1: The proposed system to benchmark the Soft-Matching (SM) loss on a landmark discovery task: first, all images available from a region are gathered; then, we pick up those with geolocation and use them to fine-tune a baseline-CNN model into a location-adapted CNN; lastly, we employ a controlled test set to compute visual features that are finally passed to the clustering algorithm.

4.2 The Landmarks Discovery System

To evaluate the performance of our loss in a landmark discovery task, we have implemented the system depicted in Figure 4.1. First, we automatically gather from Flickr all available images from a particular location (city, region, etc.) using both geo-location and textual tags to build the query. On average, only 15% of the images are geo-located. The recovered images are divided into two subsets: 1) a training set containing the geo-located samples; and 2) a test set with the remaining images. The test corpus has been manually filtered and labeled to generate the experimental ground truth. Second, an initial **baseline** model that was originally trained for a different task, such as classification or retrieval, is fine-tuned using our proposed Soft-Matching (SM) loss on the training set, producing an adapted network that we refer to as **location-CNN** (the Chapter 3 adapted-CNNs used for locations). Finally, the baseline and location CNNs receive the test set and compute descriptors that are clustered with an automatic algorithm to perform the discovery of landmarks. The assignments are then compared with the ground truth to evaluate the system performance.

It is worth noting that the proposed landmark discovery system is executed in a fully automated manner, since the training labels are generated based on the images meta data. In our experiments, human intervention is only needed to create a controlled ground truth that allow us to numerically evaluate the effectiveness of the solution and compare different alternatives.

Table 4.1: Training and test sets statistics

	Training sets			Test sets	
	Images	Pos pairs	Neg pairs	Images	Landmarks
Jerez	1000	1.5k	250k	1000	19
Madrid	4000	30k	4M	3900	15
Rome	4000	30k	4M	3100	14

4.3 Datasets

Our dataset contains Flickr images from three different cities in Europe: Rome (Italy), Madrid and Jerez de la Frontera (Spain). For each city, a train set of geo-located images has been gathered within a $10km$ radius around the center of each city. Additionally, we have used a list of generic keywords that helps to retrieve images that are relevant to our task, namely: *landmark, monument, building, park or art*. Finally, we filter out the results by allowing only one image per user in order to avoid duplicates. The test set has been built by searching for a predefined list of famous landmarks in each city, and manually cleaning the initially retrieved results. In order to provide a fair analysis, the sets are disjoint so no images are present in both sets. Table 4.1 summarizes the number of images and landmarks per city in the corresponding training and test sets, as well as the resulting number of positive and negative sample image pairs.

4.4 Compared Losses

In our experiments, we have used three loss functions to fine-tune location-CNNs. In the next paragraphs, we provide a brief description of the considered losses:

1. **Contrastive (CT)**[19]: one of the first and most successful losses used to train deep models for retrieval. It is a pair-based function that receives two images and a binary label, and tries to bring matching pairs closer in the feature space, while pushing away non-matching ones, until a margin is met.

2. **Triplet (TL)**[67]: another widely used loss for image retrieval. It uses an anchor, a positive and a negative image, and attempts to increase the relative distance between the anchor-negative pair and the anchor-positive

one, up to a margin. It uses binary query-positive and query-negative relations as CT loss.

3. **Soft-Matching (SM)**: our proposed loss function. It incorporates a mechanism to exploit image content and meta data in order to generate soft labels measuring the true matching probability of any given image pair. Then, those estimations are inserted as soft labels in a contrastive function to turn the noise-sensitive contrastive into our noise-robust SM loss.

4.5 Evaluation Metrics

To assess our models in the task of automatic landmark discovery, we have generated the visual descriptors of images in the test set, and used the *k-means* algorithm with the pre-defined number of landmarks to cluster these descriptors. The resulting partitions are then compared with the ground-truth using three classical clustering evaluation metrics: the Rand (R) [147], Fowlkes-Mallows (FM) [148] and Jaccard (J) [149] indexes. These indexes are based on counting pairs of images whose members lie in the same or different clusters when comparing the ground truth and the estimated labels. Given the ground truth clustering partition C, and an estimated partition C', the three considered metrics are computed as follows:

$$R(C, C') = \frac{n_{11} + n_{00}}{N_{pairs}} \tag{4.1}$$

$$FM(C, C') = \frac{n_{11}}{\sqrt{(n_{11} + n_{10})(n_{11} + n_{01})}} \tag{4.2}$$

$$J(C, C') = \frac{n_{11}}{n_{11} + n_{10} + n_{01}} \tag{4.3}$$

where n_{11} is the amount of pairs of images whose members are assigned to the same cluster in C and C'; n_{00} is the amount pairs of images whose members are assigned to different clusters in C and C'; n_{10} is the amount pairs of images whose members are assigned to the same cluster in C, but to different clusters in C'; and n_{01} is the amount pairs of images whose members are assigned to different clusters in C, but to the same in C'.

The metrics range from zero to one being one the perfect clustering. An in depth analysis of these metrics to compare clustering performance is given in [150].

4.6 Experimental Setup

Starting from a ResNet50 network [138] pre-trained on ImageNet[16] as a common baseline, three independent models have been fine-tuned for the corresponding training sets: **JerezNet**, **MadridNet** and **RomeNet**. The networks are trained for 10 epochs with 1000 batches per epoch using Batch Gradient Descend (BGD). Each batch contains 40 pairs of images, from which at least 10% of them are positive ($y_i \geq 0.5$) and the rest are negative ($y_i < 0.5$). Additionally, a particular image is only included once per batch, allowing us to safely use BGD instead Stochastic Gradient Descend (SGD), as it is a common practice when working with pairwise loss functions. For the parameters update, we have employed 0.9 for the momentum term, 10^{-5} as learning rate and 10^{-3} as weight decay. The affine layers of the original model have been removed and we have kept the output of the last average pooling as our feature. The models were trained using the open deep learning library PyTorch* on a NVIDIA TITAN XP GPU.

To cluster the descriptors generated by the different models, a *k-means* algorithm with a pre-defined number of clusters (the number of landmarks in the test set) has been used. For the shake of stability and statistical significance, we have repeated the clustering process ten times to account for different *k-means* random initializations. This clustering algorithm has been selected due to its simplicity, as it allows us to better isolate the influence of the proposed learning framework from the potential influence of the parameters of the clustering method.

4.7 Results

In this section, we present the main results obtained for our SM loss on an automatic landmark discovery task. In Subsection 4.7.1, we focus on the quantitative results achieved by our location-adapted CNNs in comparsion with the baseline models. Then, Subsection 4.7.2 includes some visual qualitative results than allow to gain some insight into the causes leading our networks to outperform the general models on this task. Finally, Subsection 4.7.3 presents an error analysis explaining the main weaknesses of our adapted location-CNNs.

4.7.1 Quantitative results

Table 4.2 shows the averages and standard deviations of the Rand, Jaccard and Fowlkes-Mallows indexes obtained in our experiments, using either visual descriptors generated by the baseline network Resnet50 or by our proposed

*http://pytorch.org/

Table 4.2: Average and standard deviation of the Rand, Jaccard and Fowlkes indexes as a result of comparing the proposed models: JerezNet, MadridNet and RomeNet; with the baseline model ResNet50. Best results highlighted in **bold**

Test set	Model	Rand	Jaccard	Fowlkes
Jerez	ResNet50	0.9071 $\pm$ 0.0201	0.3203 $\pm$ 0.0145	0.4944 $\pm$ 0.0208
	JerezNet	**0.9242** **$\pm$ 0.0201**	**0.3834** **$\pm$ 0.0323**	**0.5627** **$\pm$ 0.0254**
Madrid	ResNet50	0.9254 $\pm$ 0.0051	0.3817 $\pm$ 0.0239	0.5467 $\pm$ 0.0322
	MadridNet	**0.9547** **$\pm$ 0.0189**	**0.5911** **$\pm$ 0.0261**	**0.7551** **$\pm$ 0.0257**
Rome	ResNet50	0.9174 $\pm$ 0.0157	0.3728 $\pm$ 0.0298	0.5692 $\pm$ 0.0182
	RomeNet	**0.9345** **$\pm$ 0.0177**	**0.5473** **$\pm$ 0.0138**	**0.6625** **$\pm$ 0.0125**

location-adapted CNNs (JerezNet, MadridNet, RomeNet) fine-tunned under our proposed loss. Results show that for all the evaluation indexes and test sets, the location-adapted CNNs provide a notable improvement over the baseline. This means that the proposed SM loss is forcing models to dismiss visual structures from landmarks that might be prominent but not unique (and therefore not discriminative). Our method is effectively shifting the CNNs attention from common semantic visual features (doors, windows, cars, people) to specific distinctive elements of a particular city or region.

The large difference between the Rand index and the other two metrics is due to the nature of the measures. The Rand index is highly biased towards true negatives, i.e., pairs of images whose members were not in the same cluster either in the ground truth or in the estimated labels, which are the vast majority for any reasonable sized database. This can be observed in the formulation for the Rand index in equation 4.1, which is the only one including the term n_{00} in the computation. The other two indexes neglect true negatives, providing more stable results over different dataset sizes and number of clusters.

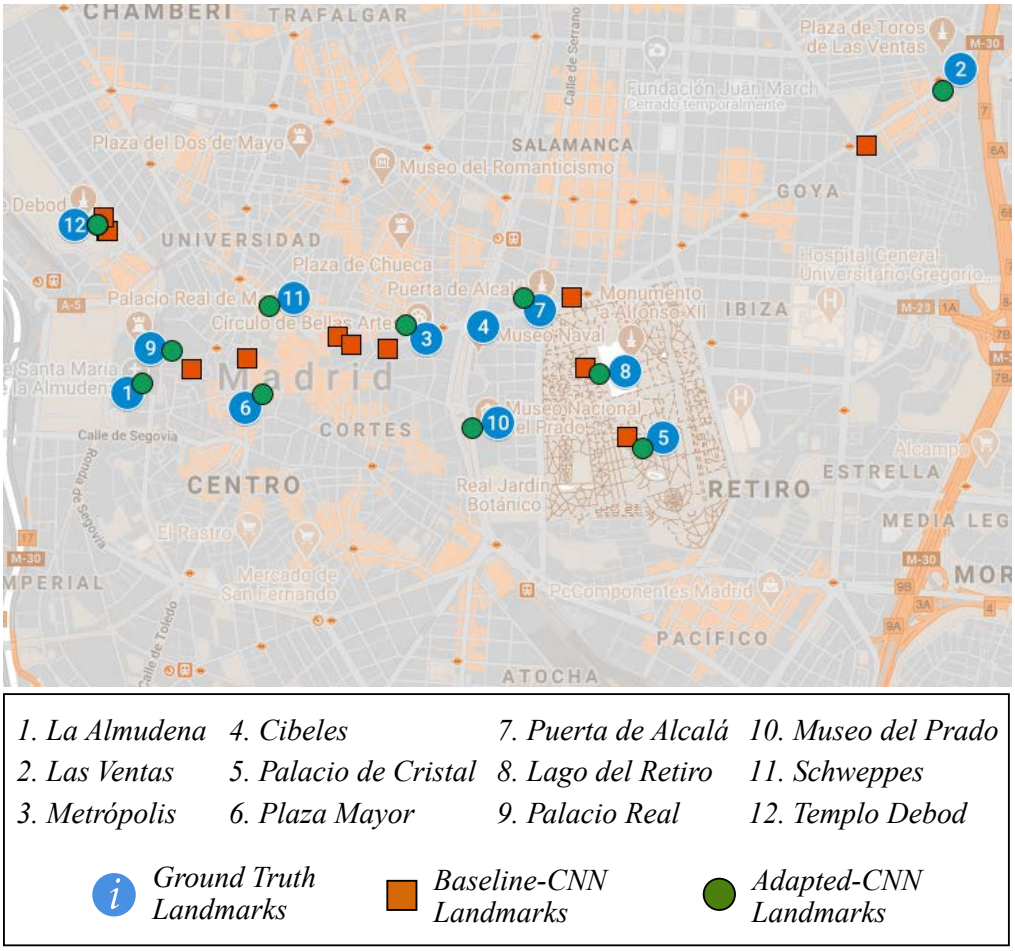


Figure 4.2: A map showing the locations of the landmarks on the Madrid test set (see Table 4.1). The figure includes the ground truth (blue circles), and the landmarks discovered by clustering the test images using the baseline-CNN (orange rectangles) and the location-CNN (green circles). Only the closest twelve of the fifteen available landmarks on the Madrid test set are shown to allow a better visualization.

4.7.2 Qualitative results

To visually illustrate the benefits of our method, Figure 4.2 shows a street view map of Madrid downtown containing three sets of GPS cluster centroids. Before we present an analysis of the map, let us first discuss the way in which it has been generated. All the points shown in the map are based on the test images from Madrid. Since we specifically chose the Flickr images without GPS information for the test set (to ensure disjoint train and test sets), we have manually geo-located all images from this set to carry out this visual experiment. Also, we added the geo-location for each of the 15 landmarks present on the test set

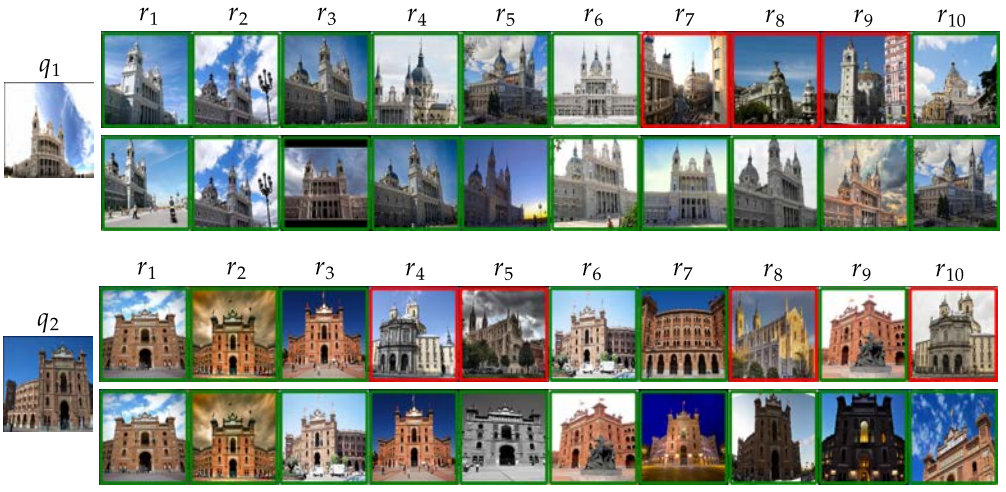


Figure 4.3: Two queries (q_1, q_2) from the Madrid test set that belong to landmarks 1 and 2 of Figure 4.2 respetively. The top-10 most similar images with respect to each query are shown using the baseline-CNN (top row for each query) and location-adapted-CNN (bottom row for each query). Green rectangles indicates relevant images.

(see Table 4.1). Figure 4.2 shows 12 out of these 15 ground truth landmarks using blue circles. The three missing landmarks were removed because of their large distance with respect to the others, which would have forced us to employ a level of zoom in the map inconvenient for visualization. To compute the GPS cluster centroids for the baseline ResNet50 model (orange squares in Figure 4.2) and our adapted MadridNet network (green circles in Figure 4.2) we took the clustering labels obtained in the quantitative experiment presented in Subsection 4.7.1, and computed the average GPS coordinates for each individual cluster.

An analysis of the map allow us to extract an interesting conclussion. The adapted model (MadridNet) is generating GPS centroids closer to the actual ground truth landmarks than the baseline (ResNet50) network. Take for instance landmarks 1 and 2 that are individually analyzed in Figure 4.3. For queries depicting the most common view of those landmarks, their top-10 most similar images for the baseline ResNet50 contain several instances belonging to other categories, while the adapted MadridNet correctly retrieved a clean top-10. When the GPS centroids are computed by averaging the geo-locations of all images belonging to the clusters, the mean is pulled away from its precise location by the incorrectly labeled instances. Hence, the proximity of the estimated GPS centroids to the actual landmarks can be understood as a qualitative measure of model visual accuracy. In that regard, our adapted model shows a superior performance.

Figure 4.4: Two queries (q_3, q_4) from the Rome test set that show the benefits of adapting a baseline-CNN to a particular city. The top-10 most similar images with respect to each query are shown using the baseline-CNN (top row for each query) and location-adapted-CNN (bottom row for each query). The figure shows the negative effect of common objects (people, cars) acting as distractors for the baseline-CNN, while the location-CNN successfully avoids the distractions and focus on the landmarks. Green rectangles indicates relevant images.

Beyond the city of Madrid, let us visually analyze the positive effect of fine-tunning general models using our approach. In Figure 4.4 we show the retrieval results for two queries belonging to Rome. The first query q_3 contains a picture depicting the Spanish steps landmark in the city of Rome. Interestingly, the photo was taken in black and white and, apart from the monument itself, the picture also contains a large amount of persons. The baseline model, which has been trained to focus on semantic categories, yields high similar feature vectors for other pictures showing people, leading to only 2 out of 10 correct images at the top-10. Instead, our location-adapted model has been trained to dismiss visual elements from the pictures that might be prominent (like the people in this case) but are not unique or discriminative of the landmark (people is likely to be found at any other monument). Query q_4 presents a similar challenge where a car takes up a large chunk of the picture foreground and the monument is relegated to a distant background. On the one hand, the baseline model focuses on the car retrieving mostly instances containing vehicles. This *distraction* leads to only 2 out of 10 correctly retrieved images. On the other hand, the RomeNet network is capable of finding other images of the same landmark and returns a perfect top-10 without any of the retrieved images containing cars.

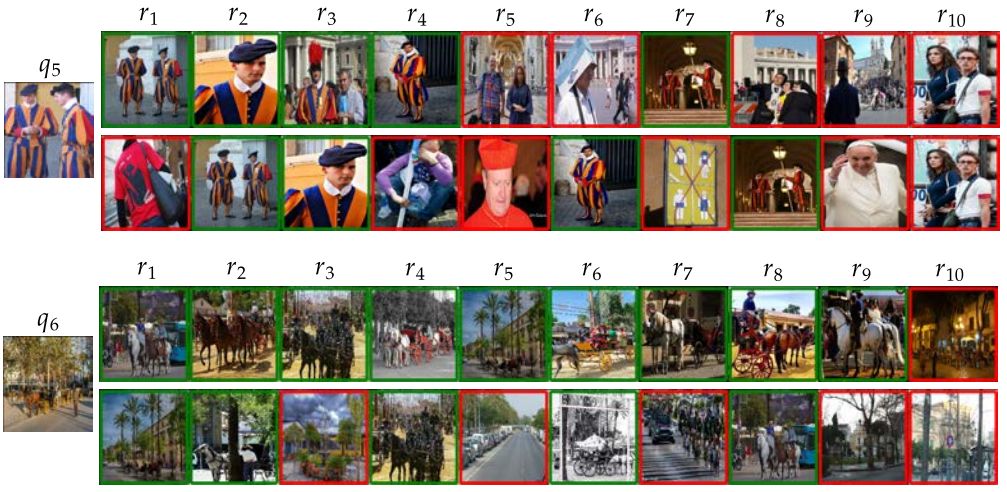


Figure 4.5: A query from Rome (q_5) and another from Jerez de la Frontera q_6 where the baseline model outperforms our proposed location-adapted networks (RomeNet and JerezNet respectively). Green rectangles indicates relevant images.

4.7.3 Error Analysis

The proposed location-CNNs have great advantages with respect to the general baseline networks. However, for the task of landmark discovery, they also present some limitations. Figure 4.5 contains two examples (q_5, q_6) where the semantic baseline model is capable of achieving better results that our location-adapted CNNs. The nature of the landmarks in a city is rarely semantic but, sometimes, this is exactly the case. Take, for instance, q_5 containing a picture of the pontifical swiss guard at the Vatican city. The baseline ResNet50 is capable to find 5 relevant instances with respect to the query, with a perfect top-4 retrieval. In contrast, RomeNet can only find 4 relevant pictures and only 2 of them are at the top-4. The cause for this behavior is that, in order to excel at most landmarks, RomeNet needs to learn to dismiss people (see q_3 from Figure 4.4). The side-effect of this learning policy is that it harms the performance when the object of interest is, in fact, people. The other main cause of under performance for our method are those landmarks that are hard to ascribe to a particular physical location. For instance, q_6 shows a typical touristic horse carriage from Jerez de la Frontera in Spain. The baseline-CNN is capable of finding 9 out of 10 relevant instances, while JerezNet only retrieves 5 out of 10. The problem is that, since the carriages move around the city, the GPS coordinates are not as reliable as in static landmarks, leading to mislabeled pairs that break the training process of the location-CNN. A similar problem can happen with landmarks that spread along large areas, such as zoos or race tracks, or big

landmarks that are typically photographed from very distant points, such as the Egypt pyramids or the Eiffel tower.

4.8 Ablation Studies

In this section, we study the effects of depriving our method from two crucial components. First, in Subsection 4.8.1 we measure the benefits of the soft labels by training our location-CNNs using other well known losses that employ binary labels. Then, in Subsection 4.8.2, we explore how the absence of landmarks in the training sets affect the capabilities of the location-CNNs to represent images from a particular city.

4.8.1 The effect of removing the soft labels

In this subsection, we study the influence of the soft-labels on the landmark discovery performance. For that purpose, we repeat the experiment described in Section 4.7, and compare the results achieved using our SM loss working with soft labels, with the contrastive (CT) and triplet (TL) losses, that employ hard binary ones. To that end, the binary labels are computed from the soft ones as follows:

$$y_b = \begin{cases} 0 & \text{if } y_i < 0.5 \\ 1 & \text{if } y_i \geq 0.5 \end{cases} \tag{4.4}$$

where y_i are the soft labels used by our SM loss and y_b are the binary ones necessary for the CT and TL losses. Table 6.2 contains the results achieved by the different losses for the three cities. It can be seen that the soft labels and our SM loss consistently achieve the best results for all test sets and clustering metrics. These results were somewhat expected given the noise-sensitive nature of the CT and TL losses. However, we can see that the CT loss is getting very competitive results for Rome. It is posible that the restrictive threshold used in equation 3.2 is fitting particularly well the Rome training set decreasing the amount of mislabeled pairs but, this might be just a coincidence since this behavior is not present for the other cities. Thus, we can conclude that using the soft labels is a critical step to consistently achieve good results in our task.

In addition, an unexpected finding is that the TL loss seems to be the mostly affected by the noise in the labels. Interestingly, this fact will be also revealed in the experiments of the second part of this thesis (see Subsection 6.5.2).

Table 4.3: Average and standard deviation of the Rand, Jaccard and Fowlkes indexes as a result of comparing the same model trained under three different loss functions: Triplet (TL), Contrastive (CT) and Soft-Matching (SM). Best results highlighted in **bold**

Test set	Model-Loss	Rand	Jaccard	Fowlkes
Jerez	JerezNet-TL	0.8467 $\pm$ 0.0181	0.1572 $\pm$ 0.0157	0.2711 $\pm$ 0.0302
	JerezNet-CT	0.8834 $\pm$ 0.0087	0.2797 $\pm$ 0.0111	0.4142 $\pm$ 0.0214
	JerezNet-SM (Ours)	**0.9242** $\pm$ **0.0201**	**0.3834** $\pm$ **0.0323**	**0.5627** $\pm$ **0.0254**
Madrid	MadridNet-TL	0.8942 $\pm$ 0.0274	0.4128 $\pm$ 0.0077	0.5974 $\pm$ 0.0275
	MadridNet-CT	0.9147 $\pm$ 0.0189	0.4390 $\pm$ 0.0170	0.6250 $\pm$ 0.0246
	MadridNet-SM (Ours)	**0.9547** $\pm$ **0.0189**	**0.5911** $\pm$ **0.0261**	**0.7551** $\pm$ **0.0257**
Rome	RomeNet-TL	0.9236 $\pm$ 0.0171	0.5209 $\pm$ 0.0154	0.6457 $\pm$ 0.0121
	RomeNet-CT	0.9248 $\pm$ 0.0323	0.5399 $\pm$ 0.0201	0.6488 $\pm$ 0.0201
	RomeNet-SM (Ours)	**0.9345** $\pm$ **0.0177**	**0.5473** $\pm$ **0.0138**	**0.6625** $\pm$ **0.0125**

4.8.2 The effect of unseen landmarks

As a second ablation study, we have tested the ability of the learned models to deal with unseen city landmarks, or even with images from other cities. In other words, we would like to check if our models are overfitting the previously seen data and would therefore perform badly on unseen scenes. This would become a significant weakness if we do not have geo-located images of a particular landmark of interest in our training set.

Table 4.4 shows the results achieved by JerezNet when tested in Madrid or Rome. It can be seen that, in this scenario, both the baseline and JerezNet achieve very similar performances. This is a very important result, as it proves

Table 4.4: Average and standard deviation of the Rand, Jaccard and Fowlkes indexes as a result of clustering images from Madrid and Rome using JerezNet. Best results highlighted in **bold**

Test set	Model	Rand	Jaccard	Fowlkes
Madrid	ResNet50	0.9254	0.3817	0.5467
		$\pm$ 0.0051	$\pm$ 0.0239	$\pm$ 0.0322
	JerezNet	**0.9287**	**0.3867**	**0.5522**
		$\pm$ **0.0153**	$\pm$ **0.0238**	$\pm$ **0.0301**
Rome	ResNet50	**0.9174**	**0.3728**	**0.5692**
		$\pm$ **0.0157**	$\pm$ **0.0298**	$\pm$ **0.0182**
	JerezNet	0.9105	0.3646	0.5525
		$\pm$ 0.0054	$\pm$ 0.0179	$\pm$ 0.0223

that our model adapts to the trained location, but does not over-fit on the training data. Consequently, it is not necessary to see all the interesting places from a city during training in order to deploy a useful model. Even for those unseen landmarks, the model would perform at least as well as a general purpose CNN.

4.9 Conclusions

In Chapters 3 and 4, we have proposed an assessed a novel training framework that relies on image content and metadata to learn location-adapted deep models that provide tuned image descriptors for specific visual contents. Our networks, which start from an initial model pretrained on a different task, are then fine-tunned by means of a custom pairwise loss function using weak labels that are computed from image content and meta-data.

Our experiments on a landmark discovery task show that the proposed location-CNNs achieve an improvement of up to a 55% over the baseline model (Jaccard index on Madrid test set). This implies that the network successfully learns the visual clues and peculiarities of the region of interest and generates image descriptors that are location-adapted. Besides, the location-CNNs are capable of dismissing visual information that might be prominent on an image if it is not specific to any particular landmark, thus, avoiding the influence of distractors decreasing the accuracy. In addition, ablation studies show that the use of weak labels is crucial to consistently outperform the baseline models, and that for those landmarks that were not present on the training set (or even images from

other cities), our proposed networks perform at least as well as the baseline CNN, which indicates a good resilience to overfitting.

Further work will explore research lines like other meta data and scenarios where specialized networks are necessary to outperform existing general models, mechanisms to update the fixed soft labels as training progresses, and new ways to incorporate the notion of soft labels into modern list-based loss function for deep retrieval.

Chapter 5

Training Deep Retrieval Models with Noisy Datasets: Bag Exponential Loss

In this chapter, we dive into the theory behind the Bag Exponential (BE) loss, the second of our contributions. Section 5.1 contains an introduction describing the problematic and main motivations that guide our work, while Section 5.2 includes several subsections that put forward the BE loss itself. Subsection 5.2.1 presents the analytical shape of our loss and relates it to previous existing functions on which we inspired our work. Subsection 5.2.1 introduces the implemented mechanism to deal with noise on the training sets that and points out to other possible applications than dealing with mislabeled samples for our method. Finally, Subsection 5.2.3 discusses efficiency aspect of our approach in comparison with other loss functions used in the literature.

5.1 Introduction

Instance image retrieval aims to find an element contained in a query in an unordered collection of images. It is different from class retrieval since the goal is not to find samples belonging to a certain category (e.g., buildings), but a concrete instance (e.g., the Big Ben). Image retrieval has been extensively studied by the community because it enables several important applications, such as place or product recognition, fingerprint or face identification, query by example and, in general, any task that benefits from transferring meta-data between related images [13].

Nowadays, deep learning and, in particular, Convolution Neural Networks (CNNs) are the state-of-the-art paradigm to tackle the instance image retrieval problem. The first clues that classification CNNs produced global features that were useful for retrieval were found in [2]. Later, the first CNN that was specifi-

cally tailored for image retrieval was proposed in [151]. This network combined a novel *siamese* architecture [66] with a rank-based loss, known as *contrastive loss* [19]. Two concurrent works [74][17] demonstrated that substituting the top fully connected layer of classification models by some pooling mechanisms over the convolutional activations turned out to be quite effective to improve performance. Beyond global image representations, other works have focused on computing deep local features which are more robust against clutter, occlusions and view point variations. In [85], the authors generate deep local descriptors (DELF) by coupling an attention model with a CNN in order to find keypoints and describe them in a single forward pass. More recently, some authors have proposed hybrid retrieval models that extract both local and global descriptors by combining state-of-the-art global CNNs with deep local networks [12]. In this chapter of the thesis, we focus on improving image global representations under noisy training datasets.

Since the very first attempts to tackle image retrieval problem using CNNs, it is clear that the performance of the developed solutions is strongly tied to the training datasets. In particular, training and test datasets must exhibit a high correlation in the type of objects or scenes in order to achieve an optimal performance [6]. This correlation allows models to learn the most effective visual patterns for the task at hand but, in exchange, requires building a training dataset for that task.

Generating training datasets for image retrieval is time-consuming and labor-intensive. There are several alternatives to carry out this task, but none is good enough for several reasons. The most straightforward method consists in defining the type of objects and use textual tags or other meta-data in search engines to retrieve a collection of potentially suitable images. This is perfectly feasible and can generate a large database with limited effort but, unfortunately, the resulting collection of pictures will very likely contain a considerable amount of noise (mostly due to labeling errors or imprecisions on the meta-data). Indeed, the presence of noise in the training datasets is known to hinder the learning process [6]. The second, and more common approach, extends the previous one by introducing (semi-)automatic algorithms to post-process the initial set of retrieved images, filtering out non-relevant samples and reducing the level of noise. Designing such methods is not straightforward and requires extensive engineering work and innovation [18][4]. Furthermore, these techniques need to manage a difficult trade-off: on the one hand, if the filtering is too restrictive, both the size of the database and its diversity will be dramatically reduced. Furthermore, the process will be particularly aggressive with those relevant samples that are less representative (e.g. hard positives). On the other hand, if the filtering is not restrictive enough, the resulting training set will still contain some degree of noise, which, even in small proportions, will significantly degrade the performance of current approaches [4]. A final concern regarding automatic

filtering methods is that the post-processing algorithms inevitably introduce a bias into the training dataset, which conditions the subsequent learning task.

In contrast to previous techniques dealing with the noise of the dataset, our approach handles noise during training, by introducing a novel loss function which is effective for training deep learning models using noisy datasets. More specifically, we employ a similar idea to the Multiple Instance Learning (MIL) framework. Since we can not rely on individual sample labels because of the noise, our loss uses bags of pairs of images from which we expect at least some of them to be true matches. The goal is to estimate the likelihood of each pair in the bag to be a true positive and weight their contribution to the loss proportionally. Since the weighting is done as the training progresses, the model can choose dynamically the importance of the different images. This eliminates potential biases found in post-processing approaches that filter the datasets before training. In the same way that deep models learn the best features to solve a task, we propose an automatic way to choose the samples of the training dataset from which learning will optimize the results.

In this chaper, we introduce a novel loss function that, inspired by the MIL framework and working with bags of matching images instead of single pairs, allows a dynamic weighing of the relevance of each sample as the training progresses. The proposed method greatly enhances the applicability of CNNs in real-world image retrieval tasks, given that the dataset cleaning step is the most labor-intensive task, which is made unnecessary by our system. Moreover, it opens the door to a new line of research in image retrieval based on automatic data weighting and selection. This chapter is devoted to a thorough descriptiosn of the method. Then, Chapter 6 describes a series of experiments that prove the efficacy of our approach.

5.2 The Bag Exponential Loss Function

In this part of the thesis, we seek to develop a loss function to train CNNs for image retrieval that is suitable to deal with noisy datasets. The proposed loss relies on two main contributions. First, we suggest to revisit the concept of *margin* typically found in previous losses to make it smoother and more tractable from a mathematical point of view. In particular, we propose to embed the margin concept into a negative exponential function, which smoothly vanishes, acting as a *soft margin*. Second, inspired by the MIL paradigm [152], we have developed a bag-based loss that allows us to weight each sample contribution and shields the learned models against noise. Figure 5.1 illustrates this concept by comparing how noise affects the computation of the gradients in traditional pair-based losses in contrast to how it affects the proposed loss. Our hypothesis is the following: assuming a training set with a certain percentage of noise

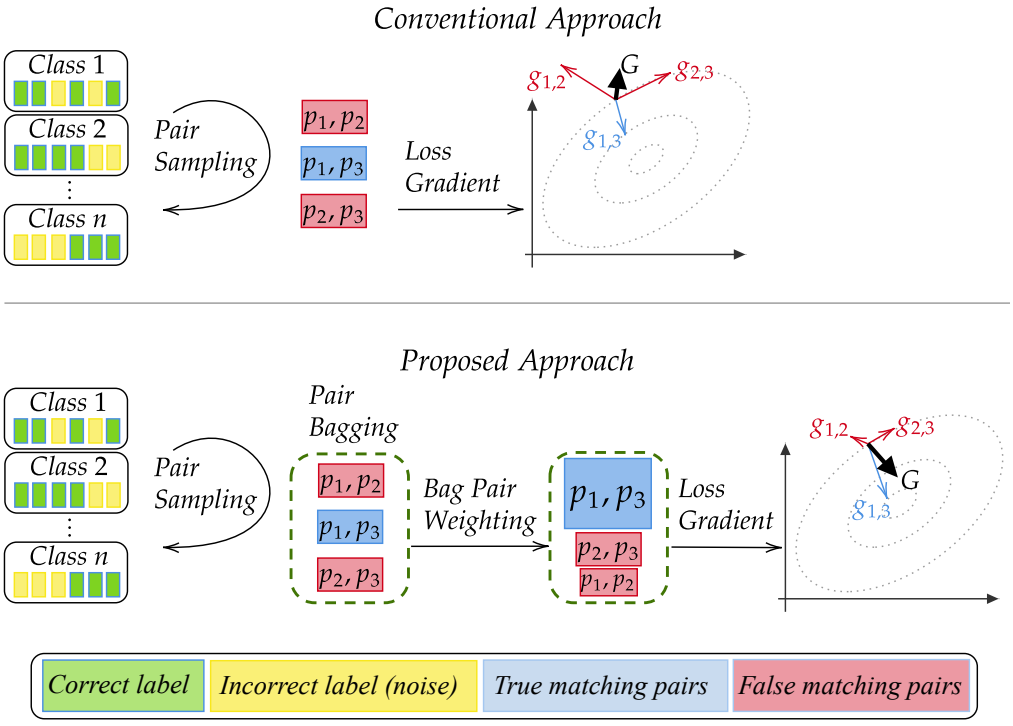


Figure 5.1: Conceptual differences between the proposed Bag Exponential loss and conventional pair-based approaches. For simplicity, only positive pairs (true or false) are considered. Our function involves a bag mechanism, inspired by MIL, that weights the relevance of the image pairs in order to generate better gradients when dealing with noisy training sets. In the figure, the relevance is represented by the size of the boxes containing the pairs, and the resulting accumulated batch gradient is represented by **G**

on each category, sampling a pair of images will yield either a true ((p_1, p_3) in Figure 5.1) or a false ((p_1, p_2) on Figure 5.1) match, causing instability during the training process. However, if we sample a large enough bag, although many of the samples may be noisy, there should be at least some relevant image pairs to learn from. Thus, as it can be seen in Figure 5.1, the goal of our system is to ensure that the noise-free pairs have the larger influence on the loss, becoming the ones responsible for the gradient directions in the backpropagation phase. In other words, we propose to go beyond instance-based retrieval losses by means of a MIL loss working with bags of images.

5.2.1 The Exponential Loss Function

For the sake of clarity, we will present the design of our loss function based on a well-known instance-based loss, the triplet loss [67]. Let $T_n = (\boldsymbol{x}_q, \boldsymbol{x}_p, \boldsymbol{x}_n)$

be a 3-tuple, where $\boldsymbol{x}_q$ is a query image, $\boldsymbol{x}_p$ is a potential positive instance with respect to $\boldsymbol{x}_q$, and $\boldsymbol{x}_n$ a potential negative. The objective of the triplet loss is typically stated as follows: for any given query, the Euclidean distance between the query and the positive image should be lower than that between the query and the negative image. Furthermore, in the original formulation a parameter was introduced to control the minimum acceptable difference between both distances (the margin), thus improving the model robustness. In our proposal, we have followed a similar principle, but we have required instead a minimum acceptable ratio between the distances from the query to negative and positive samples. In particular, this ratio should be larger than a predefined threshold α:

$$\frac{\|\boldsymbol{d}_{qn}\|}{\|\boldsymbol{d}_{qp}\|} > \alpha \tag{5.1}$$

where α somehow inherits the role of the margin in the triplet loss, preventing the model from generating relevant gradients when the corresponding sample has already been properly learned. Moreover, we have found that a suited way to accomplish the objective of the margin in a less abrupt manner was to embed it into a negative exponential function, leading to what we have named *exponential loss*:

$$L(T_n) = e^{-(\|\boldsymbol{d}_{qn}\| - \alpha\|\boldsymbol{d}_{qp}\|)} \tag{5.2}$$

where $\alpha \in \mathbb{R}_{\geq 1}$. The parameter α inside the negative exponential function naturally behaves as a *soft margin*, controlling how quickly the already well represented triplets start producing negligible gradients. Furthermore, given a sample, the exponential loss produces a gradient that is proportional to the model performance, instead of going to zero abruptly as it would happen when a hard margin is satisfied. Although we have experimentally found that soft margins provide slightly better results in our scenario, the proposed bag mechanism described in the following section might also be successfully incorporated into losses working with hard margins (see Appendix A).

5.2.2 The Bag Exponential (BE) Loss Function

Let us now consider the effect of dealing with a noisy training set, where there is a significant probability of randomly sampling a mislabeled *(query, positive)* image pair. Ideally, we would like the loss to prevent such pairs from affecting the training process. However, it is not feasible for the loss to discern, without any context, whether or not a pair of positive images is a true match. To provide some context to the loss, we propose the process illustrated in Figure 5.2, where

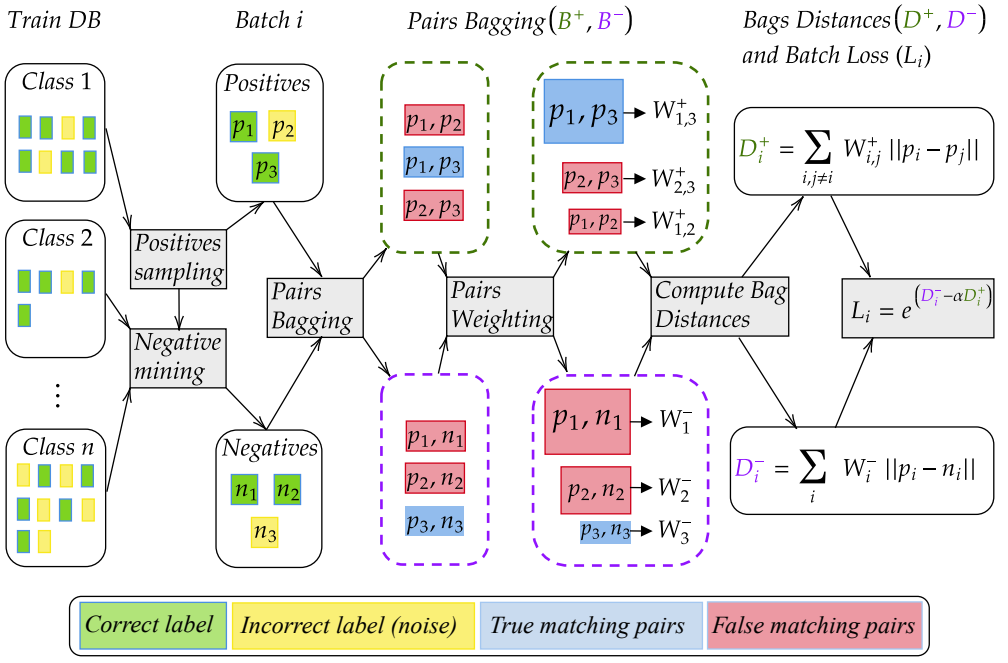


Figure 5.2: Illustration of the computation process of the loss for a mini batch of samples using the proposed Bag Exponential function. The different sizes of the boxes containing each pair inside the bags represent their relevance in the loss computation. Best viewed in color.

we have changed the typical structure of pair or triplet samples to bags of pairs. In particular, we suggest building a positive bag (B^+) of size $b(b-1)$ pairs, by sampling b queries from a category and generating all possible pairwise combinations. Likewise, we sample hard negatives, one for each of the b queries, and generate the bag of negative pairs (B^-), of size b, that is:

$$B^+ = \{\boldsymbol{p}_i, \boldsymbol{p}_j\}_{1 \leq i,j \leq b, j \neq i}$$
$$B^- = \{\boldsymbol{p}_i, \boldsymbol{n}_i\}_{1 \leq i \leq b}$$

where $\{\boldsymbol{p}_i\}_{1 \leq i \leq b}$ denote feature vectors representing images labeled as belonging to the same category; $\{\boldsymbol{n}_i\}_{1 \leq i \leq b}$ are typically images of other categories considered as hard negatives for their corresponding positives (i.e., showing small distances); and b might take values in the range from 2 (similar to the standard triplet) to the size of whole dataset. This formulation is asymmetric in terms of the size of positive and negative bags, i.e., we did not include all possible query-to-negative permutations because in our datasets for image retrieval there are no false negatives. However, the negative bag definition might be easily adapted to account for such a factor.

Considering bags instead of instances leads to extend the previous definition of the exponential loss. Taking inspiration from the MIL paradigm, we need to define the *aggregated* positive (D^+) and negative (D^-) distances computed over their corresponding bags B^+ and B^-, respectively, and reformulate our exponential loss (5.2) to produce what we call the *Bag Exponential (BE) loss*:

$$L(B^+, B^-) = e^{-(D^- - \alpha D^+)} \tag{5.3}$$

where the exponential now applies to the bag-aggregated distances. We have defined D^+ and D^- in such a way that the learning process is capable of dynamically assigning weights to the samples and, thus, governing their impact on the loss according to certain criterion. In particular, these aggregated distances take the following forms:

$$\begin{aligned} D^+ &= \sum_{i,j \neq i} w_{i,j}^+ \left\| \boldsymbol{p}_i - \boldsymbol{p}_j \right\| \\ D^- &= \sum_{i} w_i^- \left\| \boldsymbol{p}_i - \boldsymbol{n}_i \right\| \end{aligned} \tag{5.4}$$

where, according to our previous definitions of B^+ and B^-, we have considered $b(b-1)$ different pairs of positives to define D^+ and b *(positive, hard negative)* pairs to define D^-. The aggregated distances depend on two sets of weights: $w_{i,j}^+$ and w_i^-, one for each pair of positive images and the other associated with each negative sample, respectively. It is important to notice that the formulation is general enough to accommodate any proper definition of the weights, according to the purpose and the task at hand.

As the main purpose of this paper is to deal with noisy datasets we have defined the weights as follows:

$$\begin{aligned} w_{i,j}^+ &= \frac{e^{-\beta \left\| \boldsymbol{p}_i - \boldsymbol{p}_j \right\|}}{\sum_{i,j \neq i} e^{-\beta \left\| \boldsymbol{p}_i - \boldsymbol{p}_j \right\|}} \\ w_i^- &= \sum_{j \neq i} w_{i,j}^+ \end{aligned} \tag{5.5}$$

where $-\infty \leq \beta \leq \infty$ is a design parameter, each positive weight $w_{i,j}^+$ has been defined as a normalized similarity between the representations of the

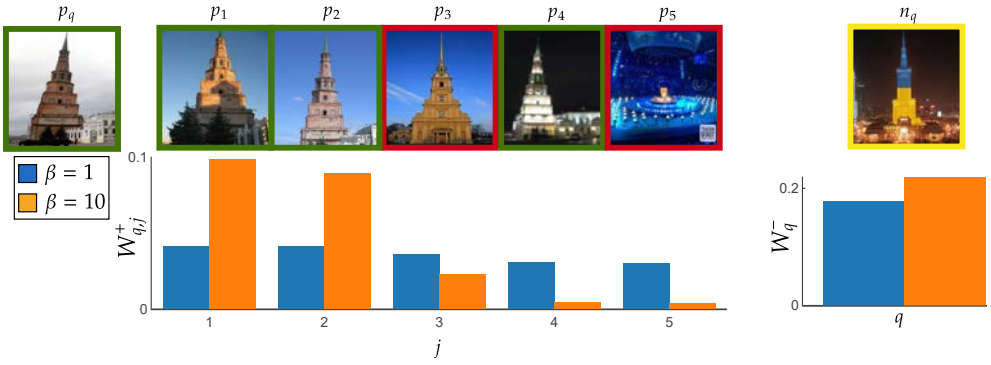


Figure 5.3: Illustration of the weight distribution resulting from eq. ((5.5)) for a positive p_q included on a bag of size 6. The set of weights for two different β values ($\beta = 1$ in blue and $\beta = 10$ in orange) are shown. Green frames denote true positives inside the bag, red frames denote false positives (noise) and the yellow frame is the true negative associated with the positive p_q. Best viewed in color.

respective images, $\boldsymbol{p}_i$ and $\boldsymbol{p}_j$, and each negative weight w_i^- is computed by accumulating all the positive weights involving the instance i, in order to balance positive and negative contributions to the loss.

It is worth discussing the role of the β parameter, as it determines the shape of the weight distribution and should be adapted to the intended task (e.g., by cross-validation). First, we focus on the positive range of potential values, which allows our bag exponential loss to be robust against noise.

We aim to deal with noise in the training dataset by assigning lower weights to those samples that are wrongly labeled (red frames in Figure 5.3). Assuming that two images forming a false positive pair will likely produce more distant representations in the feature space (i.e., large $\left\|\boldsymbol{p}_i - \boldsymbol{p}_j\right\|$ distance), and given that our exponential function generates gradients which are proportional to the distances between samples, false positive pairs would become dominant during training compared to true positive pairs.

The introduction of weights alleviates this problem, by reducing the contribution of false positive pairs to the aggregated distance D^+ and, consequently, to the loss and its gradients. In particular, small positive values of β provide a more uniform weight distribution, less dependent on the actual distances between positive pairs and, therefore, more suitable for low levels of noise ($\beta = 1$ in Figure 5.3). In contrast, large positive values of β produce a more sparse distribution of weights, concentrated on a few pairs containing the most similar images, which turns out to be more convenient for higher levels of noise ($\beta = 10$ in Figure 5.3). In the limit, when $\beta \to \infty$, our proposed aggregated weight distribution tends to an *indicator function* and the corresponding aggregated distance becomes a soft approximation of the *max operator*.

Whereas a similar approach could be applied to negative pairs as well, the

lack of false negatives in our datasets discourages it in our scenario. Instead, we have preferred to make negative weights dependent on the positive ones, and assign more weight to those non-relevant images that look similar to the dominant relevant ones.

In summary, large values of β help to drastically mitigate the effect of noise during training, but at the expense of reducing the influence of hard positive examples (i.e., image pairs of the same category that look more different and whose distances are generally larger than those of the rest of positive pairs). This effect is clearly visible in the case of $\beta = 10$ in Figure 5.3, where the fourth positive, p_4, is assigned a very small weight in contrast to what happens when $\beta = 1$. However, as we will show in the experimental section, the positive effect of the noise reduction clearly outweighs the negative effect derived from the lack of hard positives during training, and leads to optimal performance when dealing with noisy datasets.

It is also worth noting that this end-to-end approach for dealing with noise dynamically updates the weights as the network training progresses. Compared to preprocessing stages, the proposed solution is seamlessly integrated with the learning process, preventing unintended biases inherited from filtering stages, and favoring a more adequate convergence of the learning process, as we will show in the experimental section.

Beyond the noise-handling scenario, the loss function proposed in this section is general enough to tackle a wide range of applications, whenever weighting samples according to a certain criterion may be of interest. As an illustrative example of an alternative application, this approach could be used in noise-free scenarios to attribute a higher relevance to hard positives. To that end, we can use $\beta < 0$ in ((5.5)) and set uniform weights for negative samples w_i^- (to prevent hard positives from dominating the learning of negatives). We also illustrate the use of this specific configuration for the weights in the experimental section.

5.2.3 Efficiency Aspects

Since we need to simultaneously consider all the images in the bag to compute the weights in eq. (5.5), the memory requirement of our method is typically superior to those of simpler pair-based losses. In our experiments, we found our 12GB GPU to be unable to manage bag sizes larger than 10 when using the experimental setup described in Section 6.4 in combination with images in which the largest dimension is 1024 pixels. To circumvent this problem we used *multistaged backpropagation*[114] to reduce the training memory footprint. This method works by passing the mini batch through the network twice. During the first pass, in inference mode, all the features are computed and saved. During the second pass, in training mode, images are passed one at a time and the features from the first step are used to generate the corresponding gradients.

Finally, all the individual gradients are accumulated to carry out the parameter update for the complete mini-batch. This solution is equivalent to use a GPU with more memory and go through the complete mini-batch in one pass. Nevertheless, it is significantly slower since every image has to go through the model twice. In particular, we have measured the extra time incurred to train our state-of-the-art model using *multistaged backpropagation* (with smaller images that can fit in our GPU) and found a 20% overhead.

Chapter 6

Experiments on the Bag Exponential Loss

In this chapter, we present a set of experiments to test the proposed Bag Exponential (BE) loss introduced in Chapter 5. Section 6.1 contains a relation between our different claims and the parts of the chapter where they are proved, as well as an extended description of the structure of this chapter. Sections 6.2 to 6.4 introduce the datasets, compared losses and the experimental setup necessary to replicate our results. Section 6.5 is devoted to results comparing the noise robustness of different methods. Section 6.6 offers an state-of-the-art comparison for deep image retrieval. Section 6.7 test the effect of suppressing different parts of our method in an ablation study. A new approach to build image retrieval systems exploiting our approach is presented in Section 6.8. Finally, the chapter ends outlining the main conclusions achieved in Section 6.9.

6.1 Introduction

This part of the thesis explores the effects of training CNN models by means of the Bag Exponental (BE) loss presented in Chapter 5. In particular, the experimental results support the following claims: 1) the BE loss is more resilient to noise than other state-of-the-art retrieval loss functions (Sections 6.5.1 and 6.5.2); 2) the formulation of our loss is general enough to be applied with other purposes than dealing with noise, such as increasing the hard positives influence (Section 6.5.2); 3) the BE loss surpasses current state-of-the-art performance by allowing models to simultaneously choose the best visual features and samples from which to optimize (Section 6.6); 4) the most effective way to quickly deploy new retrieval CNNs for new domains is to employ our loss over a noisy training set (Section 6.8);

This chapter is organized as follows. In Section 6.2, we introduce three training sets, with various sizes and levels of noise, commonly used as bench-

marks for the state-of-the-art. Section 6.3 lists the five loss functions used in the experiments and provide a brief overview of them. Section 6.4 describes the experimental setup necessary to replicate our results. Section 6.5.1 presents an experiment that measures how several controlled levels of synthetic noise in the training data affect the performance of state-of-the-art loss functions in comparison with our Bag Exponential loss. Section 6.5.2 includes another set of experiments over the three previously described datasets commonly used as reference in the literature. Section 6.5.3 measures the influence of β from equation 5.5, a key parameter of our method. Section 6.6 shows how our loss can be combined with other complementary techniques for image retrieval to provide state-of-the-art results in various retrieval problems. Section 6.7 presents an ablation study discussing the influence of the bag size as a function of the estimated noise level in the training dataset. Section 6.8 presents a new approach to build image retrieval applications in new domains by exploiting the proposed method. Finally, Section 6.9 draws the main conclusions achieved for this second part of the thesis.

6.2 Training Datasets

Three landmark datasets have been used to fine-tune retrieval networks with various losses. Specifically, we have used the Retrieval SfM-120k (SfM) dataset [4], the Google landmarks (GL) dataset [18], and a partial subset of the Landmarks (L) dataset [6], containing those images that are still available online. The information describing them is summarized in Table 6.1. These datasets vary in size, number of categories, topic granularity and noise level. By topic granularity we refer to the instance diversity within a given topic. For example, the SfM training set has been built around a list of classical European building-like landmarks, while GL and L contain a larger variety of objects from a broader geographical coverage, such as statues, parks, paintings, natural landscapes, buildings, etc. By noise level we refer to the quantity of errors affecting the training labels, which we have coarsely discretized into three categories: low, medium and high. In particular, the SfM training set has undergone a strict automatic filtering process that successfully removed all noise; the L dataset has been generated by text-querying a search engine and accepting all returned images in bulk, if at least 20% of them look correct to a human; and the GL is an intermediate case where the filtering was not aggressive enough to remove all noise, but in exchange, it produced a larger database.

Beyond landmarks, Table 6.1 also includes the Top 1000 Paintings (T1KP) set, a paintings training set that will be discussed in Section 6.8.

Table 6.1: Summary of the training (up) and test (down) datasets used in the experiments. Topic granularity refers to the instance diversity within a given topic.

Name	*Size*	*#Training Categories*	*Noise*	*Topic Granularity*
SfM	90k	551	None	Low
GL	1.2M	12894	Low	High
L	130k	674	High	Medium
T1KP	90k	950	High	Low
Name	*Size*	*#Test queries*	*Distractors*	*Topic Granularity*
Oxf5k	5k	55	None	Low
ROxf5k	5k	70	None	Low
Oxf105k	105k	55	100k	High
Par6k	6k	55	None	Medium
RPar6k	6k	70	None	Medium
Par106k	106k	55	100k	High
Holidays	1491	500	None	High
50P	1782	1782	None	Low

6.3 Compared Losses

In our experiments, we have considered five loss functions to train deep models: two of them constitute the most well-known and broadly adopted losses (Contrastive [19] and Triplet [67]), other two currently hold the state-of-the-art performance for the benchmark datasets in the literature (Multi-similarity [99] and Quantized mAP [114]), and our proposed BE loss. In the next paragraphs, we provide a brief description of the considered losses:

- **Contrastive (CT)** [19]: one of the first losses used for this task. It is a pair-based function that receives two images and a binary label, and tries to bring matching pairs closer in the feature space, while pushing away non-matching ones, until a margin is met.

- **Triplet (TL)** [67]: another widely used loss for image retrieval. It uses an anchor, a positive and a negative image, and attempts to increase the relative distance between the anchor-negative pair and the anchor-positive one, up to a margin.

- **Multi-Similarity (MS)** [99]: this loss currently achieves state-of-the-art performance for the CUB200 [153] and the In-Shop clothes Retrieval [154] datasets used in the task of Fine-Grained Image Retrieval (FGIR). The function is a generalization of previous pair-based losses and, similarly to our proposal, weighs sample pairs to account for three different types of similarities, one related to the pair of images itself and other two related to their positive and negative neighbors.

- **Quantized mAP (mAPq)** [114]: a loss that holds the state-of-the-art performance for the task of instance image retrieval in Oxford [155], Paris [156], and their revisited versions [157]. It is a list-based loss that optimizes directly for a soft version of the Average Precision (AP) metric.

- **Bag Exponential (BE)**: the loss proposed in this paper. It relies on a soft margin by embedding the distances into an exponential function, and extends the previous pair-based losses to work with bags of instances, allowing the model to weigh the samples according to a predefined criterion. In our experiments we explore two configurations: one to deal with noisy datasets and another to pay more attention to hard-positives.

6.4 Experimental Setup

Unless explicitly stated otherwise, we have used the following configuration for all the experiments:

- **Model**: as retrieval CNN, we have used the Resnet101-GeM architecture [4], which is a popular choice on the image retrieval literature, and currently holds the state-of-the-art for several benchmarks we used in this paper [114]. It is composed of a Resnet101 [3] backbone (without the last average pooling and fully connected layers) followed by a generalized mean pooling and L2-normalization layers. The Resnet backbone has been pretrained on ImageNet [16].

- **Optimizer parameters**: as a baseline configuration, we use Adam [158] with momentum of 0.9, learning rate of $1\text{x}10^{-6}$, learning rate exponential decay of $\exp(-0.001e)$, for epoch e, and weight decay of $1\text{x}10^{-4}$.

- **Training parameters**: the models are trained for 100 epochs, each epoch consists of a full pass through 2000 training tuples, and each batch contains 5 tuples (400 model parameters updates per epoch). The final composition of each training tuple depends both on the loss function and dataset: 1) CT loss uses a query, a positive, and five negatives in all datasets; 2) TL loss considers a query, a positive and a single negative

for all sets; 3) MS, mAPq and our EB loss use b positives and b negatives, with $b = 3$ for SfM, $b = 10$ for GL and $b = 15$ for L. Images are resized to 362 pixels on their longer side, and standard color jitter with random gray scale conversions ($p = 0.1$) is used as data augmentation.

- **Parameters of the Losses**: CT uses a 0.85 margin and TL a 0.4 margin, as proposed by [4]. MS loss uses $\alpha = 3, \lambda = 1, \beta = 2$. We found these values to work better in our experiments than the ones proposed by the original authors in [99], probably due to the different nature of the problem ([99] was proposed for FGIR). The mAPq loss uses $M = 20$, which was the best in our experiments and the same suggested by the authors.

  Our BE loss uses $\alpha = 1.05$ in all experiments, which means that we aim to enforce that negative pairs produce distances at least a 5% larger than positive ones. The β parameter for the bag kernel depends on the noise present on the training dataset. In particular, unless indicated otherwise, we used $\beta = -1$ (enhancing hard positives) for the clean dataset (SfM), and $\beta = 10$ when working with noise (GL,L). Note that the optimal value for β is expected to be different for GL and L, since their noise levels differ. This can be seen in Section 6.5.3 where the effect of the β parameter value has been analyzed in detail for both GL and L.

  All the hyperparameters (those of our proposed method and those of the compared methods by other authors) have been selected by cross validation on a disjoint 30k image validation set that comes along with SfM training corpus.

- **Evaluation**: the considered loss functions are compared in terms of the mean average precision (mAP) on three classical retrieval test sets: Oxford [155] (**Oxf5k**), Paris [156] (**Par6k**) and Holidays [159] (**Holidays**). The revisited versions of Paris (**RPar6k**) and Oxford (**ROxf5k**) [157] have been also included in the experiments and, additionally, the 100k Flickr distractors provided with the Oxford dataset have been added to Oxford (**Oxf105k**) and Paris (**Par106k**). Oxford and Paris queries are cropped according to the given bounding boxes in order to comply with the standard protocol of evaluation proposed by the authors. For the Holidays dataset, we provide results with the original and rotated versions of the set. All test images have been scaled to have a larger size of 1024 pixels. A summary for all the test sets can be found in table 6.1

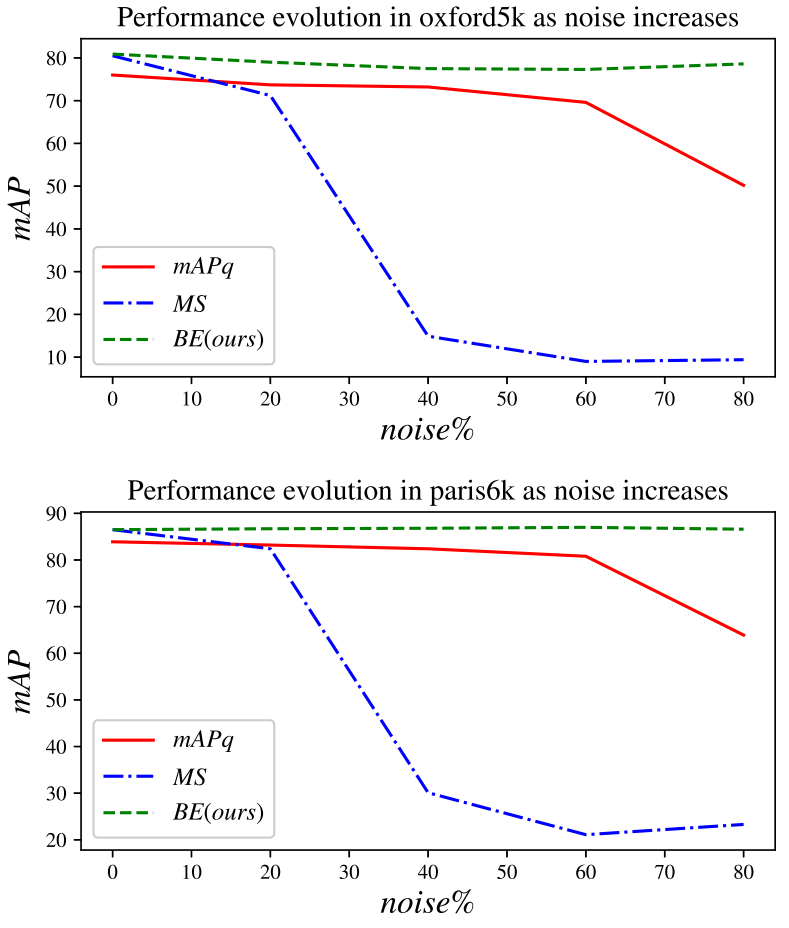


Figure 6.1: Noise robustness comparison of several state-of-the-art losses for *oxford5k* and *paris6k* datasets and a wide range of synthetic noise levels (0 to 80%).

6.5 Robustness to Noise

6.5.1 Training with Synthetic Noise Levels

Since we would like to assess how different losses behave when dealing with varying levels of noise on the training set, in this first experiment we aim to control such noise levels. Unfortunately, there are not enough datasets in the literature to cover a wide range of noise levels, and even if there were, manually labeling millions of images to quantify the actual noise levels would turn out to be unfeasible. Instead, we have artificially contaminated the noise-free SfM training set with several controlled levels of noise. Specifically, we define the noise level of a training set as the percentage of mislabeled instances in the corpus. We set a particular noise level for the complete dataset by enforcing it on

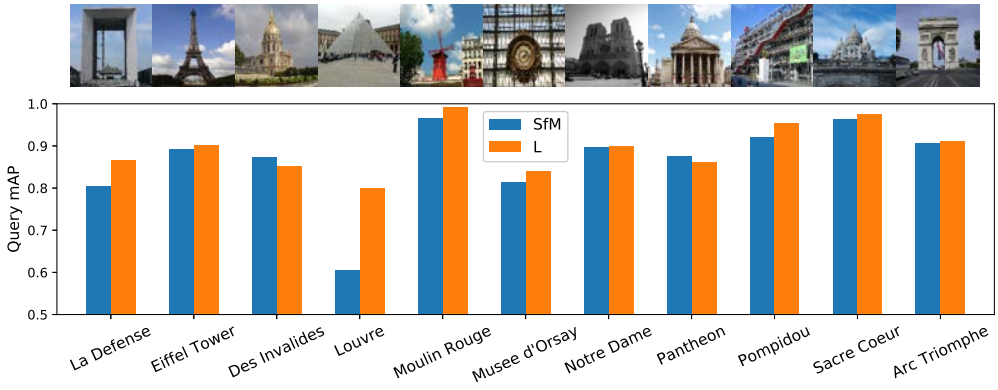


Figure 6.2: Differences in performance at the eleven queries of paris6k due to the use of distinct training sets: Retrieval-SfM-120k (SfM) and Landmarks (L). Non-building landmarks (La Defense, Louvre, Moulin Rouge, Pompidou) benefited from more diverse training sets containing landmarks other than building facades (L). Best viewed in Color.

each of its categories. Thus, all the categories have exactly the same noise level. To contaminate each category, images from others are copied into it until the target proportion of mislabeled samples is reached. Instead of using unrelated distractors as false positives, we used images from the same SfM dataset, which share the same general topic with the relevant samples.

Figure 6.1 shows the mean AP achieved in *oxford5k* and *paris6k* datasets by the compared losses for a wide range of noise levels. The training setup is the same described in Section 6.4 except for the number of epochs, which was halved to 50 (to save computation time), and the bag size, which was set to $b = 20$ (big enough for all noise levels). The BE β parameter has been set to $\beta = -1$ (focus on hard positives) for the 0% noise case, and $\beta = 10$ (noise awareness) for all other noise levels.

First of all, our proposed BE loss outperforms the other losses for every noise level, except for clean datasets ($noise\% = 0$), where is tied with MS and closely followed by mAPq. These results clearly reveal that while in noise-free environments all these losses are comparable, when the training set becomes noisier, the performance can degrade quickly, specially, for pair-based losses such as MS.

Second, the proposed BE loss is able to keep its performance level over a very wide range of noise intensities, something that does not happen with MS and mAPq, which degrade as the noise level increases. The bag mechanism of BE assigns small weights to the samples of the positive bag that are likely to be mislabeled (see eq. 5.5). In particular, if β is large enough, a single true positive pair of images can concentrate most of the weights distribution. In this experiment, we are using bags of size $b = 20$, and the highest noise level is

80%, thus, every bag contains on average 4 true positives. Obviously, the size of the bag is a very important parameter and will be discussed in detail in the ablation study of Section 6.7.

Finally, it is worth noting that the mAPq loss also exhibits a meritorious resilience to noise. In our opinion, this is due to its ranking-oriented nature. mAPq will push noisy samples until they are correctly ranked, hence avoiding the intrinsic need of pair-wise losses to minimize distances between all positives, which is really harmful in presence of mislabeled data. However, it can be seen how the proposed BE notably outperforms mAPq, specially when noise reaches high levels (e.g. 60%), where the performance of the latter suddenly degrades.

6.5.2 Training on Real Noise: The Reference Datasets

In this second experiment we compare performance achieved by the same network when it is trained using five different loss functions (see Appendix B for an additional loss function) and the three landmarks training sets introduced in Section 6.2, each one with a different qualitative noise level: none (SfM), low (GL) and high (L). To isolate the influence of the loss functions, no post-processing techniques (e.g. multiscale, whitening, query expansion, etc.) have been used. The parameterization of each loss and training set is the one described in Section 6.4. Table 6.2 summarizes the results.

As it can be observed, the proposed BE loss achieves the best performance in most of the train/test combinations: 20 out of 36. For datasets with higher levels of noise, such as GL and L, this might be somehow expected given the sample weighting mechanism embedded in the loss to deal with noise. However, BE is also very competitive compared to other losses when trained using the clean SfM dataset, which means that dealing with noise is not the only strength of our loss function. In particular, its ability to focus on hard positives in the absence of noise ($\beta = -1$) is remarkable too.

A very interesting result is the fact that the overall maximum scores in Par6k and Par106k are achieved by our method when trained on the L dataset, which contains the highest levels of noise. This result supports one of our main claims: it is possible to train highly effective CNNs for retrieval using noisy datasets (which are easy to generate), thus, increasing the applicability of CNNs for this task.

Compared to our loss, the TL performance is, in general, notably worse. In particular, it suffers a dramatic performance decrease when used in combination with noisy training datasets, leading to a very poor performance with GL, and even failing to converge to a useful solution with L. TL is, therefore, limited to be used in combination with noise-free training datasets, as SfM, and behaves particularly well when distractors are added (Oxf105k, Par106k). CT

Table 6.2: mAPs for nine Resnet101-GeM models [4] trained using three datasets and five different loss functions. No post-processing has been applied to the feature vectors. Holidays mAP evaluation contains both: unrotated/rotated* versions of the dataset. The "E","M" and "H" columns are the Easy, Medium, and Hard queries in the revisited versions of Oxford and Paris test sets. '—' means that the corresponding loss (either TL or MS) was unable to converge to a useful solution on the noisier L dataset. All figures were obtained using our own code, which has been made publicly available. The best results are highlighted in **bold**.

Train set	Loss function	Test set											
		Oxf 5k	Oxf 105k	Par 6k	Par 106k	Holidays	ROxf5k			RPar6k			
							E	M	H	E	M	H	
SfM	CT	80.6	76.3	85.5	77.2	82.1/85.4*	71.8	55.1	26.3	83.1	66.5	40.3	
	TL	80.2	**76.9**	85.3	80.2	86.0/89.5*	72.9	54.5	26.2	84.8	66.9	42.6	
	mAPq	76.0	73.0	83.9	78.6	**86.1/89.9***	69.2	49.5	20.2	84.0	65.5	38.9	
	MS	80.5	**76.9**	**86.5**	**80.5**	83.7/88.0*	**74.5**	55.0	24.3	**85.5**	**68.1**	**43.1**	
	BE(Ours)	**80.9**	76.6	**86.5**	80.3	83.3/87.3*	74.3	**56.6**	**30.0**	84.9	67.7	43.0	
GL	CT	77.5	74.9	87.1	82.4	**88.8**/91.8*	71.9	54.5	28.4	**86.9**	**71.9**	**49.2**	
	TL	74.5	72.0	86.5	82.0	88.1/**91.9***	70.4	50.9	22.6	85.6	68.8	45.4	
	mAPq	76.4	73.0	83.9	78.6	85.3/89.6*	69.5	52.5	24.9	83.5	66.0	40.5	
	MS	75.2	72.3	84.4	79.4	86.4/89.9*	69.5	50.9	22.4	84.0	68.0	43.2	
	BE(Ours)	**79.0**	**75.5**	**88.3**	**83.8**	85.3/89.1*	**73.1**	**55.1**	**29.4**	86.2	70.5	47.5	
L	CT	52.6	47.4	68.4	58.4	76.2/77.6*	42.7	30.5	8.1	66.2	50.2	23.9	
	TL	—	—	—	—	—	—	—	—	—	—	—	
	mAPq	70.6	66.2	87.4	82.7	84.2/**88.7***	61.5	45.8	18.9	83.9	**71.2**	**49.2**	
	MS	—	—	—	—	—	—	—	—	—	—	—	
	BE(Ours)	**77.0**	**72.9**	**89.5**	**85.0**	**84.5**/87.4*	**72.8**	**52.5**	**24.6**	**86.3**	70.2	47.2	

loss attains a similar performance to that of TL, although it is slightly more robust to moderated levels of noise, being competitive when trained on SfM and GL. Finally, MS loss is very successful when trained with SfM, getting the best scores in 6 out of 12 cases. However, its performance degrades quickly as noise appears, even at moderate levels, such as those in the GL training set. In contrast, mAPq seems to be competitive across a wide range of noise levels, and

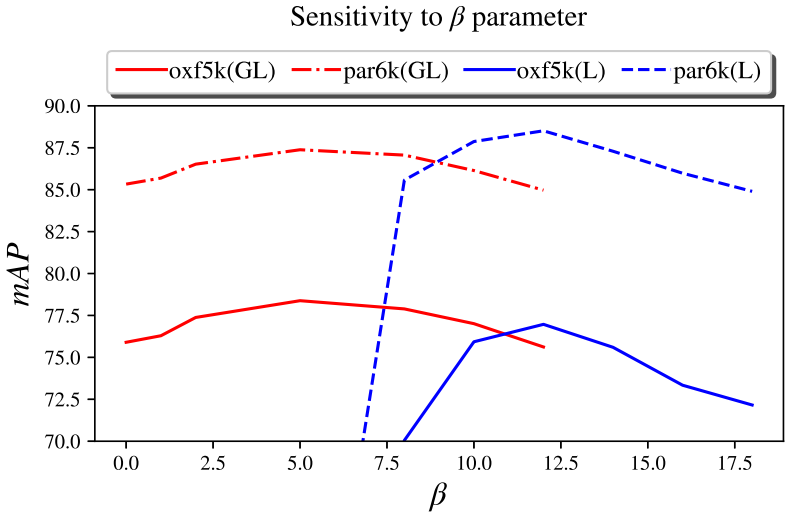


Figure 6.3: Evolution of performance in *oxford5k* and *paris6k* when a model is trained under GL and L for different values of the β parameter (eq 5.5) .

particularly outstanding when evaluated at the Holidays test set.

There are two loss functions, TL and MS, which were not able to converge to an appropriate local minimum for the L dataset. This dataset exhibits the highest noise level (we have estimated in a small subset of image pairs that the probability of sampling a false positive pair may be around 70%). Given such a high percentage of false positives, both TL and MS fail to converge because they are paying a proportional attention to them.

Finally, it is worth noticing that models trained with SfM consistently perform better in Oxford-related test sets, while higher scores are obtained in Paris and Holidays when models were trained with GL and L (even though they are noisy). In order to understand the rationale behind these results, let us refer to Figure 6.2, where we compare the test performance of each Paris query for models trained with both SfM and L. The Paris benchmark contains a broader topic granularity than Oxford, which is mainly composed by classic building-like structures, such as those present in the SfM dataset. In particular, note how training with L makes the model better at non building-like structures such as *La Defense*, *Louvre*, *Moulin Rouge*, or the *Pompidou Center*. In contrast, the SfM-trained model is superior for *Des Invalides* and *Pantheon*, which are the kind of building landmarks typically found in the SfM dataset. This deepens on the critical importance of the topic correlation required between train and test sets. Our experimental results suggest that optimal performance can be expected by generating training sets that are highly-correlated with the final task, even though if they show high levels of noise.

6.5.3 The influence of the Bag Exponential loss β parameter

Although these experiments could have been considered as a part of the ablation study (Section 6.7), we have included them in this subsection because of the direct influence of the β parameter on the robustness of the method against noise. In particular, we study the sensitivity of our BE loss to the value of the hyperparameter β (see eq. (5.5)). As it is shown in Figure 5.3, β controls the distribution of the weights associated with the image pairs in the bag. Figure 6.3 shows the performance achieved in *oxford5k* and *paris6k* when a model is trained on GL and L training sets using different values for β. The rest of the training setup is the same described in Section 6.4.

Let us first focus on the noisy L dataset. For $\beta < 8$ the performance quickly degrades. In this case, the distribution of weights is fairly uniform and the loss is giving a non-negligible weight to false positive pairs, which generates gradients that diverge from the optimal solution. As β increases, the weights start to concentrate on the portion of true positives within each bag, reducing the influence of the false pairs. However, if we keep growing the parameter (e.g. $\beta > 12.5$) the system starts to loose some performance. The rationale behind is the following: larger values for β increase the concentration of the weights on the most similar pairs (see eq. (5.5)), which in turn, reduce the effective size of the training set by disregarding most of the other images. This effect is amplified if we employ employ large bag sizes, losing diversity during training. In fact, if we take this situation to the extreme (e.g. $\beta \to \infty$ and a bag size that is large enough to accommodate all the samples on each training category), only the two most similar images per category will be considered during training. In consequence, a wrong combination of bag size and β value might be the main cause of poor performance.

Regarding the GL training dataset, small values of β are not a threat to find a suitable local minima. In fact, using $\beta = 0$, which uniformly distributes the weights to all the training pairs, provides a reasonable performance in this case. This can be also seen in the baseline comparison of losses provided in Table 6.2, where all losses perform reasonably well despite lacking any mechanism to deal with the moderate noise level of the GL set. Indeed, using small values of β also has a positive effect as it reduces influence of the false positives on each category. Finally, the behavior for large values of β is the same as that described for the L dataset.

6.5.4 Error Analysis

In this section we analyze some errors made by our method and reflect on the underlying reasons. Figure 6.4 compares the retrieval capabilities of the mAPq loss and our BE function for two illustrative queries: q_{27} from *rOxford5k* and

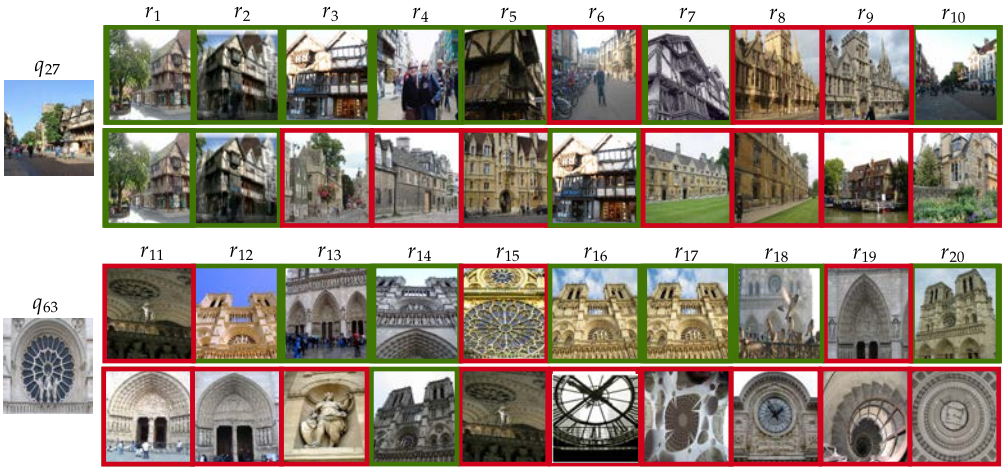


Figure 6.4: Comparison of the retrieved results for the query 27 of the *rOxford5k* dataset and the query 63 from *rParis6k*, using the mAPq loss (first row for each query) and our proposed BE loss (second row). For q_{27} the top ten returned results are shown, while for q_{63} eleventh to twentieth first ranked images are included. Green rectangles indicates relevant images.

q_{63} from *rParis6k*. The models used to generate the descriptors were trained using the GL dataset in Subsection 6.5.2.

Let us analyze q_{27} first. As can be seen, the relevant images for this query exhibit a high degree of view point variation. The mAPq loss handles well the change in perspective and ends up including seven relevant images in the top ten. In contrast, our BE loss only includes the most similar three. In the case of q_{63}, the relevant images present a high degree of scale variation instead. Since the top ten retrieved images for both losses were correct, we focus on the following ten. The mAPq retrieves seven relevant images showing a large scale variation, while our BE only includes one. These examples lead us to conclude that our method shows a preference to retrieve images presenting smaller variations in view point and scale. The reason lies in the way our loss is weighing the contributions of the different images pairs. In order to deal with noise, we used $\beta = 10$ to train the model. This makes the most similar pairs contribute more to the gradients than those presenting large variation in viewpoint and scale (hard positives). Thus, our method achieves a better overall performance by improving it robustness to noise in exchange for certain penalty on particularly hard cases.

Table 6.3: Performance comparison (mAP) of the same deep model trained on GL with three state-of-the-art losses in *oxford5k*, *paris6k* and their revisited versions. Best results highlighted in **bold**.

Loss	Test set			
	Oxf5k	Par6k	rOxf(M/H)	rPar(M/H)
mAPq†	88.1	93.1	66.3/**42.5**	80.2/60.8
MS*	89.0	94.2	66.8/41.4	79.9/60.6
BE(ours)*	**89.9**	**94.3**	**67.5**/41.8	**80.3/61.2**

† Performances provided by the original authors. * Performances computed using our own code.

6.6 Comparison with the State of the Art

In this section we show that our method, in a more competitive set-up, and combined with other extensions and post-processing techniques, can achieve state-of-the-art results in the reference datasets. Table 6.3 summarizes the performance of the same deep model [4] trained with the same database (GL), for three different state-of-the-art losses (mAPq, MS, and proposed BE). We use this model and GL as training set to replicate the same conditions used by the authors of mAPq. Since the results reported by the authors of MS were focused on datasets for fine-grained retrieval [99], we rely on our own implementation of the method.

The training setup for our method is the one described in Section 6.4, with the exceptions of β, which was reduced from 10 to 1.5, and b which was also reduced to 5. These parameters reductions are done to better adapt the method to the GL dataset, which presents only a small amount of noise. Additionally, the size of the longer side of the images during training was increased to 1024, which leads to a notable improvement in the performance at the expense of longer training times. As feature post-processing, we included multiscale and whitening from [4]. Since the scope of this paper is the loss functions, we did not report results for dimensionality reduction or query expansion techniques.

Even though the GL dataset has a low level of noise, results in Table 6.3 show that our BE loss is highly competitive and achieves the best performance in 5 out of 6 cases. This finding, in combination with those reported previously in Sections 6.5.1 and 6.5.2, leads to the conclusion that the proposed loss allows CNN-based retrieval systems to be trained with noisy training sets and achieve state-of-the-art performance. From our point of view, this represents a big leap in the applicability of this type of systems and helps to reduce the needed effort

to set-up new applications. Moreover, this result opens an emerging line of research for CNN-based image retrieval: let the models decide not only on the best features to solve the task, but also on the most relevant samples to do it [160].

6.7 Ablation Study

Among the parameters of the method, the size of the bag, b, yet deserves an in-depth discussion due to its relation to the noise level in the training dataset and its importance regarding the supporting hypothesis of our approach: there exists at least one true matching pair of positive images in each training bag. To perform this analysis, we have conducted an experiment similar to that of Section 6.5.1, but now focusing on our loss and comparing different bag sizes (including $b = 2$, i.e., no bag). Figure 6.5 shows the results achieved maintaining the same experimental setup of Section 6.5.1. Since the conclusions for both test sets (*oxford5k*, *paris6k*) are very similar, we will refer to them indistinctly throughout the discussion.

Let us first discuss the impact of removing the bag mechanism from the BE loss and focus on the $b = 2$ curves (positive and negative bags with one positive and one negative pair, respectively). If the training set is clean ($noise\% = 0$), the results are very good. In particular, we achieve $mAPs = 79.4/86.7$ for *oxford5k* and *paris6k*, respectively. However, as soon as the noise increases to 30%, the performance is severely reduced, and when noise reaches 50%, the network is unable to converge. The explanation is simple, if we randomly pick two images from categories with a 50% noise level, we get, on average, only one true positive per tuple, therefore, true matching pairs can not be formed, which leads to unreliable gradients and degeneration of the learning process.

For b values of 3 and higher, we are effectively implementing the bag mechanism. As can be seen in Figure 6.5, there is a relation that must be fulfilled between bag size and noise level for the network to properly converge. In particular, having an estimation of the noise level, $nl \in [0, 1]$, the bag size must satisfy $b \geq 2/(1 - nl)$. This selection ensures that, on average, the loss has access to at least a true matching pair of images to compute meaningful gradients. The largest noise level used in this experiment was 95%, which would have required a bag size of $b = 40$ to converge.

Finally, if the noise level could not be estimated, using a large size of the bag is always a safe alternative (at the cost of some extra-training time). As can be observed in Figure 6.5, the performance of the system for $b = 10$ is always competitive for any level of noise below 80%.

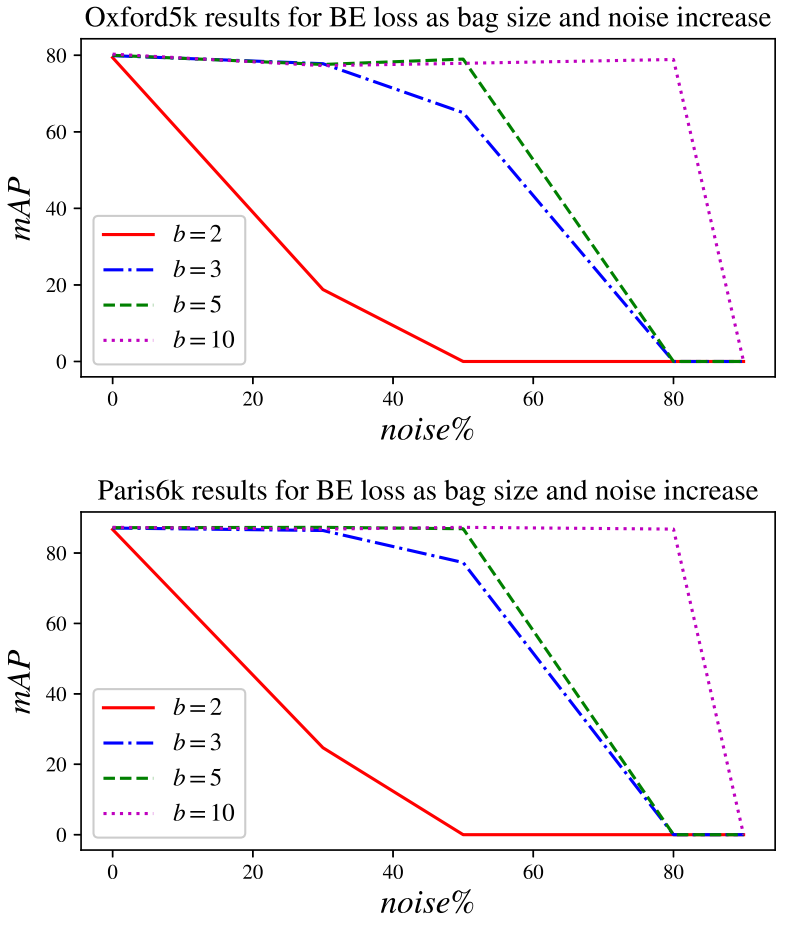


Figure 6.5: Influence of the bag size parameter, b, on the BE loss in relation with the noise level present on the training set.

6.8 A novel approach to build image retrieval applications in new domains

In this subsection we assess the potential of our approach to efficiently deploy retrieval applications in new domains for which labeled datasets are not available. To this purpose, we will only consider automatic solutions and avoid those that require manual annotations or ad hoc cleaning processes of the training data.

In particular, we have built the **Paintings Dataset (T1kP)** by simply taking the top thousand artworks of a public list *, using their names as text queries on an image search engine, and downloading the top hundred results returned

*http://es.most-famous-paintings.com/MostFamousPaintings.nsf/ListOfTop1000MostPopularPainting?OpenForm

Table 6.4: Retrieval performance (mAP) achieved by the same model (Resnet101-GeM [4]) on the 50P Painting Dataset when trained on datasets coming from different tasks (classification: ImageNet; retrieval: SfM, GL, T1kP), when trained on datasets for retrieval with varying levels of topic correlation between train and test sets (SfM, GL: Landmarks; T1kP, 50P: paintings), and when trained on retrieval datasets with different levels of noise (clean: ImageNet, SfM; noisy: GL, T1kP

Train set	Loss	50P Test set
ImageNet	CrossEntropy	65.33
SfM	mAPq	74.33
	BE (Ours)	75.70
GL	mAPq	72.77
	BE (Ours)	71.87
T1kP	mAPq	75.48
	BE (Ours)	**82.88**

per query. This has been done by means of an automated script that collected about 100k images distributed over the 1000 artwork categories. We have then randomly chosen 950 categories for the training dataset, which turns out to be quite noisy (including many irrelevant images for each category). This corpus has been generated in an analogous way to the Landmarks (L) [6] dataset, and exhibits a similar level of noise. In fact, we have replicated the exact same setup of Subsection 6.5.2 and swapped the L dataset for our new T1kP.

Moreover, in order to perform a fair comparison, we have generated a clean test dataset (**50P**). Specifically, we took the remaining 50 artworks and manually removed those images that were irrelevant given the category. This led to a total of 1782 images, all of which were used as queries in our experiments (~ 35 relevant images per query).

To assess our approach, we have compared three systems:

a) A classification CNN trained on a wide variety of topics (i.e., ImageNet).

b) A retrieval CNN trained on clean but topic-unrelated datasets (i.e., SfM and GL).

c) Our proposal, a retrieval CNN trained on a topic-related but noisy dataset (i.e., T1kP).

Table 6.4 shows mAP performances of the compared systems. Additionally, we have included the results for mAPq and our BE losses, to explicitly assess the

influence of the loss function in the process (other alternatives were not able to converge, analogously to what happened in Subsection 6.5.2 on the L dataset). The model trained for classification using the ImageNet dataset is the worst performing alternative. Hence, as expected, the alignment between training and test tasks is a key factor. Models trained for retrieval, but on topic-unrelated datasets show varying performance depending on the loss function: if the loss is not designed to account for high levels of noise, such as the mAPq, it is safer and almost equally effective to use a clean dataset (SfM) even if it does not share the topic with the target application. However, if the loss successfully handles the presence of noise (as our proposed BE loss), models clearly benefit from being trained on topic-related datasets, and lead to the optimal solution. Hence, we can conclude that the combination of an automatic approach to build noisy but topic-related datasets and our robust-to-noise BE loss function becomes an efficient and accurate solution to address image retrieval in new domains with no available labeled data.

6.9 Conclusions

In this chapter, inspired by the Multiple Instance Learning framework, we introduced a Bag Exponential Loss function to train CNNs for image retrieval. The loss combines the use of an exponential acting as a soft margin and a MIL-based approach working with bags of positive and negative pairs of images. We have thoroughly compared this loss function with two widely adopted losses, as well as two current state-of-the-art references. Our experimental results show its superior performance for both clean and, specially, noisy datasets. The rationale behind the achieved improvement in the noisy cases is that we are handling noise in an end-to-end manner and, therefore, avoiding its negative influence as well as the unintentional biases due to fixed pre-processing cleaning procedures. In addition, our method is general enough to suit other scenarios requiring different weights for the training instances (e.g. boosting the influence of hard positives during training). The proposed bag exponential function can bee seen as a back door to guide the learning process according to certain objective in a end-to-end manner, allowing the model to approach such an objective smoothly and progressively.

Our results also show that our loss allows CNN-based retrieval systems to be trained with noisy training sets and achieve state-of-the-art performance. Furthermore, we have found that it is better to use ad-hoc training sets that are highly correlated with the final task, even if they are noisy, than training with a clean set that is only weakly related. From our point of view, these results represent a big leap in the applicability of retrieval systems and help to reduce the needed effort to set-up new applications and scenarios: e.g. by allowing a fast

automatic generation of noisy training datasets and then using our bag exponential loss to deal with noise. Moreover, we also consider that this result opens a new line of research for CNN-based image retrieval: let the models decide not only on the best features to solve the task, but also on the most relevant samples to do it [160].

Future lines of work include the design of more complex schemes to generate the weights inside the loss, such as certain connectivity patterns among the different samples of the bag, kernels that saturate at very small or large similarities, and adapting the weights to work with false negatives in scenarios where they can be a problem.

Chapter 7

Conclusions and Future Lines of Work

7.1 Conclusions

This thesis has tackled the problem of Content Based Image Retrieval. We have presented two loss functions to train Convolutional Neural Networks that generate global image descriptors suitable for retrieval. The main novelty of our methods with respect to the previous work, is that the proposed functions include mechanisms to handle the noise in the training datasets explicitly, instead of taking the traditional approach consisting in a pre-cleaning process using add hoc complex methods. Training models with noisy datasets is desirable for two main reasons: 1) It avoids the bias induced by the pre-processing methods that try to find and filter noise; 2) It allows to quickly deploy CNN-based solutions for retrieval in new domains without the need of complex pipelines to generate the training sets.

In Chapter 3 we introduced a Soft-Matching (SM) loss that is capable of adapting general CNNs to new domains using noisy training sets, that can be collected without human intervention. In particular, image content and meta data are jointly exploited to infer soft labels to fine tune general models. The approach is developed upon a particular type of metadata, image geo-locations, although the philosophy of the method is extensible to other types of weak annotations, as discussed in Section 3.2.4. In Chapter 4, the function is proved to successfully specialize CNNs to particular cities or regions without incurring in overfitting. However, the method presents two limitations. First, it needs some form of weak supervision to limit the presence of false positives. Second, the method estimates the soft labels using global features computed using an auxiliary general model and fix them during training. Thus, the proficiency of the general models is therefore a limiting factor for the effectiveness of our specialized CNNs.

Chapter 4 explores the effects of fine tunning CNN models by means of the SM loss presented in Chapter 3. For that purpose, a landmark discovery system has been developed and a performance comparison has been carried out between global features computed with a general model and our own specialized CNNs trained with the proposed SM loss. The experimental results allow us to draw the following conclusions:

1. The proposed location-CNNs achieve an improvement of up to a 55% over the baseline model on the landmark discovery task. This implies that the networks have successfully learned the visual clues and peculiarities of the regions for which they have been trained, and generated image descriptors that are better location-adapted.

2. The SM loss allows models to dismiss visual information that might be prominent on an image if it is not specific to any particular landmark. Thus, it avoids the influence of distractors that might decrease the accuracy.

3. For those landmarks that were not present on the training set or images belonging to other cities, our proposed models performed at least as well as the baseline networks, which indicates a good resilience to overfitting.

4. The use of soft labels is crucial to consistently outperform the baseline models which showed notably worse performance using other losses with binary annotations.

Chapter 5 presents the second contribution of this thesis, the Bag Exponential Loss (BE). The BE loss is inspired by the Multiple Instance Learning (MIL) framework and works with bags of matching images instead of single pairs. This allows a dynamic weighting of each sample as the training progresses. The proposed method greatly enhances the applicability of CNNs in real-world image retrieval, given that the dataset cleaning step is the most labor-intensive task, and our method made it unnecessary. Moreover, allowing the CNN to handle noise in an end-to-end manner, eliminates potential biases found in post-processing approaches that filter the datasets before training. In the same way that deep models learn the best features to solve a task, we propose an automatic way to choose the samples of the training dataset from which learning will optimize the results.

Chapter 6 explores the effects of training CNN models for retrieval by means of the BE loss presented in Chapter 5. In particular, the experimental results allow us to conclude that:

1. The BE loss is more robust to noise than other state-of-the-art retrieval functions. This has been proved for both synthetic and real noisy datasets, and for two different topics: landmarks and paintings.

2. The formulation of our loss is general enough to be applied with other purposes than dealing with noise, such as increasing the hard positives influence.

3. The BE loss surpasses current state-of-the-art performance by allowing models to simultaneously choose the best visual features and samples from which to optimize, and opens the door to a disruptive line of research: learning with automatic sample selection.

4. The most efficient way to deploy retrieval CNNs into new domains is to employ our loss over automatically generated datasets, even if they are noisy. This notably increases the applicability of deep retrieval since generating and cleaning the training sets is the most labor intensive task for this technology.

7.2 Futures Lines of Research

In this final section of the manuscript, we discuss possible future lines of research related to our contributions.

First, the proposed SM Loss has been used in this thesis to adapt CNNs to particular cities or regions. The retrieval models acquire this location-oriented knowledge by finding out which specific fine-grained architectural patterns differentiate particular objects of interest (a specific building) from the rest in the city. In other words, the models are encouraged to dismiss any visual features that are common across different landmarks in the corpus. Thus, a natural extension to this work is to modify the proposed loss function in order to employ different sources of metadata to gain this ability in other domains.

One of the main weaknesses of the SM loss is the need for a general model that allows the initial computation of the weak labels. Consequently, the final performance of our method is somehow limited by the proficiency of that initial model. The development of some technique to allow updating the initial labels as training progresses and the model becomes better at the task is another important field of study based on our work.

Second, the Bag Exponential Loss (BE) is exploited in this manuscript to train deep retrieval models under noisy datasets. The hypothesis that supports our method is the following: assuming a training set with a certain percentage of noise on each category, sampling a pair of images from a training category will yield either a true or a false match, causing instability during the training process. However, if we sample a large enough bag, although many of the samples may be noisy, there should be at least some relevant image pairs to learn from. The way in which the loss decides how much weight give to each sample, is by using a similarity kernel based on euclidean distances.

The principal future line of research that we identify is the design of more complex schemes to generate the weights inside the loss, such as certain connectivity patterns among the different samples of the bag, kernels that saturate at very small or large similarities, or weights that account for false negatives in scenarios where they can be a problem.

Another interesting research line could lead to a transformation of our loss from a pair-based to a list-based. List-based losses [114] are becoming more popular because they optimize for a metric more closely related to the final task than the pair-based alternatives. Finding a way to include the weighting mechanism inside such losses could improve the CNN performance even further.

Appendices

Appendix A

The Influence of the Alpha Ratio in the Bag Exponential Loss

In our Bag Exponential (BE) loss (Chapter 5) an α ratio (equation 5.1) is used instead of a common absolute margin. The ratio was introduced in preliminary experiments because we found that it provided slightly better results than a margin based on absolute distances. However, our MIL-based approach can be successfully combined with a standard triplet using L2-normalized features. In this Appendix, we present results including bag-based distances into the standard triplet loss function. These results are summarized in Table A.1. In particular, we use the Google-Landmarks (GL) training set and compared our proposed loss (BE-alpha-ratio), the standard triplet with bag-based distances (Triplet-with-bags) and the standard triplet (Triplet). As it can be seen, although the results of the triplet with bags are good, the proposed ratio provides a slight advantage over the standard margin.

Table A.1: mAP results for the same CNNs trained under the Google-Landmarks dataset (see Section 6.2) employing three different loss functions: the standard triplet (Triplet), a triplet working over bag distances computed with our MIL approach but using common absolute margins (Triplet-with-bags) and, the proposed BE loss (BE-alpha-ratio) using an α ratio as margin (see Subsection 5.2.1). Best results in **bold**.

Training set	Loss	Test set	
		Oxford5k	Paris6k
Google-Landmarks	Triplet	74.5	86.5
	Triplet-with-bags	78.7	87.5
	BE-alpha-ratio	**79.0**	**88.3**

Appendix B

Comparing the Results for the SNCA Loss

In the comparative performance study presented on Subsection 6.5.2, pair-based as well as list-based losses are included. However, no center-based approach is covered. We tried to include a function from such category, the Scalable Neighborhood Component Analysis (SNCA) [110]. However, we faced some difficulties because of the large models that we use and the high dimensionality of the feature vectors in our problem, in contrast to those found in the problems addressed by the original SNCA authors. In our baseline, we used resnet101 with 2048-dimensional feature vectors, while the original work for the SNCA loss combine resnet18 and resnet50 with dimensionality reduction techniques to end up with 128-dimensional features. In particular, the memory footprint for SNCA-related methods is too high for Google-Landmarks dataset (it is far from fitting in our 12GB Nvidia Titan GPU). Although it fits for Landmarks dataset, it fails to converge (as well as almost any other loss) due to the large amount of noise in this dataset. We were able to train the model using the SfM dataset and the results included in Table B.1 were obtained after a thorough selection of the three hyperparameters of the SNCA loss (temperature, memory-momentum, and margin).

As it can be observed, SCNA is rather competitive in paris6k, but not in oxford5k. In our honest opinion, it is difficult to establish a fair comparison with SNCA using SfM as training set. While SNCA was initially designed to address a classification problem, SfM dataset was created to support image retrieval. Hence, SfM does not have categories as such and, although we did our best to group all related images into the same category, we found that there are thousands of categories with only 3 or 4 images, which does not fit well with the datasets used in the original SNCA work. We believe that an adaptation of the SNCA method to better fit these datasets is feasible, but it falls out of the scope of this thesis.

Table B.1: mAP results for the same CNNs trained with the SfM dataset employing different loss functions. See Section 6.2 for more information on the SfM dataset and Subsection 6.5.2 for the complete results without the SNCA loss. Best results in **bold**.

Training set	Loss	Test set	
		Oxford5k	Paris6k
SfM	Contrastive	80.6	85.5
	Triplet	80.2	85.3
	mAPq	76.0	83.9
	MS	80.5	**86.5**
	SNCA	72.8	84.0
	BE (Ours)	**80.9**	**86.5**

Bibliography

[1] S. Papert, "The summer vision project," 1966.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25* (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), pp. 1097–1105, Curran Associates, Inc., 2012. , 4, 18, 25, 48

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. , 5, 61

[4] F. Radenovic, G. Tolias, and O. Chum, "Fine-tuning CNN image retrieval with no human annotation," *CoRR*, vol. abs/1711.02512, 2017. , 6, 23, 49, 59, 61, 62, 66, 70, 73

[5] T. Martínez-Cortés, I. González-Díaz, and F. Díaz-de-María, "Automatic learning of image representations combining content and metadata," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, pp. 1972–1976, Oct 2018.

[6] A. Babenko, A. Slesarev, A. Chigorin, and V. S. Lempitsky, "Neural codes for image retrieval," *CoRR*, vol. abs/1404.1777, 2014. , 6, 18, 23, 26, 49, 59, 73

[7] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, *Convolutional Deep Belief Networks for Scalable Unsupervised Learning of Hierarchical Representations*, p. 609616. New York, NY, USA: Association for Computing Machinery, 2009. , 5

[8] H. Bannour, *Building and Using Knowledge Models for Semantic Image Annotation*. Theses, Ecole Centrale Paris, Feb. 2013. , 12

[9] C. Harris and M. Stephens, "A combined corner and edge detector," in *In Proc. of Fourth Alvey Vision Conference*, pp. 147–151, 1988. , 13

[10] Y.-G. Jiang, J. Yang, C.-W. Ngo, and A. Hauptmann, "Representations of keypoint-based semantic concept detection: A comprehensive study," *Multimedia, IEEE Transactions on*, vol. 12, pp. 42 – 53, 02 2010. , 16

[11] L. Liu, J. Chen, P. Fieguth, G. Zhao, R. Chellappa, and M. Pietikinen, "A survey of recent advances in texture representation," 01 2018. , 17

[12] B. Cao, A. Araujo, and J. Sim, "Unifying deep local and global features for image search," *arXiv*, pp. arXiv–2001, 2020. , 20, 21, 49

[13] A. Torralba, R. Fergus, and W. T. Freeman, "80 million tiny images: A large data set for nonparametric object and scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, pp. 1958–1970, Nov 2008. 4, 48

[14] A. Karpathy and F. Li, "Deep visual-semantic alignments for generating image descriptions," *CoRR*, vol. abs/1412.2306, 2014. 4

[15] M. Jones and P. Viola, "Fast multi-view face detection," *Mitsubishi Electric Research Lab TR-20003-96*, vol. 3, no. 14, p. 2, 2003. 4

[16] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and F. Li, "Imagenet large scale visual recognition challenge," *CoRR*, vol. abs/1409.0575, 2014. 5, 26, 38, 61

[17] A. Gordo, J. Almazán, J. Revaud, and D. Larlus, "Deep image retrieval: Learning global representations for image search," *CoRR*, vol. abs/1604.01325, 2016. 6, 49

[18] H. Noh, A. Araujo, J. Sim, and B. Han, "Image retrieval with deep local features and attention-based keypoints," *CoRR*, vol. abs/1612.06321, 2016. 6, 49, 59

[19] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2, pp. 1735–1742, 2006. 7, 22, 28, 32, 36, 49, 60

[20] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 1349–1380, Dec 2000. 10

[21] L. Zheng, Y. Yang, and Q. Tian, "Sift meets cnn: A decade survey of instance retrieval," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 5, pp. 1224–1244, 2017. 10

[22] Yihong Gong, Hongjiang Zhang, Chuan, and Sakauchi, "An image database system with content capturing and fast image indexing abilities," in *1994 Proceedings of IEEE International Conference on Multimedia Computing and Systems*, pp. 121–130, 1994. 10

[23] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Qian Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker, "Query by image and video content: the qbic system," *Computer*, vol. 28, no. 9, pp. 23–32, 1995. 10, 11, 12

[24] B. M. Scassellati, S. Alexopoulos, and M. D. Flickner, "Retrieving images by 2D shape: a comparison of computation methods with human perceptual judgments," in *Storage and Retrieval for Image and Video Databases II* (C. W. Niblack and R. C. Jain, eds.), vol. 2185, pp. 2 – 14, International Society for Optics and Photonics, SPIE, 1994. 11

[25] D. Mumford, "Mathematical theories of shape: do they model perception?," *Geometric Methods in Computer Vision*, vol. 1570, 09 1991. 11

[26] R. Schettini, "Multicolored object recognition and location," *Pattern Recognition Letters*, vol. 15, no. 11, pp. 1089 – 1097, 1994. 11

[27] E. Rivlin and I. Weiss, "Local invariants for recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 3, pp. 226–238, 1995. 11

[28] L. J. Guibas, B. Rogoff, and C. Tomasi, "Fixed-window image descriptors for image retrieval," in *Storage and Retrieval for Image and Video Databases III* (W. Niblack and R. C. Jain, eds.), vol. 2420, pp. 352 – 362, International Society for Optics and Photonics, SPIE, 1995. 11

[29] D. Slater and G. Healey, "The illumination-invariant recognition of 3d objects using local color invariants," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, pp. 206–210, Feb 1996. 11

[30] T. Gevers and A. W. Smeulders, "Content-based image retrieval by viewpoint-invariant color indexing," *Image and Vision Computing*, vol. 17, no. 7, pp. 475 – 488, 1999. 11

[31] A. Pentland, R. W. Picard, and S. Sclaroff, "Photobook: Content-based manipulation of image databases," *International Journal of Computer Vision*, vol. 18, pp. 233–254, Jun 1996. 11

[32] M. J. Swain and D. H. Ballard, "Color indexing," *International Journal of Computer Vision*, vol. 7, pp. 11–32, Nov 1991. 12

[33] M. A. Stricker and M. Orengo, "Similarity of color images," in *Storage and Retrieval for Image and Video Databases III* (W. Niblack and R. C. Jain, eds.), vol. 2420, pp. 381 – 392, International Society for Optics and Photonics, SPIE, 1995. 12

[34] B. Moghaddam, W. Wahid, and A. Pentland, "Beyond eigenfaces: probabilistic matching for face recognition," in *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 30–35, 1998. 12

[35] R. C. Veltkamp and M. Hagedoorn, "State of the art in shape matching," in *Principles of visual information retrieval*, pp. 87–119, Springer, 2001. 12

[36] K. Ohba and K. Ikeuchi, "Detectability, uniqueness, and reliability of eigen windows for stable verification of partially occluded objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 1043–1047, Sep. 1997. 12

[37] R. Deriche and G. Giraudon, "A computational approach for corner and vertex detection," *International Journal of Computer Vision*, vol. 10, pp. 101–124, Apr 1993. 13

[38] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms." `http://www.vlfeat.org/`, 2008. 14

[39] H. P. Moravec, *Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover*. PhD thesis, Stanford, CA, USA, 1980. AAI8024717. 13

[40] C. Schmid and R. Mohr, "Local grayvalue invariants for image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 530–535, May 1997. 13

[41] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2*, ICCV '99, (USA), p. 1150, IEEE Computer Society, 1999. 13, 15

[42] K. Mikolajczyk and C. Schmid, "An affine invariant interest point detector," in *Computer Vision — ECCV 2002* (A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, eds.), (Berlin, Heidelberg), pp. 128–142, Springer Berlin Heidelberg, 2002. 14, 15

[43] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *European conference on computer vision*, pp. 430–443, Springer, 2006. 14

[44] E. Mair, G. D. Hager, D. Burschka, M. Suppa, and G. Hirzinger, "Adaptive and generic corner detection based on the accelerated segment test," in *European conference on Computer vision*, pp. 183–196, Springer, 2010. 14

[45] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," *Image and vision computing*, vol. 22, no. 10, pp. 761–767, 2004. 14

[46] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *2011 International conference on computer vision*, pp. 2564–2571, Ieee, 2011. 14

[47] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," in *European conference on computer vision*, pp. 778–792, Springer, 2010. 14

[48] A. Alahi, R. Ortiz, and P. Vandergheynst, "Freak: Fast retina keypoint," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 510–517, Ieee, 2012. 14

[49] J. Figat, T. Kornuta, and W. Kasprzak, "Performance evaluation of binary descriptors of local features," in *International Conference on Computer Vision and Graphics*, pp. 187–194, Springer, 2014. 14

[50] D. Bojani, K. Bartol, T. Pribani, T. Petkovi, Y. D. Donoso, and J. S. Mas, "On the comparison of classic and deep keypoint detector and descriptor methods," in *2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA)*, pp. 64–69, Sep. 2019. 15

[51] V. Balntas, K. Lenc, A. Vedaldi, and K. Mikolajczyk, "Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors," *CoRR*, vol. abs/1704.05939, 2017. 15

[52] D. G. Lowe, "Local feature view clustering for 3d object recognition," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 1, pp. I–I, IEEE, 2001. 15

[53] S. Obdrzalek and J. Matas, "Object recognition using local affine frames on distinguished regions.," in *BMVC*, vol. 1, p. 3, Citeseer, 2002. 15

[54] F. Schaffalitzky and A. Zisserman, "Automated scene matching in movies," in *International Conference on Image and Video Retrieval*, pp. 186–197, Springer, 2002. 15

[55] F. Schaffalitzky and A. Zisserman, "Multi-view matching for unordered image sets, or how do i organize my holiday snaps?," in *European conference on computer vision*, pp. 414–431, Springer, 2002. 15

[56] D. Tell and S. Carlsson, "Combining appearance and topology for wide baseline matching," in *European Conference on Computer Vision*, pp. 68–81, Springer, 2002. 15

[57] T. Tuytelaars and L. Van Gool, "Wide baseline stereo matching based on local, affinely invariant regions.," in *BMVC*, vol. 412, Citeseer, 2000. 15

[58] I. González-Díaz, C. E. Baz-Hormigos, and F. Díaz-de-María, "A generative model for concurrent image retrieval and roi segmentation," *IEEE Transactions on Multimedia*, vol. 16, no. 1, pp. 169–183, 2014. 15

[59] I. González-Díaz, M. Birinci, F. Díaz-de-María, and E. J. Delp, "Neighborhood matching for image retrieval," *IEEE Transactions on Multimedia*, vol. 19, no. 3, pp. 544–558, 2017. 15

[60] Sivic and Zisserman, "Video google: a text retrieval approach to object matching in videos," in *Proceedings Ninth IEEE International Conference on Computer Vision*, pp. 1470–1477 vol.2, Oct 2003. 15

[61] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2, pp. 2161–2168, Ieee, 2006. 16

[62] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *2007 IEEE conference on computer vision and pattern recognition*, pp. 1–8, IEEE, 2007. 16

[63] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *European conference on computer vision*, pp. 143–156, Springer, 2010. 17, 18

[64] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *2010 IEEE computer society conference on computer vision and pattern recognition*, pp. 3304–3311, IEEE, 2010. 17

[65] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu, "Learning fine-grained image similarity with deep ranking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014. 18

[66] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a "siamese" time delay neural network," in *Proceedings of the 6th International Conference on Neural Information Processing Systems*, NIPS'93, (San Francisco, CA, USA), pp. 737–744, Morgan Kaufmann Publishers Inc., 1993. 18, 49

[67] G. Chechik, V. Sharma, U. Shalit, and S. Bengio, "Large scale online learning of image similarity through ranking," *J. Mach. Learn. Res.*, vol. 11, pp. 1109–1135, Mar. 2010. 18, 22, 36, 51, 60

[68] A. Sharif Razavian, J. Sullivan, A. Maki, and S. Carlsson, "A baseline for visual instance retrieval with deep convolutional networks," in *International Conference on Learning Representations, May 7-9, 2015, San Diego, CA*, ICLR, 2015. 18

[69] A. Babenko and V. Lempitsky, "Aggregating deep convolutional features for image retrieval," *arXiv preprint arXiv:1510.07493*, 2015. 18

[70] Y. Kalantidis, C. Mellina, and S. Osindero, "Cross-dimensional weighting for aggregated deep convolutional features," in *European conference on computer vision*, pp. 685–701, Springer, 2016. 18

[71] A. S. Razavian, J. Sullivan, S. Carlsson, and A. Maki, "Visual instance retrieval with deep convolutional networks," *ITE Transactions on Media Technology and Applications*, vol. 4, no. 3, pp. 251–258, 2016. 18

[72] A. Mousavian and J. Kosecka, "Deep convolutional features for image based retrieval and scene categorization," *arXiv preprint arXiv:1509.06033*, 2015. 18

[73] G. Tolias, R. Sicre, and H. Jégou, "Particular object retrieval with integral max-pooling of cnn activations," *arXiv preprint arXiv:1511.05879*, 2015. 18

[74] F. Radenović, G. Tolias, and O. Chum, "Cnn image retrieval learns from bow: Unsupervised fine-tuning with hard examples," in *Computer Vision – ECCV 2016* (B. Leibe, J. Matas, N. Sebe, and M. Welling, eds.), (Cham), pp. 3–20, Springer International Publishing, 2016. 18, 19, 49

[75] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 18, 20

[76] E.-J. Ong, S. Husain, and M. Bober, "Siamese network of deep fisher-vector descriptors for image retrieval," *arXiv preprint arXiv:1702.00338*, 2017. 18

[77] Y. Verdie, K. Yi, P. Fua, and V. Lepetit, "Tilde: A temporally invariant learned detector," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5279–5288, 2015. 19

[78] K. M. Yi, Y. Verdie, P. Fua, and V. Lepetit, "Learning to assign orientations to feature points," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 107–116, 2016. 19

[79] X. Han, T. Leung, Y. Jia, R. Sukthankar, and A. C. Berg, "Matchnet: Unifying feature and metric learning for patch-based matching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3279–3286, 2015. 19

[80] S. Zagoruyko and N. Komodakis, "Learning to compare image patches via convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4353–4361, 2015. 19

[81] E. Simo-Serra, E. Trulls, L. Ferraz, I. Kokkinos, P. Fua, and F. Moreno-Noguer, "Discriminative learning of deep convolutional feature point descriptors," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 118–126, 2015. 19

[82] J. Zbontar and Y. LeCun, "Computing the stereo matching cost with a convolutional neural network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1592–1599, 2015. 19

[83] V. Balntas, E. Johns, L. Tang, and K. Mikolajczyk, "Pn-net: Conjoined triple deep network for learning local image descriptors," *arXiv preprint arXiv:1601.05030*, 2016. 19

[84] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua, "Lift: Learned invariant feature transform," in *European Conference on Computer Vision*, pp. 467–483, Springer, 2016. 19

[85] H. Noh, A. Araujo, J. Sim, T. Weyand, and B. Han, "Large-scale image retrieval with attentive deep local features," in *Proceedings of the IEEE international conference on computer vision*, pp. 3456–3465, 2017. 19, 49

[86] A. Mishchuk, D. Mishkin, F. Radenovic, and J. Matas, "Working hard to know your neighbor's margins: Local descriptor learning loss," in *Advances in Neural Information Processing Systems*, pp. 4826–4837, 2017. 19

[87] Y. Ono, E. Trulls, P. Fua, and K. M. Yi, "Lf-net: learning local features from images," in *Advances in neural information processing systems*, pp. 6234–6244, 2018. 19

[88] D. Mishkin, F. Radenovic, and J. Matas, "Repeatability is not enough: Learning affine regions via discriminability," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 284–300, 2018. 20

[89] A. Barroso-Laguna, E. Riba, D. Ponsa, and K. Mikolajczyk, "Key. net: Keypoint detection by handcrafted and learned cnn filters," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 5836–5844, 2019. 20

[90] M. Dusmanu, I. Rocco, T. Pajdla, M. Pollefeys, J. Sivic, A. Torii, and T. Sattler, "D2-net: A trainable cnn for joint detection and description of local features," *arXiv preprint arXiv:1905.03561*, 2019. 20

[91] J. Revaud, P. Weinzaepfel, C. De Souza, N. Pion, G. Csurka, Y. Cabon, and M. Humenberger, "R2d2: Repeatable and reliable detector and descriptor," *arXiv preprint arXiv:1906.06195*, 2019. 20

[92] Z. Luo, T. Shen, L. Zhou, J. Zhang, Y. Yao, S. Li, T. Fang, and L. Quan, "Contextdesc: Local descriptor augmentation with cross-modality context," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2527–2536, 2019. 20

[93] H. Taira, M. Okutomi, T. Sattler, M. Cimpoi, M. Pollefeys, J. Sivic, T. Pajdla, and A. Torii, "Inloc: Indoor visual localization with dense matching and view synthesis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7199–7209, 2018. 20

[94] O. Siméoni, Y. Avrithis, and O. Chum, "Local features and visual words emerge in activations," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 11651–11660, 2019. 20

[95] P.-E. Sarlin, C. Cadena, R. Siegwart, and M. Dymczyk, "From coarse to fine: Robust hierarchical localization at large scale," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 12716–12725, 2019. 21

[96] J. Wang, F. Zhou, S. Wen, X. Liu, and Y. Lin, "Deep metric learning with angular loss," *CoRR*, vol. abs/1708.01682, 2017. 22

[97] K. Sohn, "Improved deep metric learning with multi-class n-pair loss objective," in *Advances in Neural Information Processing Systems 29* (D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, eds.), pp. 1857–1865, Curran Associates, Inc., 2016. 22

[98] M. T. Law, N. Thome, and M. Cord, "Quadruplet-wise image similarity learning," in *2013 IEEE International Conference on Computer Vision*, pp. 249–256, 2013. 22

[99] X. Wang, X. Han, W. Huang, D. Dong, and M. R. Scott, "Multi-similarity loss with general pair weighting for deep metric learning," *CoRR*, vol. abs/1904.06627, 2019. 22, 60, 61, 62, 70

[100] B. G. V. Kumar, B. Harwood, G. Carneiro, I. D. Reid, and T. Drummond, "Smart mining for deep metric learning," *CoRR*, vol. abs/1704.01285, 2017. 22

[101] H. Shi, Y. Yang, X. Zhu, S. Liao, Z. Lei, W. Zheng, and S. Z. Li, "Embedding deep metric for person re-identication A study against large variations," *CoRR*, vol. abs/1611.00137, 2016. 22

[102] C. Wang, X. Zhang, and X. Lan, "How to train triplet networks with 100k identities?," *CoRR*, vol. abs/1709.02940, 2017. 22

[103] F. Faghri, D. J. Fleet, J. R. Kiros, and S. Fidler, "VSE++: improved visual-semantic embeddings," *CoRR*, vol. abs/1707.05612, 2017. 22

[104] C. Wu, R. Manmatha, A. J. Smola, and P. Krähenbühl, "Sampling matters in deep embedding learning," *CoRR*, vol. abs/1706.07567, 2017. 22

[105] A. Mishchuk, D. Mishkin, F. Radenovic, and J. Matas, "Working hard to know your neighbor's margins: Local descriptor learning loss," *CoRR*, vol. abs/1705.10872, 2017. 22

[106] X. He, Y. Zhou, Z. Zhou, S. Bai, and X. Bai, "Triplet-center loss for multi-view 3d object retrieval," 2018. 22

[107] X. Zheng, R. Ji, X. Sun, B. Zhang, Y. Wu, and F. Huang, "Towards optimal fine grained retrieval via decorrelated centralized loss with normalize-scale layer," in *AAAI*, 2019. 22

[108] B. Chen and W. Deng, "Deep embedding learning with adaptive large margin n-pair loss for image retrieval and clustering," *Pattern Recognition*, vol. 93, pp. 353 – 364, 2019. 22

[109] Q. Qian, L. Shang, B. Sun, J. Hu, H. Li, and R. Jin, "Softtriple loss: Deep metric learning without triplet sampling," 2019. 22

[110] Z. Wu, A. A. Efros, and S. Yu, "Improving generalization via scalable neighborhood component analysis," in *European Conference on Computer Vision (ECCV) 2018*, 2018. 22, 82

[111] X. Wei, J. Wu, and Q. Cui, "Deep learning for fine-grained image analysis: A survey," *CoRR*, vol. abs/1907.03069, 2019. 22

[112] K. He, Y. Lu, and S. Sclaroff, "Local descriptors optimized for average precision," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 23

[113] X. Wang, Y. Hua, E. Kodirov, G. Hu, R. Garnier, and N. M. Robertson, "Ranked list loss for deep metric learning," *CoRR*, vol. abs/1903.03238, 2019. 23

[114] J. Revaud, J. Almazan, R. S. Rezende, and C. R. d. Souza, "Learning with average precision: Training image retrieval with a listwise loss," in *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 23, 56, 60, 61, 79

[115] F. Cakir, K. He, X. Xia, B. Kulis, and S. Sclaroff, "Deep metric learning to rank," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 23

[116] T.-Y. Liu, *Learning to Rank for Information Retrieval.* Springer, 2011. 23

[117] R. Arandjelovic, P. Gronát, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: CNN architecture for weakly supervised place recognition," *CoRR*, vol. abs/1511.07247, 2015. 23

[118] A. Gordo, J. Almazán, J. Revaud, and D. Larlus, "End-to-end learning of deep visual representations for image retrieval," *International Journal of Computer Vision*, vol. 124, pp. 237–254, Sep 2017. 23

[119] S. Sukhbaatar, J. Bruna, M. Paluri, L. Bourdev, and R. Fergus, "Training convolutional networks with noisy labels," 2014. 23, 24

[120] I. González-Díaz, J. Benois-Pineau, J.-P. Domenger, D. Cattaert, and A. [de Rugy], "Perceptually-guided deep neural networks for ego-action prediction: Object grasping," *Pattern Recognition*, vol. 88, pp. 223 – 235, 2019. 23, 24

[121] G. Patrini, A. Rozza, A. Menon, R. Nock, and L. Qu, "Making deep neural networks robust to label noise: a loss correction approach," 2016. 24

[122] D. Hendrycks, M. Mazeika, D. Wilson, and K. Gimpel, "Using trusted data to train deep networks on labels corrupted by severe noise," 2018. 24

[123] Z. Zhang and M. R. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," *CoRR*, vol. abs/1805.07836, 2018. 24

[124] J. Han, P. Luo, and X. Wang, "Deep self-learning from noisy labels," 2019. 24

[125] K.-H. Lee, X. He, L. Zhang, and L. Yang, "Cleannet: Transfer learning for scalable image classifier training with label noise," 2017. 24

[126] M. Ren, W. Zeng, B. Yang, and R. Urtasun, "Learning to reweight examples for robust deep learning," 2018. 24

[127] J. Li, Y. Wong, Q. Zhao, and M. Kankanhalli, "Learning to learn from noisy labeled data," 2018. 24

[128] M. Sun, T. X. Han, M.-C. Liu, and A. Khodayari-Rostamabad, "Multiple instance learning convolutional neural networks for object recognition," 2016. 24

[129] P. H. O. Pinheiro and R. Collobert, "From image-level to pixel-level labeling with convolutional networks," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pp. 1713–1721, 2015. 24

[130] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Is object localization for free? - weakly-supervised learning with convolutional neural networks," pp. 685–694, 2015. 24

[131] A. Diba, V. Sharma, A. Pazandeh, H. Pirsiavash, and L. Van Gool, "Weakly supervised cascaded convolutional networks," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5131–5139, July 2017. 24

[132] A. Kolesnikov and C. H. Lampert, "Seed, expand and constrain: Three principles for weakly-supervised image segmentation," in *European Conference on Computer Vision (ECCV)*, pp. 695–711, Springer, 2016. 24

[133] T. Durand, N. Thome, and M. Cord, "Weldon: Weakly supervised learning of deep convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4743–4752, June 2016. 24

[134] T. Durand, T. Mordan, N. Thome, and M. Cord, "WILDCAT: Weakly Supervised Learning of Deep ConvNets for Image Classification, Pointwise Localization and Segmentation," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5957–5966, 2017. 24

[135] J. H. Su, C. Y. Chin, J. Y. Li, and V. S. Tseng, "Efficient big image data retrieval using clustering index and parallel computation," in *2017 IEEE 8th International Conference on Awareness Science and Technology (iCAST)*, pp. 182–187, Nov 2017. 25

[136] Y. Cao, M. Long, J. Wang, and S. Liu, "Deep visual-semantic quantization for efficient image retrieval," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 916–925, July 2017. 25

[137] X. Lu, L. Song, R. Xie, X. Yang, and W. Zhang, "Deep hash learning for efficient image retrieval," in *2017 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, pp. 579–584, July 2017. 25

[138] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 770–778, 2016. 26, 38

[139] M. Thoma, "A survey of semantic segmentation," 2016. 26

[140] L. Wang, D. Li, Y. Zhu, L. Tian, and Y. Shan, "Dual super-resolution learning for semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 26

[141] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," 2018. 26

[142] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *International Journal of Computer Vision*, vol. 128, pp. 261–318, Feb 2020. 26

[143] S. Devi, P. Malarvezhi, R. Dayana, and K. Vadivukkarasi, "A comprehensive survey on autonomous driving cars: A perspective view," *Wireless Personal Communications*, vol. 114, pp. 2121–2133, Oct 2020. 26

[144] W. Zhou, H. Li, and Q. Tian, "Recent advance in content-based image retrieval: A literature survey," *CoRR*, vol. abs/1706.06064, 2017. 26

[145] M. Kaya and H. Ş. Bilge, "Deep metric learning: A survey," *Symmetry*, vol. 11, no. 9, p. 1066, 2019. 28

[146] I. Melekhov, J. Kannala, and E. Rahtu, "Siamese network features for image matching," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 378–383, Dec 2016. 28

[147] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical Association*, vol. 66, no. 336, pp. 846–850, 1971. 37

[148] E. B. Fowlkes and C. L. Mallows, "A method for comparing two hierarchical clusterings," *Journal of the American Statistical Association*, vol. 78, no. 383, pp. 553–569, 1983. 37

[149] A. Ben-Hur, A. Elisseeff, and I. Guyon, "A stability based method for discovering structure in clustered data," vol. 2002, pp. 6–17, 02 2002. 37

[150] S. Wagner and D. Wagner, "Comparing clusterings - an overview," *Technical Report 2006-04*, 01 2007. 37

[151] S. Bell and K. Bala, "Learning visual similarity for product design with convolutional neural networks," *ACM Transactions on Graphics*, vol. 34, pp. 98:1–98:10, 07 2015. 49

[152] M.-A. Carbonneau, V. Cheplygina, E. Granger, and G. Gagnon, "Multiple instance learning: A survey of problem characteristics and applications," *Pattern Recognition*, vol. 77, p. 329353, May 2018. 50

[153] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The caltech-ucsd birds-200-2011 dataset," Tech. Rep. CNS-TR-2011-001, California Institute of Technology, 2011. 61

[154] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, "Deepfashion: Powering robust clothes recognition and retrieval with rich annotations," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1096–1104, 2016. 61

[155] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, June 2007. 61, 62

[156] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, June 2008. 61, 62

[157] F. Radenovic, A. Iscen, G. Tolias, Y. Avrithis, and O. Chum, "Revisiting oxford and paris: Large-scale image retrieval benchmarking," *CoRR*, vol. abs/1803.11285, 2018. 61, 62

[158] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations*, 12 2014. 61

[159] H. Jegou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *Computer Vision – ECCV 2008* (D. Forsyth, P. Torr, and A. Zisserman, eds.), (Berlin, Heidelberg), pp. 304–317, Springer Berlin Heidelberg, 2008. 62

[160] Y. Mohsenzadeh, H. Sheikhzadeh, and S. Nazari, "Incremental relevance sample-feature machine: A fast marginal likelihood maximization approach for joint feature selection and classification," *Pattern Recognition*, vol. 60, pp. 835 – 848, 2016. 71, 75